Title
Systems Theory for Pharmaceutical Drug Discovery

Author
Aswani, Anil Jayanti

Publication Date
2010

Peer reviewed|Thesis/dissertation

Systems Theory for Pharmaceutical Drug Discovery

by

Anil Jayanti Aswani

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering - Electrical Engineering and Computer Sciences

and the Designated Emphasis

in

Computational and Genomic Biology

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Claire Tomlin, Chair
Professor Laurent El Ghaoui
Professor Peter Bickel

Spring 2010

Systems Theory for Pharmaceutical Drug Discovery

Abstract

Systems Theory for Pharmaceutical Drug Discovery

by

Anil Jayanti Aswani

Doctor of Philosophy in Engineering-Electrical Engineering and Computer Sciences

and the Designated Emphasis in Computational and Genomic Biology

University of California, Berkeley

Professor Claire Tomlin, Chair

Biological networks are comprised of thousands of interacting components, and these networks have complicated patterns of feedback and feed-forward motifs. It is practically impossible to use intuition to determine whether simultaneously modifying multiple pharmaceutical targets has a good therapeutic response. Even when a drug is discovered which is safe in humans and highly-effective against its target, the medical effect on the disease may be underwhelming. This provides a strong impetus for developing a systems theory for pharmaceutical drug discovery. This thesis discusses system theoretic tools which are useful for doing drug discovery. The first class of tools discussed is system identification tools, and case studies of parametric modeling are given. A new statistical system identification procedure which exploits the geometric and hierarchical structure of many biological (and engineering) systems is presented, and this new procedure is applied to engineering and biological systems. The second class of tools discussed is a new set of target selection tools. Given mathematical models of biological networks, these tools select a set of targets for pharmaceutical drugs. The targets are selected to achieve good medical outcomes for patients by reducing the effect of diseases on pathways and ensuring that the targets do not too adversely affect healthy cells. The ultimate goal of the work presented in this thesis is to create a framework which can be used to rationally select new drug targets and also be able to create personalized medicine treatments which are tailored to the particular phenotypic behavior of an individual's disease.

To God...

Contents

List of Figures

List of Tables

Acknowledgments

It has been an honor to work with Claire and Peter. I have always considered it a bit of serendipity that Claire joined Berkeley as faculty the same year I started my graduate studies. Her research interests and style were a good fit for me, and working under Claire has led to fruitful results. I have learned much from her guidance and advice, and it will continue to be valued after I graduate. I began working with Peter, on a project with LBL, as an adviser for my minor. His experience and philosophy provided new ways to envision the world, and his humor injected a bit of levity into research.

I am grateful for the emotional support of my family and loved ones, without which I would have not been able to finish. My father Jayanti and my mother Maya provided frequent conversations of encouragement, and always offered assistance in financial matters. My brother Deepak always supported my decisions and his career advice was invaluable. Malini has provided immeasurable emotional encouragement, in spite of her challenging work.

I would like to show my gratitude to the other individuals which have been important in my time at Berkeley. The undergraduate students I officially advised—Nick, Hari, Eugene, and Tremaine—were a pleasure to work with and helped me to progress as an individual, in addition to advancing my research; the other students I unofficially advised provided valuable experiences. Mark Biggin and others at LBL were instrumental in much of my biology research, and results would not have been possible without their data and biological expertise. Ping Hsu provided excellent teaching mentorship, and Ruth helped in navigating paperwork. Laurent and Yun Song provided useful conversations. There were countless other students that provided the unique experiences in attending Berkeley.

Chapter 1

Introduction

The discovery of drugs has a long history with ancient beginnings. The first drugs were herbal substances discovered by hunter-gatherer societies through serendipity and trial-and-error. Rational drug design would not begin until later in history, and it was fueled by advances in basic science and biology by people like Antoine van Leeuwenhoek, Charles Darwin, Claude Bernard, Robert Koch, and Louis Pasteur. Still, many of the early successes such as the smallpox vaccinations of Edward Jenner (also from Berkeley, though pronounced differently and in another country) or the antibiotics of Paul Ehrlich were discovered through keen observation or more methodical forms of trial-and-error.

Rational drug discovery began to grow in the late-19th and 20th century. The creation of the Food and Drug Administration by Teddy Roosevelt would eventually lead to an increase in the amount of scientific research and evidence required before a new drug could be legally sold. It became necessary to show the efficacy and safety of a new drug, and this process was both time-consuming and expensive. Failures in the approval process, such as thalidomide (though the United States did not approve the drug), led to even further restrictions on the drug approval process.

Drug targets—molecules within a cell which are affected by the drug in order to cause an improvement in the disease—were selected through lengthy, microbiology research on single molecules, their relationship to diseases, and the effect of the interruption or changing of the molecule. Much of drug design, the process of determining chemical compounds which affect the target, was (and still is to some extent) conducted through trial-and-error on large libraries of chemical compounds. Experiments on animal models and clinical trials on humans added to the length and cost of discovering new drugs.

Drug design, which is the selection of a chemical compound, has begun to change in recent years. The increasing power of computers has enabled the use of computational tools to help with this process. Models founded in physical chemistry can be used to predict the chemical properties of compounds and the ability of existing and novel compounds to bind with or affect a selected drug target. Chemometric approaches have also been used to varying extents. Such tools have helped improve the chance that a new drug can successfully pass through the regulatory process.

Yet the pharmaceutical industry is facing increasing difficulties. Discovering new drugs has become more difficult, and there is a sense in the industry that the "low-hanging fruit" has been picked. An increasing number of potential drugs never survive the lengthy approval process due to reasons of safety or efficacy. Even when a drug is discovered which is highly-effective against its target and safe in humans, the medical effect on the disease may be underwhelming, as has been the case with Gefitinib which strongly represses its target protein HER2 which is implicated

in many human cancers. This provides a strong impetus for new approaches and techniques for drug discovery.

One area that has not been greatly explored by academic and pharmaceutical researchers is the process of drug target selection. Recent successes in the treatment of cancer and HIV by the use of so-called "drug cocktails" (the use of multiple drugs, each with a different target) suggest multi-target therapy as a promising direction for the future. Unfortunately, the pharmaceutical industry is weary of pursuing such approaches because of the high-perceived risk of novel targets as opposed to established targets. Some of this discomfort is justified, because the process of target selection is not done in a rigorous or principled manner.

Biological networks are comprised of thousands of interacting components, and there are complicated patterns of feedback and feed-forward motifs in these networks. Trying to intuit the effect of simultaneously modifying multiple (or even one) targets is impossible. Just as was the case with drug design, the natural progression is the development of computational tools to aid in this daunting task. There is great cause for optimism in this endeavor because of the successes in engineering spurred on by computational, systems-theoretic, and statistical tools.

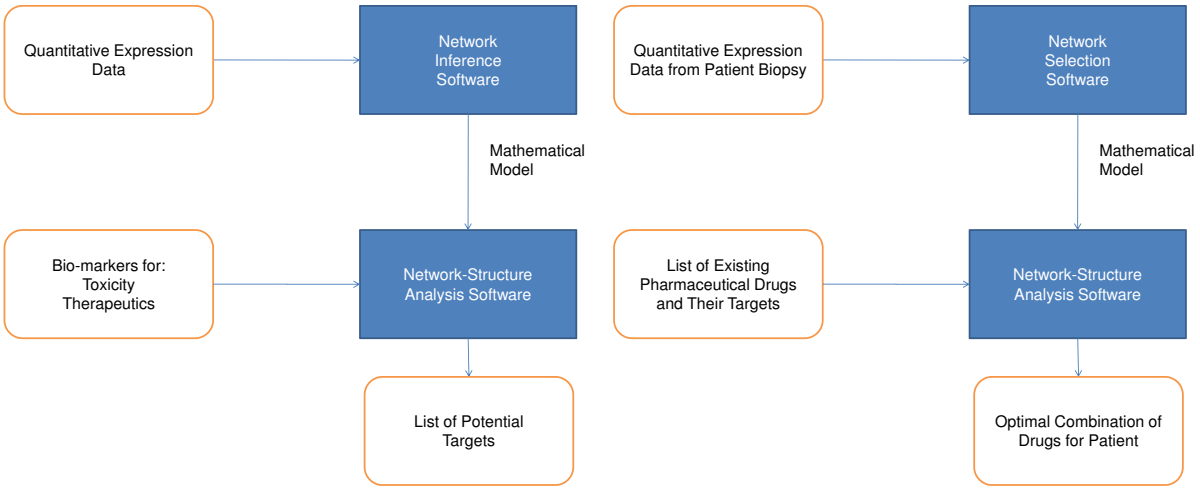
Two tool sets are necessary for developing a unified framework for systems-level drug discovery: system identification and target selection. System identification is synonymous with mathematical modeling, and in this context, it is the process of developing a set of equations which describe the behavior of a biological network. In this thesis, the focus is on networks that are biomolecular pathways; though nothing precludes the use of these techniques on other networks or even engineering systems. In contrast, target selection will be the process of taking a mathematical model of a network, selecting components of this network which affect positive changes when influenced by a drug, and understanding the behavior of the network with and without the affecting the selected components.

The real goal of this work is to create a framework which can be used to rationally select new drug targets and also to be able to create personalized medicine treatments which are tailored to the particular phenotypic behavior of an individual's disease. In the first case, which is illustrated in Figure 1.1a, we would conduct high-throughput experiments, generate a mathematical model, and use this mathematical model to identify multiple drug targets which can be candidates for the traditional process of drug discovery. In the second case, which is illustrated in Figure 1.1b, we would take a sample of diseased tissue from a patient, submit the sample to high-throughput laboratory tests, and then select an optimal choice of existing drugs to give to the particular patient based on the manifestation and specific sub-type of the disease.

1.1 System Identification

System identification has its origins in the natural philosophy of individuals like Ibn al Haytham, Robert Boyle, Galileo Galilei, and Isaac Newton. Developing mathematical models to describe a natural phenomenon came with increased interest in quantification of said phenomena. This historically began with classical physics where experimentation was relatively easy because of the size of objects studied. The quantification and modeling of chemistry began at a later date because of the increased difficulty in dealing with atomic and molecular objects. Biology would not begin to be mathematically modeled until recently; one of the earliest such works was the seminal paper by Alan Turing on morphogenesis.

The increased difficulty in doing biological experiments has stunted growth in the quantification and modeling of biology, but the rapid increase in new technologies presents great op-



(a) Drug Discovery Flow (b) Personal Medicine Flow

Figure 1.1: The goal of this research is to create a framework which can be used to rationally select new drug targets and also to be able to create personalized medicine treatments which are tailored to the particular phenotypic behavior of an individual's disease.

portunity. High-throughput technologies allow for the gathering of large amounts of quantitative data, and the question has turned to how to best utilize this data. There is great similarity in the types of data obtained in this biological context and in engineering projects, and this suggests the application of engineering techniques and tool sets to modeling biological networks. There is great variety in the approaches that might be used, and there is the possibility for developing new computational tools which exploit structure and constraints unique to biological and engineering systems.

System identification is the process of using experimental data to create a mathematical model which describes a system, and it has been the focus of significant research in the engineering community. Broadly speaking, system identification can be classified into either parametric or nonparametric approaches. Parametric approaches are well suited to when there is substantial existing knowledge about the system to be modeled, while nonparametric approaches are well suited to situations in which there is a large amount of experimental data and little existing knowledge about the system. The existing knowledge can range from physics-based equations like Newton's laws to domain-knowledge on a biological pathway generated from gene-knockout experiments.

In parametric system identification, one uses existing knowledge to write down an equation structure which describes the system. This equation has unknown parameters which are determined using the experimental data. A very classical example is a mass-spring system in which a mass is attached to a wall with a spring, and the mass can slide along the floor without the influence of frictional forces. Newton's equations and Hooke's law can be used to write a model for this system $\ddot{x} = -kx$, and then experimental measurements of the acceleration $\ddot{x}$ and position x of the block can be used to determine the parameter k, which is the spring constant. Determining the parameters can be posed as a regression problem, and there is a well-studied, statistical theory on this topic (though the issue of convergence to local minima in nonlinear regression is swept under the rug). Identifiability of the parameters is typically ignored in practice, but in general it requires Lipschitz continuity of the model with respect to the parameters.

Nonparametric system identification is less-commonly used, but there is growing interest in these techniques spurred on by the machine-learning and artificial-intelligence communities. In this class of techniques, a system is modeled with a truncated series expansion; ostensibly if the series were infinitely long, then the system would be modeled exactly with the correct choice of parameter values. The limitations of this technique are with the large amounts of data required to get accurate parameter values, and the asymptotic amounts of data required have been characterized by statisticians. Smaller amounts of data lead to inaccurate parameter values, and this leads to inaccurate models.

In the context of engineering and of modeling biological networks, both techniques have their uses depending on the system and assumptions made about it. Parametric modeling is well-studied, and likely only minimal advances in the computational and statistical properties of these techniques is possible. On the other hand, parametric modeling is undergoing a recent resurgence in interest spurred on by the exploitation of special structure, such as sparsity or manifold structure, to improve the performance of such techniques. Such structure is commonly found in engineering systems and biological systems, so the development of techniques which use this structure is important.

1.2 Drug Multi-Target Selection

When designing pharmaceuticals for a disease, researchers currently use their intuition and knowledge about the particular disease modalities to choose which cellular components to target with pharmaceuticals. Such an approach is limited by intuition, and one can only consider a few cellular components at a time. In the present situation, the pharmaceutical industry has remained risk-averse because of the lack of accurate predictions and knowledge afforded by such intuition.

Yet in recent years, there has been a recognition that it can be beneficial to use several different drugs, so called "drug cocktails". Unfortunately, simultaneously choosing and designing drugs to target multiple components of a disease modality is a difficult task. It is difficult to predict the simultaneous effect of multiple drugs on a large network of cellular components. Furthermore, it is difficult to choose what combination of drugs provides the best clinical outcome.

An important problem facing the health-care industry is how to systematically do drug discovery. The first aspect of this is: How should the targets for drugs be chosen? A related problem is: How can drugs be designed to minimize their adverse effects on healthy cells? These are challenging problems, but they are also important ones. Computational tools which can select novel targets for diseases and accurately predict the effect of modifying those targets will enable the pharmaceutical industry to more strategically take on acceptable levels of risk in the development of new drugs. Hopefully, this will lead to new drugs, or combinations of existing drugs, which are effective against diseases. There is the issue of ensuring that the designed drugs only affect the desired targets and not other parts of the network, but this is one of the things that drug companies are relatively good at doing.

Attempts to aid with drug discovery using systems tools are relatively recent, and these tools have not yet been applied in the pharmaceutical industry. One large class of tools solves the problem of either ensuring viability or ensuring non-viability, and this is useful for when one would like to kill cancer cells or bacterial pathogens. Another class of tools tries to suitably modify the steady-state behavior of diseased pathways to make them behave more favorably. These tools have specific applicability, and this necessitates the development of newer tools which better encode pharmaceutical and medical effects through appropriate mathematical models and tools.

1.3 Organization

This thesis is organized into two parts. The first part discusses system identification, and the second part discusses target selection. For the most part, proofs are relegated to the Appendix except in a few cases where the proofs are short. The focus of this thesis is on drug discovery, but much of the content is equally applicable to engineering systems. An application of our new system identification tools to a real engineering system is given in the Appendix, and notes are scattered throughout whenever something is applicable to engineering systems. Though the introduction contains no citations, the appropriate references to earlier work will be given in each chapter. Lastly, the contributions of this thesis are summarized below.

The first part of the thesis covers system identification. Parametric modeling of biological systems is the subject of the first two chapters, and some of the insights provided by these models are discussed. Next, a statistical system identification procedure is given. The novel feature of this procedure is the development of new statistical tools which consider geometrical and hierarchical organization in systems in order to improve the estimation and identification of models. Such structure is commonly found in biological, engineering, and financial systems. This part of the thesis concludes with a chapter on multi-task learning, and specifically answers the question of how to combine local models in order to do variable selection and build a picture of global structure.

The second part of the thesis covers pharmaceutical drug target selection. The effect of pharmaceuticals can be abstracted to a graph-theoretic interpretation, and this provides the content of the first chapter. The question of selecting drug targets can be posed as a graph-theoretic problem or as an optimization problem. Algorithms for doing target selection are provided, and these algorithms are designed to be scalable to large networks and biological pathways. These algorithms are applied to a particular cancer pathway to demonstrate their ability to select reasonable targets, and simulations show that drugs which affect the selected targets do indeed meet the desired medical objectives.

Part I

System Identification

Chapter 2

Planar Cell Polarity in *Drosophila melanogaster*

Identifying mathematical models of biological systems is important for doing drug discovery, because these models can be used with drug multi-target selection tools which aid with drug discovery. Though these target selection tools are the subject of the second part of this thesis, this first chapter and part of the thesis begins with a case study of the identification of a network for which there is significant prior, biological knowledge. This allows us to use standard parametric system identification tools in generating the models.

Drosophila embryogenesis is divided into different stages [127, 119] of development, during which the embryo undergoes the development of particular features. Concentrations of gene products (mRNA or protein) control the physical development of structures such as limbs or nerves, and this patterning occurs through a complicated network of interactions between proteins, mRNA, and DNA [127, 119]. These interactions are highly nonlinear because processes such as transcription, translation, and diffusion lead to complicated biochemical behaviors.

A general system found not only in *Drosophila* but also in mammals (including humans) is planar cell polarity (PCP), which is the organization of a planar array of cells. Though it is a common motif in many organisms, some parts of this phenomenon are still poorly understood. More importantly, this is an important mechanism because it is involved in certain human conditions, including: misalignment of cochlear receptor cells, spina bifida, and certain types of cancers.

An interesting example of PCP is the orientation of the direction of hair growth on the wings of *Drosophila melanogaster*, and this orientation of wing hairs occurs during the puparium stage of development. It is worth noting that this species of *Drosophila* is often referred to as the common fruit fly. As seen in Figure 2.1, the hair on the wings of *Drosophila* grows in a uniform direction, towards the distal side of the wing. This alignment of hair is a robust system. It is able to function well in the presence of environmental variations or local genetic mutations. This particular PCP system is well-suited for study: A wide-amount of experimental research has been conducted to date on this particular system, and it is relatively easy to gather additional experimental data.

The aim of this chapter is to perform system identification on a particular instance of the PCP system, and this is done using standard tools from parametric system identification. The hope is that by learning a mechanistic model, we can then use it to help develop treatments for malfunctioning PCP systems in humans. It may be possible to use these models to gain a better understanding of the biochemical basis for certain related human diseases. The identified model can potentially be used to understand the effects of treatments on human variants of this system.

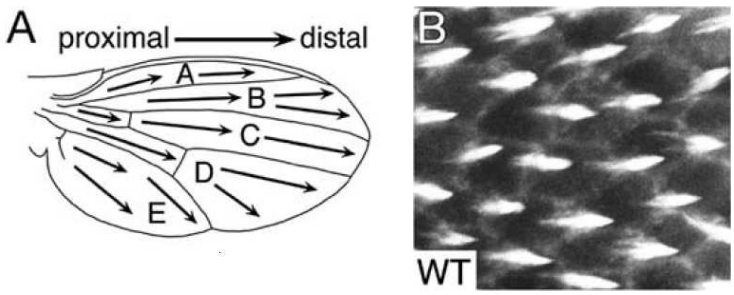


Figure 2.1: The wing hairs in *Drosophila* align towards the distal edge of the wing [180].

This work was done in collaboration with the Axelrod Laboratory in the Medical School at Stanford University.

2.1 Literature Review

A variety of models for PCP on *Drosophila* wings have been proposed. One class of models requires the presence of hypothesized diffusible factors, referred to as factor X or Z [1]; however, such diffusible factors have not been experimentally observed or identified. Another proposed model [203] incorporates only the currently identified proteins, and the corresponding mathematical model [5] reproduces all experimentally observed phenotypes.

The mathematical model proposed [5] works very well with reproducing the experimental data collected to date. However, the model assumes that the proteins involved in PCP move through diffusion. Recent experimental evidence [180] implies that this is not the case. This newer data suggests that the proteins Dishevelled (Dsh), Frizzled (Fz), and Flamingo (Fmi) are transported via microtubules. Microtubules can be thought of as unidirectional railroads inside a cell; they carry material throughout the cell, along fixed paths.

Experimental and genetic analysis has implicated the proteins: Fz, Dsh, Prickle Spiny-legs (Pk), Van Gogh (Vang), and Fmi as key players in the establishment of PCP on the *Drosophila* wing [205, 23, 193, 203, 26]. Fluorescent tagging of these proteins has shown that in an unmutated wing cell, Dsh and Fz localize to the distal edge of a single cell—whereas Pk and Vang localize to the proximal edge of a single cell. Fmi is found at both the proximal and distal edges of a single cell, but not at the other boundaries. This can be seen in Figure 2.2. Furthermore, the hair on a single cell has been observed to grow in the direction of greatest Dsh concentration [5].

Dsh has been shown to bind with Pk and Vang [23], and Fmi has been shown to bind with another copy of Fmi [205]. But, direct reactions between other proteins have not yet been determined. However, the colocalization of Dsh, Fz, and Fmi has been observed; similarly, a colocalization of Pk, Vang, and Fmi has been observed [118]. This has led to the hypothesization of the existence of DshFzFmi and PkVangFmi complexes. Furthermore, the unique localization of the proteins has led to the hypothesis of an ectodomain (intercelluar) repulsion of Fz with Fz on an adjacent cell and Pk with Pk on an adjacent cell [5, 118]. The picture of the model is then that Dsh, Fz, and Fmi organize into a large complex, with a similar organization of Pk, Vang, and Fmi. These two complexes then form an intercellular complex across cell membranes—a manifestation of the hypothesized ectodomain interaction.

It has been observed that Pk and Vang inhibit the recruitment of Dsh to a cell edge [5].

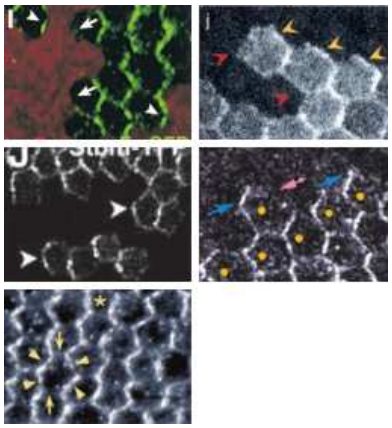


Figure 2.2: Protein localization is important to understanding planar cell polarity (PCP). The particular localization of Fz, Dsh, Vang, Pk, and Fmi [205, 23, 193, 203, 26] are shown from left to right and top to bottom.

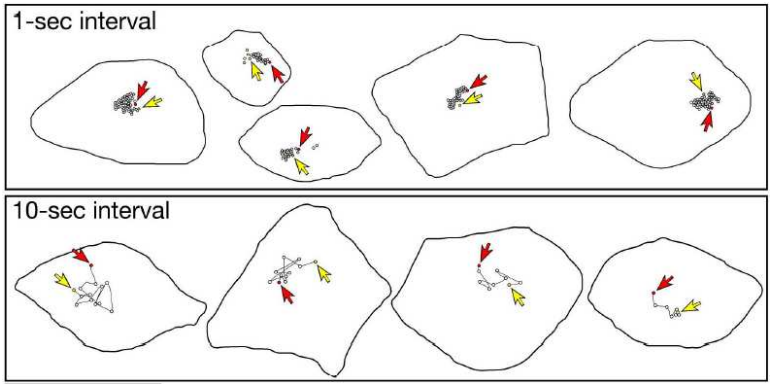


Figure 2.3: Some sample paths of intracellular transport of Fz [180] show its biased stochastic nature.

The exact nature of the inhibition is unknown, but what is known is that Dsh can bind with Pk and Vang. However, this inhibition is not due to competitive binding (i.e., Dsh and Pk bind which prevents Fz and Dsh from binding) because Dsh is not colocalized with Pk and Vang.

Recent evidence shows that Fz, Dsh, and Fmi move inside the cell along proximal-distal paths [180]. The motion is fairly chaotic at short time-intervals, but over longer time intervals the motion is more systematic. Several sample paths are shown in Figure 2.3. Moreover, this movement was found to be associated with microtubules. Interestingly, the microtubules have been observed to align on the proximal-distal axis of a cell, as seen in Figure 2.4. Furthermore, a slight bias in the number of microtubules pointing towards the distal edge, as opposed to pointing towards the proximal edge, has been observed.

2.2 Mathematical Modeling

The mathematical model developed must be able to reproduce not only the phenotype (observed, biological behavior) of the correct hair direction for wildtype (unmutated) *Drosophila*,

it must also be able to reproduce the correct phenotype for *Drosophila* with local mutations in the wings. Ideally, the model would match actual protein concentrations within the cells, but it is difficult to get such measurements. As a result, the phenotype of hair direction is used to qualitatively judge the model.

Before the model is presented, it is useful to note the assumptions made. Protein concentrations are assumed to be continuous variables. This is not actually the case in real life because only a discrete number of protein molecules can exist. This assumption can be interpreted as meaning that our continuous protein concentration represents an average concentration over a small region. Another simplifying assumption is that inhibition of a reaction is equivalent to no reaction at all.

At the core of the model is a set of protein interactions. These protein interactions are modeled as the binding of several proteins into larger complexes. It is not known whether these complexes actually exist, but the logical interaction of the proteins is still maintained by this model. Another feature of this model is that it has been designed to maintain a high level of generality. In order to maintain consistency with the experimental observations, the model considers a series of

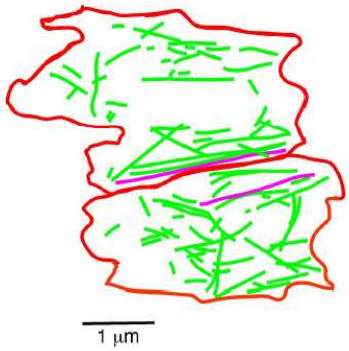


Figure 2.4: Microtubules preferentially align along the proximal-distal axis [180].

reactions. Note that the dagger (†) symbol represents proteins in an adjacent cell membrane:

$$\begin{aligned}
\mathrm{Dsh} + \mathrm{Fz} &\underset{\lambda_1}{\overset{R_1}{\rightleftharpoons}} \mathrm{DshFz} \\
\mathrm{Fz} + \mathrm{Pk}^\dagger &\underset{\lambda_2}{\overset{R_2}{\rightleftharpoons}} \mathrm{FzPk}^\dagger \\
\mathrm{Pk} + \mathrm{Vang} &\underset{\lambda_2}{\overset{R_3}{\rightleftharpoons}} \mathrm{PkVang} \\
\mathrm{DshFz} + \mathrm{Pk}^\dagger &\underset{\lambda_4}{\overset{R_4}{\rightleftharpoons}} \mathrm{DshFzPk}^\dagger \\
\mathrm{Dsh} + \mathrm{FzPk}^\dagger &\underset{\lambda_5}{\overset{R_5}{\rightleftharpoons}} \mathrm{DshFzPk}^\dagger \\
\mathrm{Fz} + \mathrm{Pk}^\dagger\mathrm{Vang}^\dagger &\underset{\lambda_6}{\overset{R_6}{\rightleftharpoons}} \mathrm{FzPk}^\dagger\mathrm{Vang}^\dagger \\
\mathrm{FzPk}^\dagger + \mathrm{Vang}^\dagger &\underset{\lambda_7}{\overset{R_7}{\rightleftharpoons}} \mathrm{FzPk}^\dagger\mathrm{Vang}^\dagger \\
\mathrm{DshFz} + \mathrm{Pk}^\dagger\mathrm{Vang}^\dagger &\underset{\lambda_8}{\overset{R_8}{\rightleftharpoons}} \mathrm{DshFzPk}^\dagger\mathrm{Vang}^\dagger \\
\mathrm{DshFzPk}^\dagger + \mathrm{Vang}^\dagger &\underset{\lambda_9}{\overset{R_9}{\rightleftharpoons}} \mathrm{DshFzPk}^\dagger\mathrm{Vang}^\dagger \\
\mathrm{Dsh} + \mathrm{FzPk}^\dagger\mathrm{Vang}^\dagger &\underset{\lambda_{10}}{\overset{R_{10}}{\rightleftharpoons}} \mathrm{DshFzPk}^\dagger\mathrm{Vang}^\dagger
\end{aligned}$$

In this model, the reaction rates (i.e., R_1, λ_1, etc.) are all constants. This is a departure from the earlier model [5] which contained reaction rates that were functions of other protein concentrations, in order to model the inhibition of reactions. In this newer model, the inhibition of reactions is modeled as the lack of a reaction. For example, it is hypothesized that Fz inhibits Fz on an adjacent cell. This is modeled as the lack of a reaction involving the creation of a $\mathrm{DshFzFz}^\dagger\mathrm{Dsh}^\dagger$ complex. This greatly simplifies the model, and reduces the numerical and computational complexity of model.

The most novel feature of this new model is the addition of a model of microtuble transport of proteins. This transport is modeled as the convection of protein from the cell membrane into the microtubules. The rate of convection is assumed to be constant, but what varies is the number of microtubules leading from one edge to another and the rate of endocytosis of DshFz complexes caused by $\mathrm{PkVang}^\dagger$ complexes (as seen in Figure 2.5a). Lastly, the motion within the microtubule is modeled as convection.

The model for the microtubule transport represents three experimentally observed features. First, the number of microtubules pointing towards the distal end slightly outnumbers the number pointing towards the proximal end. This disparity in direction is modeled as different rates of convection, since it is expected that a larger number of microtubules would lead to faster transport of proteins. Second, Dsh binds with Pk and Vang, but it also inhibits the recruitment of Dsh. The hypothesis underlying this model is that Pk and Vang cause the loading of DshFz into microtubules (technically speaking, vesicles associated with the microtubules) by inducing endocytosis. In other words, the hypothesis is that a larger concentration of PkVang complex leads to an increase in the rate of DshFz endocytosis, and the asymmetry in protein concentration is due to the asymmetry in the microtubule distribution as shown in Figure 2.5b.

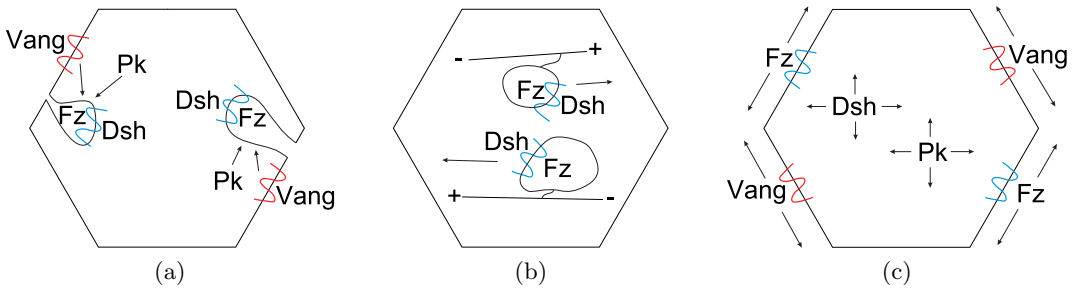


Figure 2.5: (a) PkVang complexes on the membrane of a cell lead to the endocytosis of DshFz complexes, which then become attached to microtubules and are actively transported. The active transport along microtubules is mathematically modeled as convection. (b) There is a slight asymmetry of microtubules which direct vesicles towards the distal edge of the cell. The distal edge is on the right side of the cell. (c) Some proteins diffuse in the cytoplasm, whereas the membrane-bound proteins only diffuse on the membrane. Complexes between adjacent cells also diffuse along the membrane.

Defining the net forward rate of reactions as:

$$
\begin{aligned}
P_1 &= R_1[\mathrm{Dsh}][\mathrm{Fz}] - \lambda_1[\mathrm{DshFz}] \\
P_2 &= R_2[\mathrm{Fz}][\mathrm{Pk}^\dagger] - \lambda_2[\mathrm{FzPk}^\dagger] \\
P_3 &= R_3[\mathrm{Pk}][\mathrm{Vang}] - \lambda_3[\mathrm{PkVang}] \\
P_4 &= R_4[\mathrm{DshFz}][\mathrm{Pk}^\dagger] - \lambda_4[\mathrm{DshFzPk}^\dagger] \\
P_5 &= R_5[\mathrm{Dsh}][\mathrm{FzPk}^\dagger] - \lambda_5[\mathrm{DshFzPk}^\dagger] \\
P_6 &= R_6[\mathrm{Fz}][\mathrm{Pk}^\dagger\mathrm{Vang}^\dagger] - \lambda_6[\mathrm{FzPk}^\dagger\mathrm{Vang}^\dagger] \\
P_7 &= R_7[\mathrm{FzPk}^\dagger][\mathrm{Vang}^\dagger] - \lambda_7[\mathrm{FzPk}^\dagger\mathrm{Vang}^\dagger] \\
P_8 &= R_8[\mathrm{DshFz}][\mathrm{Pk}^\dagger\mathrm{Vang}^\dagger] - \lambda_8[\mathrm{DshFzPk}^\dagger\mathrm{Vang}^\dagger] \\
P_9 &= R_9[\mathrm{DshFzPk}^\dagger][\mathrm{Vang}^\dagger] - \lambda_9[\mathrm{DshFzPk}^\dagger\mathrm{Vang}^\dagger] \\
P_{10} &= R_{10}[\mathrm{Dsh}][\mathrm{FzPk}^\dagger\mathrm{Vang}^\dagger] - \lambda_{10}[\mathrm{DshFzPk}^\dagger\mathrm{Vang}^\dagger]
\end{aligned}
$$

It is not clear whether or not the complexes diffuse along the cell membrane. In general, proteins within the membrane are free to diffuse along the membrane; however, certain proteins are rigidly fixed in the membrane. To allow for maximum generality, the model allows all molecules to

diffuse, but at different rates. The model along the edge of the cell is:

$$\begin{aligned}
\partial_t[\text{Dsh}] &= -P_1 - P_5 - P_{10} - D_1\nabla^2[\text{Dsh}] \\
\partial_t[\text{Fz}] &= -P_1 - P_2 - P_6 - D_2\nabla_s^2[\text{Fz}] \\
\partial_t[\text{Pk}] &= -P_2^\dagger - P_3 - P_4^\dagger - D_3\nabla^2[\text{Pk}] \\
\partial_t[\text{Vang}] &= -P_3 - P_7^\dagger - P_9^\dagger - D_4\nabla_s^2[\text{Fz}]
\end{aligned}$$

$$\begin{aligned}
\partial_t[\text{DshFz}] &= P_1 - P_4 - P_8 - D_5\nabla_s^2[\text{DshFz}] + C \\
\partial_t[\text{FzPk}^\dagger] &= P_2 - P_5 - P_7 - D_6\nabla_s^2[\text{FzPk}^\dagger] \\
\partial_t[\text{PkVang}] &= P_3 - P_6^\dagger - P_8^\dagger - D_7\nabla_s^2[\text{PkVang}]
\end{aligned}$$

$$\begin{aligned}
\partial_t[\text{DshFzPk}^\dagger] &= P_4 + P_5 - P_9 - D_8\nabla_s^2[\text{DshFzPk}^\dagger] + C_1 \\
\partial_t[\text{FzPk}^\dagger\text{Vang}^\dagger] &= P_6 + P_7 - P_{10} - D_9\nabla_s^2[\text{FzPk}^\dagger\text{Vang}^\dagger]
\end{aligned}$$

$$\partial_t[\text{DshFzPk}^\dagger\text{Vang}^\dagger] = P_8 + P_9 + P_{10} - D_{10}\nabla_s^2[\text{DshFzPk}^\dagger\text{Vang}^\dagger] + C_2,$$

where

$$C = \sum_b R_{\text{ab}} \left([\text{Pk}]^{k_1} + [\text{Vang}]^{k_2} + [\text{PkVang}]^{k_3} \right) [\text{DshFz*}].$$

The term DshFz* indicates the summation of all DshFz complexes. This summation is over each edge b and the different values of R_{ab} represent the number of microtubules from edge b to edge a. The terms C_1 and C_2 are defined as

$$\begin{aligned}
C_1 &= C\frac{[\text{DshFzPk}^\dagger]}{[\text{DshFz*}]} \\
C_2 &= C\frac{[\text{DshFzPk}^\dagger\text{Vang}^\dagger]}{[\text{DshFz*}]},
\end{aligned}$$

and they help ensure that total DshFz amount is conserved. In the cell interior, the model includes:

$$\begin{aligned}
\partial_t[\text{Dsh}] &= -D_1\nabla^2[\text{Dsh}] \\
\partial_t[\text{Pk}] &= -D_3\nabla^2[\text{Pk}]
\end{aligned}$$

There are two types of diffusion in this model, and this is pictorially represented in Figure 2.5c. These variables are subject to the boundary conditions that there is no flow of proteins out of the cells.

2.3 Numerical Implementation

The discretization of the PDE was done using the Finite Volume method [70]. This discretization method was used because of the unique numerical and mathematical properties which

allow for the maintenance of numerical conservation of mass. More specifically, this method considers the flux through the edges of the discretization grid. By discretizing the PDE in this manner, there is a "telescoping" of fluxes so that the total flux, and hence mass, is conserved.

The grid that was used with this discretization method is shown in Figure 2.6. This particular grid was chosen for several reasons. First, the general shape of the cells on the wing are hexagons. So each cell has been modeled as a hexagon. Second, the hexagon is gridded into six portions, with each portion corresponding to a portion of the membrane and cytoplasm of the cell.

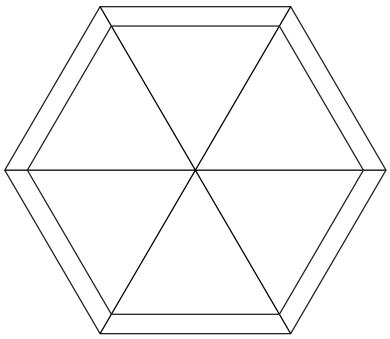


Figure 2.6: Grid of Single Cell

The numerical integration of the discretized PDE was conducted using an embedded Runge-Kutta method with Dormand-Prince coefficients [60]. An interesting aspect of the current version of the simulation code developed is that the code simulates an infinitely large array of cells. This is done by simulating an array of cells in which the neighbors of cells on the boundary are cells on the opposite boundary [5]. This is a valid technique because an infinite array of cells possesses translational invariance and as a result, each cell is identical.

The model presented has parameters which must be chosen, and the parameters were numerically identified using gradient descent on a square-loss function. The gradient was computed using a simple finite difference scheme [16]. In this case, the square loss was defined as:

$$\sum_k \|x_k - \hat{x}_k\|_2{}^2,$$

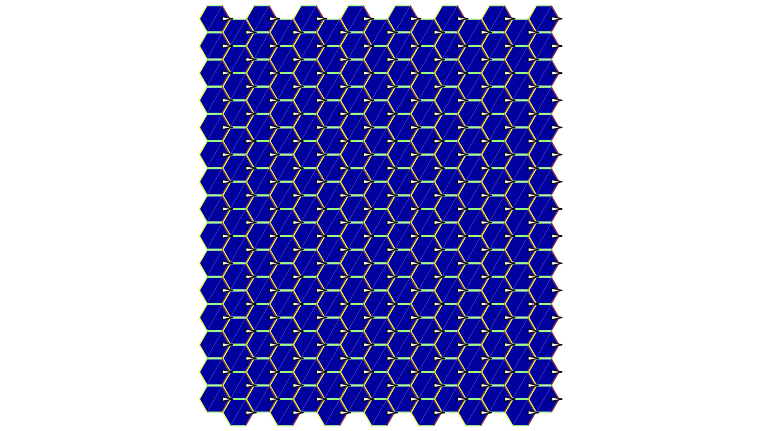
where x_k is a vector of the experimentally observed direction of hair on the k-th cell and $\hat{x}_k$ is the simulated direction of the hair on the k-th cell. The direction vector is a unit vector, and the direction is encoded by its angle. The experimentally observed hair direction of the Vang clone was used to fit the parameters. The Vang clone is an array in which a center square of cells are lacking the Vang protein. This distinguishes the parameter identification methods used in [5] which used a discontinuous loss function coupled with the Nelder-Mead simplex method to identify parameters. What is remarkable about this model is that it is able to replicate the experimental behavior of the system after fitting only the Vang clone; in comparison, the model in [5] was fit to four different clones before it was able to replicate the experimental behavior of the system.

2.4 Simulation Results

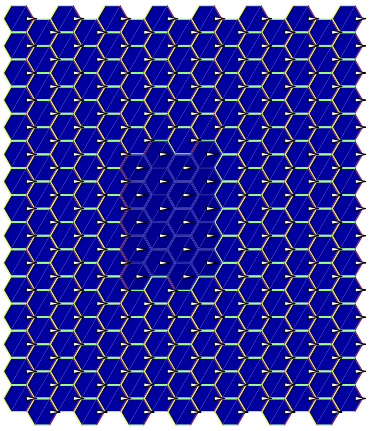
To see how well the model fits experimental results, we try several different scenarios. The first scenario is a wildtype simulation in which there are no genetic mutations on the fly wing. It is

shown in Figure 2.7a, and all the hairs point towards the distal edge of the fly wing which is to the right of the figure. The hairs are shown by the triangles, and the colors represent Dsh concentration which determines hair direction. Greater Dsh concentration is represented by brighter colors closer to red. The model behaves as expected for the wildtype case.

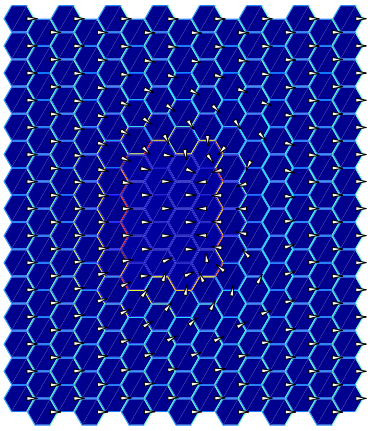
The next scenarios involve mutations on the fly wing in a square in the middle of the wing. These mutations respectively remove either Dsh, Fz, Pk, or Vang; corresponding simulation results are seen in Figures 2.7b, 2.7c, 2.7d, and 2.7e. These results match what is experimentally observed, and it matches the results of previous mathematical models [5]. As stated earlier, the benefit of this model is that it encodes new biology learned since [5].



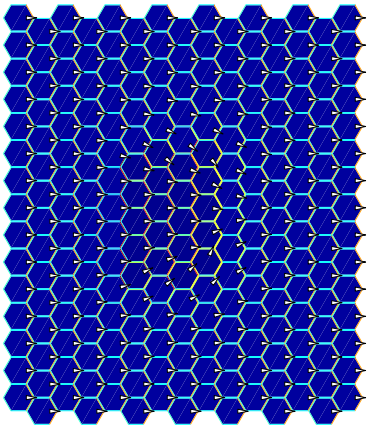
(a) Wildtype



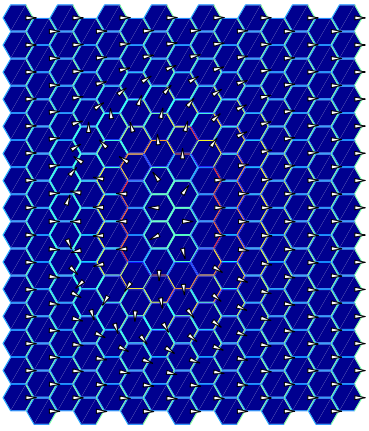
(b) Dsh Clone



(c) Fz Clone



(d) Pk Clone



(e) Vang Clone

Figure 2.7: Simulation of the model for an array of cells matches the experimentally observed behaviors. The wildtype results are cells with no mutations, wheras the other simulations are for situations in which a rectangular portion of cells in the middle of the array have various mutations which remove the corresponding protein from the mutated cell.

Chapter 3

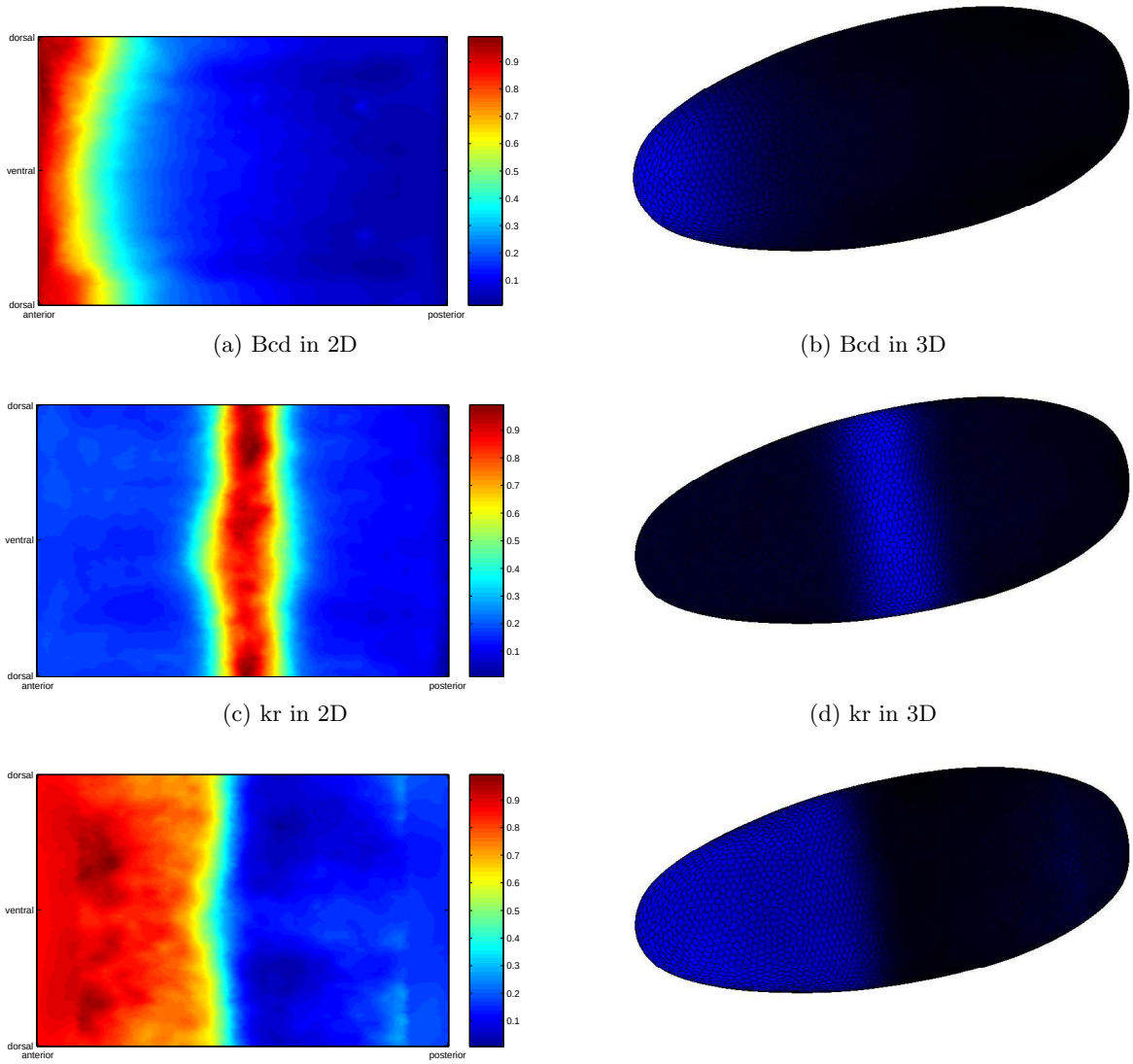
Modeling of Hunchback Pattern Formation in *Drosophila melanogaster*

Chapter 2 discussed parametric identification of a system in *Drosophila* which is related to human ailments. Similarly, this chapter provides a case study of performing system identification on another system in *Drosophila* by the use of parametric methods. The system of focus in this chapter is not immediately related to any human diseases, but this case study provides insights into the process of identifying mathematical models of biological systems.

Hunchback mRNA (hb) forms a pattern in the anterior domain of *Drosophila* during late stage 4 of embryogenesis. The formation of this pattern is well studied because of the relative ease of performing experiments. It is generally accepted that maternal Bicoid protein (Bcd) induces the transcription of hb in the anterior of the embryo [61, 127, 119]. However, the role of Krüppel (Kr) protein in hb regulation is less clear. Some suggest Kr represses hb to help establish the posterior boundary of anterior hb mRNA expression; others suggest Kr does not regulate hb [81, 111, 134, 92, 133]. The wild type patterns in late stage 4 of Bcd, Kr, and hb can be seen in Figure 3.1.

Identifying this network can be posed as a system identification problem solved with nonlinear regression [16] using experimental measurements collected by Berkeley Drosophila Transcription Network Project (BDTNP) [138, 72]. The general approach is to do nonlinear regression on a parametric, nonlinear partial differential equation (PDE) model with two spatial dimensions. This model incorporates *planar* diffusion, and it contains terms which aggregate the biological process of transcription which occur on time-scale that is faster than the entire process of transcription. The aggregate model terms can be mathematically justify using singular perturbation theory, and such models have been successfully used in biology [36, 210].

Actual implementation of this nonlinear regression requires the use certain numerical methods. The regression is posed as a least squares problem which is evaulated by solving a PDE. The PDE is discretized using the finite-volume method at the cellular level of resolution, and the forward Euler method is used to numerically integrate the discretized PDE. The regression itself is performed using a quasi-Newton method, and its results are interpreted in the biological context and compared to previous work on this system [157, 92].



(a) Bcd in 2D

(b) Bcd in 3D

(c) kr in 2D

(d) kr in 3D

(e) hb in 2D

(f) hb in 3D

Figure 3.1: In late stage 4, Bcd protein has a gradient-like pattern with the bulk of the concentration occurring at the anterior end of the embryo. Kr has a single-stripe pattern, and hb has a solid pattern with a sharp boundary in the center of the embryo.

3.1 Data Collection and Processing

The data used has been collected and processed by the BDTNP [138, 72]. Experimental measurements of protein and mRNA concentrations are taken by analyzing images of *Drosophila* embryos that are fluorescently stained to label expression patterns and nuclear DNA. The embryos are imaged using two-photon microscopy to produce three-dimensional image stacks. The image stacks are processed to produce a virtual embryo [138, 72], which can be thought of as an average of factor concentrations and an average of *Drosophila* geometry at the cellular level.

The resultant virtual embryo contains average cell geometry, protein and mRNA concentration in each cell (approximately 3000 cells in stage 4), and embryo geometry information. This includes the location and shape of each cell in the embryo, the neighbors of each cell, and the distance between neighbors. This data is particularly useful for the numerical discretization of the PDE via the finite volume method. The data in the virtual embryo is given for several different points in time—over stages 4 and 5 of the *Drosophila* embryo development—though only stage 4 is used in the modeling.

Figure 3.1 shows experimental data which has been processed using the approach of Sections 3.1.1. The virtual embryo has an irregular grid that is generated through a Voronoi tessellation based on the location of nuclei in the embryo [138]. Because of this irregular grid, exploiting symmetry to do noise-reduction filter is not straightforward; a filtering strategy which exploits symmetry in patterns is discussed in Section 3.1.1. Additionally, Kr protein data is not available, and so Kr mRNA data is used in its place. This is biologically reasonable to do because the mRNA and protein data for Krüppel has been experimentally observed to be similar with the exception of a slight time delay [72].

3.1.1 Filtering

A k-nearest-neighbors filter in which the filtered value of a cell is equal to the average of its k-nearest-neighboring cells is used. This is a two-dimensional filter which is applied to the surface of the embryo. The filter is applied to all the factors to alleviate the general noise present in the data, and it is a required compromise due to the limitations of the imaging techniques used by the BDTNP.

It has been experimentally observed that the patterns of Bcd, Kr, and hb are symmetric about the dorsal-ventral axis (see Figure 3.1). In other words, the left and right sides of the embryo are symmetric with respect to each other. This observation suggests the idea of mirroring the data to reduce noise. Specifically, a modified k-nearest-neighbors filter is used in which the filtered value of a cell is equal to the average of the k-nearest-neighboring cells on both sides of the embryo. That is, the data is averaged over all the cells that would be neighbors about a single cell in case of a 2D projection about the dorsal-ventral axis. To achieve better regression results and to avoid incorrect local minima, the values of hunchback mRNA are specially filtered: The value of the x-cells near the anterior and posterior edges is calculated as the y-cells that occur after the x-cells in the same plane.

3.2 Mathematical Model

The approach to this system identification problem requires having a parametric model, meaning an equation of motion with unidentified parameters. There are two potential models for the regulation of the posterior boundary of anterior hb domain.

It is generally accepted that high concentrations of Bcd protein lead to high concentrations of hb [61, 127, 119], and this is incorporated into both models. Additionally, diffusion and degradation of hb mRNA are widely accepted facets of behavior [61, 127], and so this is also incorporated into both models. The first model (3.1) takes only these effects into account, and it is given by the equation

$$\frac{\partial[hb]}{\partial t} = a_1 \frac{[Bcd]^{a_2}}{[Bcd]^{a_2} + a_3^{a_2}} + a_4 \nabla^2 [hb] - a_5 [hb]. \tag{3.1}$$

The first term in (3.1) is a fractal, Michaelis-Menten type reaction [172, 173, 53] which corresponds to promotion, and it is a common model in biological systems [210, 53, 92]. It has a sigmoidal shape—approaching 1 as $[Bcd] \to \infty$ and approaching 0 as $[Bcd] \to 0$. The second term is a diffusion term for hb[61, 127]. The third term is a degradation term, because hb has a finite lifetime [61, 127].

The second model takes into account the previously described possibility that Kr protein may repress Hb transcription [81, 111, 134, 92, 133]. The model (3.2) is given by

$$\frac{\partial[hb]}{\partial t} = b_1 \left(\frac{[Bcd]^{b_2}}{[Bcd]^{b_2} + b_3^{b_2}} \right) \left(1 - \frac{[Kr]^{b_4}}{[Kr]^{b_4} + b_5^{b_4}} \right) + b_6 \nabla^2 [hb] - b_7 [hb]. \tag{3.2}$$

The new term in (3.2) is a fractal, Michaelis-Menten type reaction [172, 173, 53] that corresponds to inhibition. In particular, this term approaches 0 as $[Kr] \to \infty$ and approaches 1 as $[Kr] \to 0$.

3.3 Nonlinear Regression

A nonlinear regression is used to identify the parameters in (3.1) and (3.2), and it can be framed as a nonlinear optimization:

$$\arg\min f(\vec{\gamma}) = \sum_{\vec{x} \in \mathcal{M}} \left\| [hb]_{\vec{\gamma}}(1, \vec{x}) - [\hat{hb}](1, \vec{x}) \right\|, \tag{3.3}$$

subject to:

$$[hb]_{\vec{\gamma}}(0, \vec{x}) = 0 \qquad \forall \vec{x} \in \mathcal{M}, (3.1) \text{ or } (3.2), \tag{3.4}$$

where $\mathcal{M}$ is the set of points corresponding to the cells in the virtual embryo, $\vec{\gamma}$ are the parameters of the model, $[hb]_{\vec{\gamma}}(t, \vec{x})$ is the solution to the PDE given in (3.1) or (3.2), and $\hat{m}(1, \vec{x})$ is the set of experimentally measured hb concentration values from the virtual embryo that has been processed as described in Section 3.1.

This nonlinear optimization problem is solved using a quasi-Newton method [27], and the common Broyden-Fletcher-Goldfarb-Shanno (BFGS) method [27] is used to approximate the Hessian matrix. A relatively large portion of the space of model coefficients ($\vec{\gamma}$) was used as the initial parameter values for the quasi-Newton method, in hopes that a good local minima can be discovered by covering a large portion of this space. Because evaluations of the function $f(\vec{\gamma})$ are fast in this case, the gradient can be computed using a naive finite differences method:

$$\left. \frac{\partial f}{\partial \gamma_i} \right|_{\vec{\gamma}^*} \approx \frac{f(\vec{\gamma}^*) - f(\vec{\gamma}^* + \delta \vec{e_i})}{\delta}, \tag{3.5}$$

where $\vec{e_i}$ is the standard basis vector in $\mathbb{R}^g$, where g is the number of parameters in the model.

Calculating the numerical approximation to the gradient requires evaluation of $f(\vec{a})$, and this requires solving the PDE given in (3.1) or in (3.2). The finite volume method [70] is used to spatially discretize the PDE. The discrete cells of the finite volume grid are the cells of the virtual embryo, and the cells are approximated as polygons that lie upon the surface of the embryo. The polygonal nature of the biological cells makes them an attractive option for the finite volume grid. The numerical discretization of the resulting ordinary differential equation (ODE) is performed using the forward Euler method because the dynamics of the PDE (and the resulting ODE, after discretization) is simple and does not display complicated behaviors such as chaos or stiffness.

3.4 Regression Results

The parameters returned—after performing the nonlinear regression procedure described in Section 3.3 using the experimental data shown in Figure 3.1—are shown substituted into the parametric PDEs. Performing the nonlinear regression for (3.1) gives:

$$\frac{\partial[hb]}{\partial t} = 0.91\frac{[Bcd]^{5.32}}{[Bcd]^{5.32} + 0.14^{5.32}} + 0.65\nabla^2[hb] - 0.067[hb]. \tag{3.6}$$

The hb concentration predicted by this model with these parameter values is shown in Figures 3.2c and 3.2d. Similarly, performing the nonlinear regression for (3.2) gives:

$$\frac{\partial[hb]}{\partial t} = 0.90\left(\frac{[Bcd]^{3.23}}{[Bcd]^{3.23} + 0.11^{3.23}}\right)\left(1 - \frac{[Kr]^{5.38}}{[Kr]^{5.38} + 0.67^{5.38}}\right) + 0.65\nabla^2[hb] - 0.0054[hb], \tag{3.7}$$

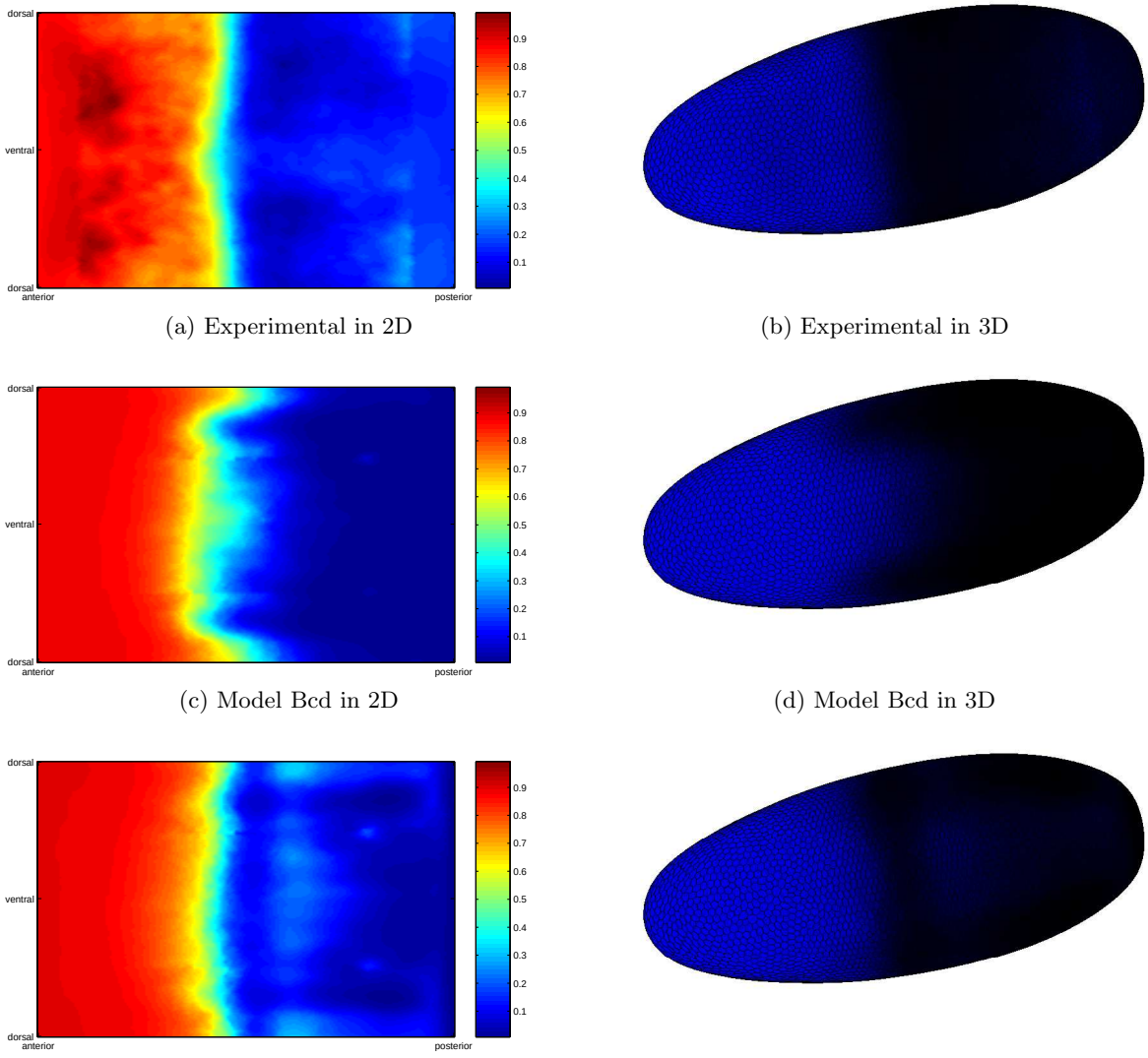
and the hb concentration predicted by this model is shown in Figures 3.2e and 3.2f.

3.4.1 Measures of Goodness-of-Fit

The fit of the model to the experimental data can be statistical measured in a variety of ways. The simplest such measure is mean-squared error (MSE), and here it corresponds to the value of $f(\vec{\gamma})$, where $\vec{\gamma}$ contains the parameters of the model and $f(\cdot)$ is given in (3.3). A lower value of MSE qualitatively means that the model better fits the experimental data, and the MSE values of each model are shown in Table 3.1.

The models can also be compared using the Akaike information criterion (AIC) and the Bayesian information criterion (BIC), which are statistical methods for model selection. They give a numerical result which qualitatively measures the difference between the data and the model [39] by considering both the predictive-power of the model and the number of parameters in the model. This makes AIC and BIC more expressive than simply comparing models by their MSE, and here lower values are better. If the model errors are assumed to be distributed as independent and identical Gaussians, then AIC and BIC can be more easily computed; these values are given in Table 3.1.

The model in (3.2) has lower MSE than the model in (3.1). It also qualitatively looks more like the experimental pattern. The model in in (3.2) also has a lower AIC and BIC value. Consequently, the system identification results suggest that the model in (3.2) is better than that given in (3.1).



(a) Experimental in 2D

(b) Experimental in 3D

(c) Model Bcd in 2D

(d) Model Bcd in 3D

(e) Model Bcd and Kr in 2D

(f) Model Bcd and Kr in 3D

Figure 3.2: We can compare the hb patterns generated by each model to both the experimental data and to themselves.

	Model in 3.1	Model in 3.2
MSE	74.22	41.69
AIC	-9.52×10^3	-1.30×10^4
BIC	-2.67×10^4	-3.02×10^4

Table 3.1: Measures of Goodness-of-Fit

3.4.2 Biological Interpretation

Note that there is no assertion regarding the biological validity of either model. Rather, it is only stated that the latter model produces a more accurate pattern than the former model. This supports the hypothesis that Kr is involved in hb pattern formation, and the fact that model in (3.1) does not generate the proper pattern (the boundary is too diffuse) supports the related hypothesis that Bcd alone is not sufficient to generate the hb pattern.

This mathematical modeling effort can be compared to those in [92, 96, 91]. Two-dimensional data along the surface of the embryo is used here, whereas the experimental data used in [92, 96, 49] consists of one-dimensional data along the flattened embryo. Additionally, the mathematical model in [92] is a static input-output model that only considers the effect of Bcd:

$$[hb] = hb_{max}\frac{[Bcd]^n}{[Bcd]^n + K^n}, \tag{3.8}$$

where $hb_{max}, K \in \mathbb{R}$ and n is an integer.

The model and experimental data in [92] match well, but this is misleading. Flattening the embryo and making experimental measurements in one-dimension effectively acts as an averaging filter, and it is akin to taking the two-dimensional data of Figure 3.1 and averaging along one of the dimensions to make it one-dimensional. Referring to Figure 3.1, it is not surprising that averaging hb and Bcd in this manner and then using a mathematical model with just Bcd would be enough to generate the hb pattern. Considering the full two-dimensional data, it is clear that Bcd alone is not enough to generate the hb pattern. Additionally, the results of the model presented here show that the addition of Kr to the model can generate the hb pattern.

The coefficient n is interpreted to be the number of Bcd molecules that cooperatively bind to generate the hb pattern [92]. In [92], the experimental data was matched to the model to compute a value of $n = 5$. In the results here, the model in (3.1) which also uses only Bcd has a value of $n = 5.32$. These values are somewhat high, and such situations are biologically infrequent. However, the model in (3.2) which uses both Bcd and Kr has a value of $n = 3.23$ which is more biologically frequent.

The models here can also be compared to that in [157]; this model also found that Kr is sufficient for the establishment of the posterior boundary of anterior hb mRNA expression. The models here have three primary differences with [157]. One difference is the form of activation used in [157] is not a fractal, Michaelis-Menten type reaction. Another difference is that the model and experimental data used in [157] only had anterior-posterior resolution, and no dorsal-ventral resolution like the BDTNP data. The results show that fitting a model with Kr to experimental data with anterior-posterior and dorsal-ventral resolution is still sufficient to reproduce the anterior hb mRNA expression. The third significant difference is that the model in [157] also had a slight repression of hb by Knirps protein, which is another protein involved in embryogenesis.

3.4.3 Future Works

Interpreting the comparison between the two models requires caution. Though one model was found to be better than the other, this is not sufficient evidence for a biological relationship. A set of biological experiments to delineate the differences between the two models is needed. The real benefit of the system identification procedure is that guides the biologist in conducting future experiments. Because the model in (3.2) produces better results than the model in (3.1), it is indeed reasonable to conduct future experiments that gauge the role of Kr. Because the model in (3.1) did not produce satisfactory results, the results suggest that the role of other factors, on hb pattern formation, should be considered. Additional system identification with new models using additional factors can be used to guide future, biological experiments.

Chapter 4

Locally-Linear System Identification

The parametric system identification procedures used in Chapters 2 and 3 require significant prior knowledge about the system being modeled. This prior knowledge is used to formulate a specific functional form of the equations with free tuning parameters that are picked using regression procedures in order to match experimental data. The advantage of this approach is that it provides models that are highly interpretable, with specific terms of the equations corresponding to specific phenomenon in the system. The disadvantage is that it *does* require extensive knowledge about the system.

An alternative form of system identification is the class of nonparametric techniques [135]; they are sometimes used in the robotics and control communities. The advantage of these techniques is that they are well-suited for situations in which there is little or no prior knowledge about the system being modeled. The disadvantage of nonparametric techniques is that they are statistically not as well behaved as parametric techniques. One purpose of this chapter is to provide a new statistical technique for nonparametric system identification [12, 14] that has better statistical performance than existing techniques.

The advantage of requiring limited prior knowledge is important in biological applications, because there is often little or unreliable knowledge about the large networks being modeled. This makes nonparametric methods particularly useful for network identification problems in biology. In such problems, we are interested in learning not only parameter values of the model, but also the interconnections between different parts of the network. In fact, for some of the drug multi-target selection tools discussed in the second part of this thesis, it is more important to know the interconnections than to know the parameter values.

From an informal perspective, systems described well by ordinary differential equations (ODEs) can be nonparametrically identified by matching trajectory data of the system with an ODE whose functional form is given by a series expansion. These techniques are useful for situations in which there is little *a priori* knowledge about the dynamics of the system. Suppose an engineering or biological system is described by the following general ODE:

$$\dot{x} = f(x) + g(x, u), \tag{4.1}$$

where $f(x)$ describes the nonlinear, zero-input dynamics and $g(x, u)$ describes the way the system depends on the inputs. If $g(x, u)$ is known by the designer, as is the case for many engineering systems, then the problem is simplified into identifying $\dot{x} = f(x)$. This framework is expressive enough to identify time-varying systems by choosing an augmented state vector $\tilde{x} = (x, t)$.

Local linearization techniques [135, 209, 22, 200, 109] are an important, special class of

nonparametric system identification. The prevalence of control tools which utilize piecewise-affine models or linear models makes this class of techniques important. The dynamics of the system are identified for points nearby to point (x_0, u_0) by linearization

$$\dot{x} = f(x) + g(x, u) \approx Ax + Bu + c, \tag{4.2}$$

for all x, u such that $\|x - x_0\|_2 + \|u - u_0\|_2 \leq h$; where $h > 0$ is small. A picture of the global dynamics can be gleaned by identifying a model for many different values of (x_0, u_0).

Nonparametric identification techniques involve solving regression problems of the form

$$Y = X\beta, \tag{4.3}$$

where $Y \in \mathbb{R}^n$ is a vector of noisy *response variables*, $X \in \mathbb{R}^{n \times p}$ is a matrix of noisy *predictor variables*, and $\beta \in \mathbb{R}^p$ is a vector of *regression coefficients*. The ordinary least squares (OLS) [41] solution is given by,

$$\beta_{\text{OLS}} = \arg\min_{\beta} \|Y - X\beta\|_2^2 \tag{4.4}$$

$$= (X'X)^{-1}X'Y, \tag{4.5}$$

where X' is the matrix transpose of X. Because these problems arise from using data local to (x_0, u_0), it is not uncommon for X to be a wide matrix, meaning $p > n$. This causes problems because the matrix inverse $(X'X)^{-1}$ does not exist. Furthermore, in some systems (e.g., gimbal lock [158]) the trajectories evolve on a manifold with dimension lower than that of the state-space. Once again, $(X'X)^{-1}$ does not exist.

From a practical standpoint, the two situations are equivalent and are known in the statistics literature as collinearity (or near-collinearity for ill-conditioned covariance matrix $X'X$) of the predictors. A variety of techniques has been developed to deal with this, including: partial least squares (PLS) [213, 73], principal components regression (PCR) [144, 73], ridge regression (RR) [105, 73], Moore-Penrose pseudoinverse (MP) [33, 120], and elastic net (EN) [223]. RR and EN can handle the collinearity aspect but do not explicitly recognize that the predictors lie on a lower dimension hyperplane. On the other hand, PLS, PCR, and MP consider the manifold structure but can have inconsistent performance. There is hope that this inconsistency can be improved with new estimators. Note that the approach in [33] applies to a more general setup in which the data lies on a lower dimensional manifold which might not be a hyperplane.

Of techniques that explicitly consider manifold structure (i.e., PLS, PCR, MP), none of these can take advantage of any sparsity in the system. Sparsity of an ODE using the local linearization in (4.2) is defined as

$$\sum_{i=1}^{p} 1(A_{ij} \neq 0) \leq c_0(p), \tag{4.6}$$

where $c_0(p)$ is an increasing function of p. This equation means that there is an upper bound on the number of non-zero entries in the matrix A. Strictly speaking, this condition can be somewhat relaxed [32]. Dynamical systems found in engineering problems are typically sparse because they are an interconnection of preexisting, engineered components. Biological systems (especially regulatory networks) are often sparse as well because the organization of such networks is known to be hierarchical. It is important to develop identification techniques that take advantage of this special structure.

This chapter begins with a formal problem statement of local linearization identification and an explanation of how this is related to (4.3). A new statistical technique is briefly introduced, and it is studied in more depth in the next chapter. The regression problem (4.3) is interpreted as a regression on an embedded submanifold, which is the setup of [33]. Lastly, the overall system identification procedure is described.

There are two advantages to this interpretation: First, it clarifies the meaning of the regression coefficients, which is a common source of confusion in the literature. Secondly, it allows the development of a new technique which can simultaneously exploit sparsity and collinearity in order to improve the performance of estimating the regression coefficients.

4.1 Problem Statement

Consider the dynamical system given in (4.1) with trajectory given by $x_e(t)$ and initial condition $x_e(0)$, where e is an index that enumerates over a set of different trajectories (alternatively, experiments). Furthermore, suppose that noisy measurements of the states and inputs are made

$$\begin{aligned} x[t,e] &= x_e(t) + z \\ u[t,e] &= u_e(t) + \eta, \end{aligned} \tag{4.7}$$

for all $t = t_1, \ldots, t_T$ and for all $e = 1, \ldots, E$; where the system is measured at T time-points for E experiments. If the system identification is done in real-time, then $E = 1$; however, if a system is being experimentally characterized, then typically $E > 1$. Also, z, η are vectors of noise with zero mean and finite variance σ^2, ν^2.

The problem of interest is: Given a point (x_0, u_0), for the system (4.1) with measured data (4.7), determine the local linearization (4.2) of the system.

4.1.1 Relation to Linear Regression

For now, pretend that there is access to non-noisy measurements of the relevant variables: $x_e(t), u_e(t), \dot{x}_e(t)$. The situation with noisy measurements will be considered later. Define an indicator function by

$$K_h(e,t) = 1(\|x_e(t) - x_0\|_2^{\,2} + \|u_e(t) - u_0\|_2^{\,2} \leq h^2). \tag{4.8}$$

This indicator function is zero if $x_e(t), u_e(t)$ are far from x_0, u_0 and is one otherwise. This can be generalized by choosing $K_h(e,t) = 1/h^p K(x_e(t)/h, u_e(t)/h)$, where $K(\cdot,\cdot)$ is a general window (e.g. unit-hold, bicubic, or Epanechnikov kernels). Note that windowing the data and doing a linear regression is a special case of nonparametric, local polynomial regression [169]. In the present case, the regression model is given by

$$W^{1/2} \begin{bmatrix} \dot{x_1}(t_1)' \\ \vdots \\ \dot{x_E}(t_T)' \end{bmatrix}' = W^{1/2} A \begin{bmatrix} x_1(t_1)' \\ \vdots \\ x_E(t_T)' \end{bmatrix}' + W^{1/2} B \begin{bmatrix} u_1(t_1)' \\ \vdots \\ u_E(t_T)' \end{bmatrix}' + W^{1/2} c \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}, \tag{4.9}$$

where $W = \mathrm{diag}(K_{h_x,h_u}(1,1), \ldots, K_{h_x,h_u}(E,T))$. Solving for A, B, and c effectively identifies the local linearization.

This regression model can be rewritten in matrix form as

$$W^{1/2} Y_m = W^{1/2} X_m \Xi, \tag{4.10}$$

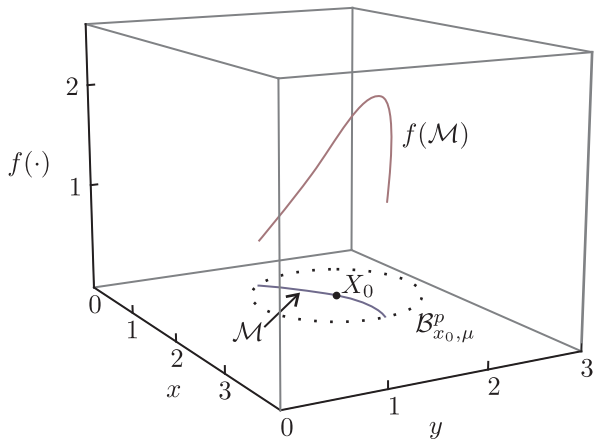


Figure 4.1: In a small ball $\mathcal{B}^p_{x_0,\mu}$ about the point X_0, the predictors form the manifold $\mathcal{M}$. The response variable is a function of the predictors, which lie on this manifold. Here the manifold is of dimension $d = 1$, and the number of predictors is $p = 2$.

where

$$Y'_m = \begin{bmatrix} \dot{x}_1(t_1) & \dots & \dot{x}_E(t_T) \end{bmatrix} \tag{4.11}$$

$$X'_m = \begin{bmatrix} x_1(t_1) & \dots & x_E(t_T) \\ u_1(t_1) & \dots & u_E(t_T) \\ 1 & \dots & 1 \end{bmatrix} \tag{4.12}$$

$$\Xi' = \begin{bmatrix} A & B & c \end{bmatrix}. \tag{4.13}$$

Solving for Ξ by considering one column of (4.10) at a time is equivalent to (4.3).

4.2 Manifold Interpretation

To solve the regression problem (4.3), the problem is interpreted as a regression on a manifold which has a locally Euclidean metric; this is the setup of [33]. This sort of situation occurs if there are constraints on the dynamics of the system which cause it to evolve on a lower-dimensional manifold. It also occurs if the trajectory is undersampled with respect to the dimensionality of the system. Suppose there is locally a manifold $\mathcal{M} \subseteq \mathbb{R}^p$, where $\dim(\mathcal{M}) = d$. Also, suppose there is a function $f(\cdot) : \mathcal{M} \to \mathbb{R}$. This situation is seen in Figure 4.1, and only the data points are known in this regression problem; the manifold $\mathcal{M}$ and the function $f(\mathcal{M})$ are unknown.

The gradient (and Jacobian) of a function on a manifold does not always exist, but there is a generalization of gradients to functions on manifolds. This generalization is known as the exterior derivative of a function, and the interested reader can refer to [129] for a formal definition. The exterior derivative of a function is sometimes referred to as a differential or as the exterior derivative of a 0-form. It is enough to intuitively know that an exterior derivative can be thought of as a sum of directional derivatives along the directions of the manifold, and it is a well-defined quantity because directional derivatives are well-defined on manifolds. The exterior derivative at a point p lies in the cotangent space at p, which is the subspace that is parallel to the manifold at p; more formally, the tangent space is parallel to the manifold at p, and the cotangent space is

isomorphic to the tangent space via the Euclidean metric because we are dealing with embedded submanifolds in an ambient space of $\mathbb{R}^p$.

To understand why gradients do not always exist, consider the following partial derivative

$$\frac{\partial f}{\partial x_1} = \lim_{h \to 0} \frac{f(x_1 + h, \ldots, x_p) - f(x_1, \ldots x_p)}{h}. \tag{4.14}$$

In general, even though $(x_1, \ldots, x_p) \in \mathcal{M}$ holds, it can happen that $(x_1 + h, \ldots, x_p) \notin \mathcal{M}$ for arbitrarily small h. Thus, the partial derivative is ill-defined because $f(\cdot)$ is not defined for values not in $\mathcal{M}$.

The manifold interpretation leads to interpreting the regression coefficients as the exterior derivative of a function. This is important because it means that the regression coefficients only give information about the directions that the manifold samples, and these directions are formally known as the tangent space of the manifold. This has important implications for control because it says that nonparametric system identification techniques that regularize the regression problem are unable to learn all the dynamical modes of the system. This is problematic if, for instance, the unlearned modes are unstable.

The manifold interpretation of the regression problem leads to a new form of regularization for the regression problem, and this is the focus of Chapter 5. A very brief summary is given here. The cotangent space is learned by forming a local covariance matrix $X'WX$ and then computing its eigenvalue decomposition. Deviations from the cotangent space are penalized by projecting the regression vector perpendicular to the cotangent space and having a corresponding penalty. Specifically, the nonparametric exterior derivative estimator (NEDE) is given as

$$\hat{\beta}_{\text{NEDE}} = \arg\min_{\beta} \|W^{1/2}(Y - X\beta)\|_2^2 + \lambda \|\Pi\beta\|_2^2. \tag{4.15}$$

The first term is an OLS setup (4.4), the second term is known as a Tikhonov-type regularization term, and the W matrix makes this a weighted least-squares. The regularizing matrix Π is a projection matrix that projects orthogonal to the cotangent space.

This estimator can be suitably modified to exploit sparsity and to handle the "large p, small n" case. The latter case is when p is on the order of or larger than n; or, asymptotically speaking: $p/n \neq o(1)$. It is known [143, 115, 114], that in this setting the covariance matrix $X'X$ and cross-covariance matrix $X'Y$ are ill-behaved. Specifically, the eigenvalues and range space of eigenvectors do not converge to the correct values. The exact details are given in Chapter 5.

Details on our entire class of estimators are also given in Chapter 5. For reference, we briefly explain the acronyms used to refer our estimators. The manifold regularization used in all of our estimators is indicated by the stem (-EDE-). Nonparametric forms of our estimators are indicated with a prefix of (N-), and a lack of this prefix indicates a linear, parametric estimator. The sparsity-inducing regularization is indicated by the characters (-AL-), and the regularization for "large p, small n" case is denoted by the suffix (-P).

For the discussion here, it is useful to note that Theorem 5.3.2 shows that the NEDE estimator is consistent:

$$\|\hat{\beta}_{\text{NEDE}} - \beta\|_2 = O_p(n^{-1/(d+4)}).$$

If the relationship between the predictors and response variables is linear, then W can be chosen to be the identity matrix and the rate of convergence is in fact $O_p(n^{-1/4})$ as shown by Theorem 5.4.3.

4.3 Mass-Spring Example

Consider the mass-spring system shown in Figure 4.2, and consider the positions of the two masses:

$$\xi(t) = \begin{bmatrix} \xi_1(t) & \xi_2(t) \end{bmatrix}. \tag{4.16}$$

This system is interesting because the motion lies in either two-dimensions or one-dimension depending on the stiffness of the springs. If spring 2 (the spring in between the two masses) is very stiff, then the spring is basically a rigid rod. In this case, the two masses remain a fixed distance L from each other, meaning that $\xi_2(t) = \xi_1(t) + L$ and the motion is one-dimensional. If the spring in between the two masses is weak, then the two masses vibrate at different rates and the motion is two-dimensional.

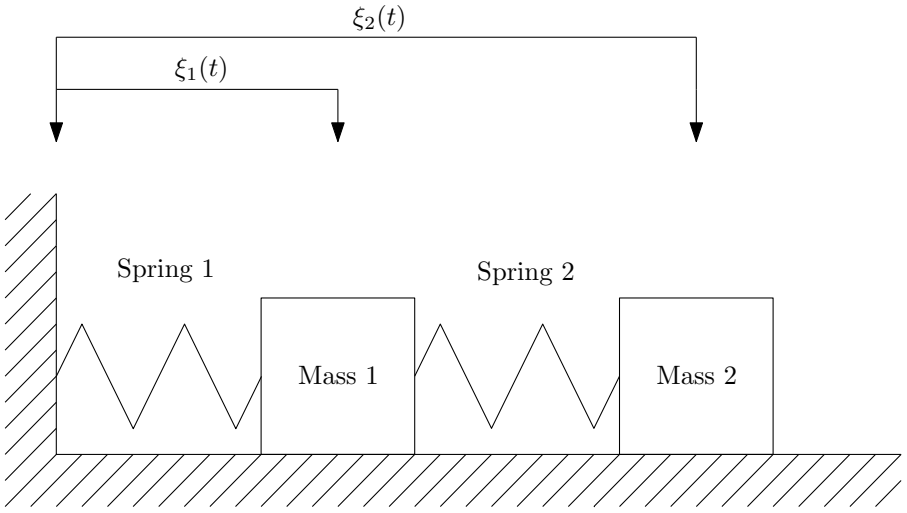


Figure 4.2: A mass-spring system with two masses and two springs is shown, where the masses slide on the ground and spring 1 is attached to the wall. We can make measurements of the position and acceleration of the masses and use these measurements to compute the stiffness of the springs. In general, the two masses oscillate at different rates related to the stiffness of each spring; this behavior is two-dimensional. If spring 2 is much stiffer than spring 1, the second spring acts like a rigid rod; this behavior is one-dimensional.

An interesting problem is to determine the stiffness of the springs based on measurements of the positions and accelerations of the masses. This can be framed as a regression problem. Suppose the masses have initial conditions:

$$\begin{bmatrix} \xi_1(0^-) \\ \xi_2(0^-) \end{bmatrix} = \begin{bmatrix} L \\ 2L \end{bmatrix} \qquad \begin{bmatrix} \frac{d\xi_1}{dt}(0^-) \\ \frac{d\xi_2}{dt}(0^-) \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}.$$

The equations describing the motions of the two masses can be written using Newton's law and are given by:

$$\begin{aligned} \frac{d^2\xi_1}{dt^2} &= -\frac{k_1}{m}(\xi_1 - L) + \frac{k_2}{m}(\xi_2 - \xi_1 - L) \\ \frac{d^2\xi_2}{dt^2} &= -\frac{k_2}{m}(\xi_2 - \xi_1 - L), \end{aligned} \tag{4.17}$$

where k_1, k_2 describe the stiffness of the two springs, L is the length of the unstretched springs, and m is the mass of each block (they have equal masses). If the second spring is very stiff ($k_2 \gg k_1$), then the equations of motion are approximately given by:

$$\begin{aligned} \frac{d^2\xi_1}{dt^2} &= -\frac{k_1}{2m}(\xi_1 - L) \\ \xi_2(t) &= \xi_1(t) + L. \end{aligned} \tag{4.18}$$

The idea is to make measurements of the relevant variables, and then do a regression to determine the coefficients. Assume that $n = 1250$ measurements are made:

$$\begin{aligned} y_1(t) &= \frac{d^2\xi_1}{dt^2}(t) + \mathcal{N}(0, 0.5^2) \\ y_2(t) &= \frac{d^2\xi_2}{dt^2}(t) + \mathcal{N}(0, 0.5^2) \\ x_1(t) &= \xi_1(t) + \mathcal{N}(0, 0.1^2) \\ x_2(t) &= \xi_2(t) + \mathcal{N}(0, 0.1^2) \end{aligned}$$

at evenly spaced values of $0 \leq t \leq 25$. This is a realistic example because accelerometers can directly measure acceleration, and position is quite easy to measure.

Regressing the predictors x_1, x_2 onto the response variables y_1, y_2 will indirectly give the stiffness of the springs. If the system is well described by (4.17), then the linear model will be given by

$$\begin{aligned} \begin{bmatrix} \frac{d^2\xi_1}{dt^2}(t) \\ \frac{d^2\xi_2}{dt^2}(t) \end{bmatrix} &= B_1 \begin{bmatrix} 1 \\ \xi_1(t) \\ \xi_2(t) \end{bmatrix} \\ B_1 &= \begin{bmatrix} (k_1 - k_2)L/m & -(k_1 + k_2)/m & 0 \\ k_2 L/m & k_2/m & -k_2/m \end{bmatrix}. \end{aligned} \tag{4.19}$$

On the other hand, if the system is well described by (4.18), then the linear model will be given by

$$\begin{aligned} \begin{bmatrix} \frac{d^2\xi_1}{dt^2}(t) \\ \frac{d^2\xi_2}{dt^2}(t) \end{bmatrix} &= B_2 \begin{bmatrix} 1 \\ \xi_1(t) \\ \xi_2(t) \end{bmatrix} \\ B_2 &= \begin{bmatrix} \frac{3}{2\sqrt{2}m}k_1 L & -\frac{1}{2\sqrt{2}m}k_1 & -\frac{1}{2\sqrt{2}m}k_1 \\ \frac{3}{2\sqrt{2}m}k_1 L & -\frac{1}{2\sqrt{2}m}k_1 & -\frac{1}{2\sqrt{2}m}k_1 \end{bmatrix}. \end{aligned} \tag{4.20}$$

The coefficients take this form because when the model is (4.18), only the exterior derivative, which takes the above form, can be computed.

Regressions for models (4.17) and (4.18) are done, and then the estimated regression coefficients are compared to (4.19) and (4.20). For the numerical experiment, the values of $k_1 = 0.4, k_2 = 0.25, L = 1, m = 1$ were chosen for model (4.17), and the values of $k_1 = 0.4, k_2 = 10000, L = 1, m = 1$ were chosen for model (4.18). The different estimators were compared using normalized estimation error (nMSE) defined as

$$nMSE(\hat{B}, B) = \|\hat{B} - B\|_F{}^2 / \|B\|_F{}^2. \tag{4.21}$$

The results over 50 replications of this numerical experiment are tabulated in Table 4.1.

	Model (4.17), with $k_1 = 0.4, k_2 = 0.25, L = 1, m = 1$			
	$nMSE(\hat{B}, B_1)$		$nMSE(\hat{B}, B_2)$	
OLS/MP	0.096	(0.011)	1.422	(0.130)
RR	0.091	(0.009)	1.286	(0.115)
EN	0.091	(0.009)	1.286	(0.115)
PLS	0.096	(0.011)	1.422	(0.130)
PCR	0.096	(0.011)	1.422	(0.130)
EDE	0.091	(0.009)	1.286	(0.115)
ALEDE	0.091	(0.009)	1.286	(0.115)
EDEP	0.091	(0.009)	1.286	(0.115)
ALEDEP	0.091	(0.009)	1.286	(0.115)
	Model (4.18), with $k_1 = 0.4, k_2 = 10000, L = 1, m = 1$			
	$nMSE(\hat{B}, B_1)$		$nMSE(\hat{B}, B_2)$	
OLS/MP	1.000	(0.000)	0.231	(0.162)
RR	1.000	(0.000)	0.118	(0.058)
EN	1.000	(0.000)	0.135	(0.074)
PLS	1.000	(0.000)	0.160	(0.167)
PCR	1.000	(0.000)	0.162	(0.166)
EDE	1.000	(0.000)	0.112	(0.060)
ALEDE	1.000	(0.000)	0.129	(0.077)
EDEP	1.000	(0.000)	0.111	(0.060)
ALEDEP	1.000	(0.000)	0.128	(0.078)

Table 4.1: Averages and standard deviations over 50 replications of normalized mean-squared estimation error (nMSE) for models (4.17) and (4.18) when compared to the regression coefficients B_1 (4.19) and B_2 (4.20).

When the model is (4.17), the predictors lie on a two-dimensional space and the computed regression coefficients all match B_1 (4.19) better than they match B_2 (4.20). Because the system is two-dimensional, each estimator has roughly the same performance.

When the model is (4.18), the predictors lie on a one-dimensional space and the computed regression coefficients all match B_2 (4.20) better than they match B_1 (4.19). The best performance is achieved by the EDE and EDEP estimators, though the other estimators work well. This is not surprising because the predictors are one-dimensional and the new estimators can improve estimation in such scenarios. EDEP does improve estimation very slightly in this situation, and it was one of the few situations where thresholding the covariance matrices provided an actual improvement in estimation error. This is likely due to the fact that there are $n = 1250$ data points and so the bootstrap is more accurate in this case.

4.4 System Identification Procedure

Our system identification procedure has two steps. The first step is a presmoothing step, and the second step is the application of the estimators in Section 4.2 to the regression problem defined in Section 4.1.1. Because the estimators are the focus of the next chapter, in this section the focus will be on the first step. For simplicity only an informal sketch of the results will be given on the presmoothing.

Because all variables are measured with noise, a situation known as errors-in-variables, the estimates must be carefully made. This is challenging because if it is not accounted for, then the estimates will be incorrect even in the limit of infinite data points (i.e. an uncorrected estimator is not consistent). Deregularization procedures, total least squares, and deconvolution-type approaches are common approaches for dealing with errors-in-variables [175, 207, 122].

The general system identification problem has special structure via time-dependence. The data can be presmoothed in time, and then this presmoothened data can be used with the second step of our system identification procedure. This procedure will produce consistent estimates in the errors-in-variables scenario. Presmoothing the data in time is actually a special case of instrumental variables [156], which is another technique used to deal with errors-in-variables.

Presmoothing in time can be implemented using filters. Electrical engineers are often partial to FIR and IIR filters [161], particularly the Savitsky-Golay filter [174] or a general low-pass filter. These filters can have poor transient (boundary) behavior, and local polynomial regression [169] has better transient behavior. The use of Kalman filters [124] is not possible, because the system dynamics are not known.

My position on the use of the various filters is as follows: When the identification needs to be real-time, the use of FIR or IIR filters is recommended because of the computational simplicity afforded by the FFT algorithm. When the identification is done offline, local polynomial regression is preferred for its better transient behavior.

The local polynomial regression procedure [169] can be used on the data $(x[t,e], u[t,e])$ to produce presmoothened data $(\hat{x}_e(t), \hat{u}_e(t))$. This procedure fits a polynomial to local windows of data and uses the fitted polynomial to estimate the trajectory values, input values, and the time-derivative of the trajectory. There is choice in which order of polynomial r to choose: It should be neither be too large, to prevent over fitting, nor too low, to prevent under fitting. Choosing orders $r = 2$ or $r = 3$ (i.e. quadratic or cubic) works well in practice.

The statistical justification for presmoothing the data is to deal with errors-in-variables, and it is actually a special case of using time t as an instrumental variable. To intuitively understand why leads to consistency, note that under regularity conditions and fixed p, [169] it uniformly holds that:

$$\|\hat{x}_e(t) - x_e(t)\|_2^2 = O_p(n^{-(2r+2)/(2r+3)}) \tag{4.22}$$

$$\|\hat{u}_e(t) - u_e(t)\|_2^2 = O_p(n^{-(2r+2)/(2r+3)}) \tag{4.23}$$

$$\|\hat{\dot{x}}_e(t) - \dot{x}_e(t)\|_2^2 = O_p(n^{-(2r+2)/(2r+5)}) \tag{4.24}$$

Applying Slutsky's theorem [34],

$$\|X'WX - \hat{X}'\hat{W}\hat{X}\|_2^2 = O_p(n^{-(2r+2)/(2r+3)}) \tag{4.25}$$

$$\|X'WY - \hat{X}'\hat{W}\hat{Y}\|_2^2 = O_p(n^{-(2r+2)/(2r+5)}). \tag{4.26}$$

Note that $\tilde{\beta}_{NEDE}$, which are the estimated coefficients of the NEDE estimator using the noisy data, has an explicit form

$$\tilde{\beta}_{NEDE} = (\hat{X}'\hat{W}\hat{X} + \lambda\hat{N}\hat{N}')^{-1}\hat{X}'\hat{W}\hat{Y}, \tag{4.27}$$

and let $\hat{\beta}_{NEDE}$ be the estimator which uses the non-noisy values X, W, and Y. Another application of Slutsky's Theorem gives

$$\|\hat{\beta}_{NEDE} - \tilde{\beta}_{NEDE}\|_2 = O_p(n^{-(2r+2)/(2r+5)}). \tag{4.28}$$

Using the approach and results in [146, 222], it can be shown that that

$$\|\hat{\beta}_{NEDE} - \hat{\beta}_{NALEDE}\|_2 = O_p(n^{-1/(d+4)}) \tag{4.29}$$

which implies that

$$\|\hat{\beta}_{NEDE} - \beta\|_2 = O_p(n^{-(2r+2)/(2r+5)} + n^{-1/(d+4)}) = O_p(n^{-1/(d+4)}) \tag{4.30}$$

$$\|\hat{\beta}_{NALEDE} - \beta\|_2 = O_p(n^{-(2r+2)/(2r+5)} + n^{-1/(d+4)}) = O_p(n^{-1/(d+4)}). \tag{4.31}$$

This result shows that presmoothing and then using the NEDE or NALEDE estimators leads to consistent estimates of the parameters. It also shows that the asymptotic rate of convergence is unchanged by having to first presmooth the data. Of course, the finite sample behavior will have increased bias and variance when compared to estimates without noisy measurements.

4.5 Applications to Engineering Systems

The interpretation of the regression coefficients as an exterior derivative is important, and it is common source of confusion. The PLS and PCR techniques also compute the exterior derivative of the function, but the regression coefficients are often interpreted as a gradient or a Jacobian. The distinction is important, because the exterior derivative contains derivative information in only the directions of the tangent space of the manifold. If the true function exists on more than just the manifold formed by the predictors X (this case can occur if we do not have sufficient measurements), then the regression coefficients do not contain information about any directions not spanned by the tangent space. This also explains why the rate of convergence for the estimators depends on the dimension of the manifold on which the predictors lie.

The statistical system identification procedure developed in this chapter can be used to identify engineering systems (in addition to biological systems), and the statistical estimators developed in Chapter 5 can identify more accurate models than possible with existing estimators. The mass-spring system provided in this chapter is an example of this. The benefit of this example is that it is a case in which the manifold structure comes about from real physical properties of the system, and so the manifold structure can be intuitively visualized. A second example which is provided in Appendix B uses real data taken from a quadrotor helicopter [106, 107]. This example suggests that our tools can identify better models of real engineering systems.

Because the identified system contains exterior derivative information and not gradient information, the control engineer must be careful in designing and implementing a controller. This approach and interpretation provides a link between the system identification and geometric control [170] based on either differential geometry or exterior differential systems. The interplay between these two areas is an interesting future direction of this work.

Chapter 5

Regression on Manifolds: Estimation of the Exterior Derivative

Lasso-type regularization [199, 120, 66, 222] and the Dantzig selector [42] are popular techniques for variable selection which have gained favor over more classical techniques based on Akaike or Bayesian Information Criterion or Mallowes-Cp. However, it is well-known that lasso-type regularization and the Dantzig selector have trouble handling collinearity. This prompted work on extensions [223], though further improvements are possible.

Collinearity is equivalent to having predictors which lie on lower-dimensional manifolds, and it suggests that manifold learning may be used to regularize ill-posed regression problems. This geometric insight has not been fully understood and exploited, though several techniques [120, 223, 28, 33] have explored this area. It is not strictly necessary to learn the manifold for prediction [33], but it has been shown that doing so can improve estimation in a min-max sense [154].

This chapter considers variable selection and coefficient estimation when the predictors lie on a manifold whose dimension is lower than that of the ambient space [14]. The focus is on the case where this manifold is nonlinear. Handling a global, linear manifold is a simple extension, and this extension is briefly discussed. Predicting function values on a nonlinear manifold was first studied in [33], but the authors did not study estimation of derivatives. The discussion does not consider global estimation and variable selection on nonlinear manifolds because [86] showed that learning the manifold globally is either poorly defined or computationally expensive.

Collinearities are interpreted in the language of manifolds, and this provides the two messages of this chapter. This interpretation suggests a new regularization for regression in the presence of collinearities or near-collinearities. This interpretation also gives a novel interpretation of regression coefficients when there is significant collinearity of the predictors.

On a statistical level, the idea is to learn the manifold formed by the predictors and then use this to regularize the regression problem. This regularization is derived from concepts in manifold geometry on the exterior derivative [148, 129]. The main idea is to learn the manifold either locally (in the case of a local, nonlinear manifold) or globally (in the case of a global, linear manifold). The regression is then posed as a least-squares problem with an additional term which penalizes for the regression vector lying in directions perpendicular to the manifold.

This manifold interpretation provides a new interpretation of the regression coefficients. The gradient describes how the function changes as each predictor is changed independently of other predictors, but this is impossible when there is collinearity of the predictors: The gradient

does not exist [189]. The exterior derivative of a function [148, 129] tells how the function value changes as a predictor and its collinear terms are simultaneously changed, and it has applications in control engineering [170], physics [148], and mathematics [189]. The regression coefficients in the presence of collinearities are estimates of the exterior derivative of the function, and this concept is particularly useful in high-dimensional system identification for biological and control engineering systems [12, 17]. This interpretation is discussed in more detail in Chapter 4.

The exterior derivative interpretation is important because it says that the regression coefficients only give derivative information in directions parallel to the manifold and give *no* derivative information in directions perpendicular to the manifold. If regression coefficients are computed for only the directions parallel to the manifold, then the regression coefficients are unique and they are uniquely given by the exterior derivative.

Similar interpretations are found in the literature [144, 73, 77, 120, 218, 78], but the interpretation give here is novel because of two main reasons. This is the first time the geometry has been interpreted in the manifold context, and this is important for many application domains. This interpretation makes it easy to show existing regularization techniques are really estimates of the exterior derivative.

This has important implications for the interpretation of estimates calculated by existing techniques. In this chapter, a link is established between the new estimators presented here and to both principal components regression (PCR) [144, 73] and ridge regression (RR) [105, 120]. Links between PCR, RR, and other regularization techniques can be shown [104, 87, 98, 77].

Past techniques have recognized the importance of geometric structure in doing regression. Ordinary least squares (OLS) performs under collinearity of predictors, leading to the development of more tools. RR [105, 120] provides shrinkage of the OLS estimator, and elastic net (EN) [223] combines RR with lasso-type regularization. The Moore-Penrose pseudoinverse (MP) [120] explicitly considers the manifold, but it is known to be highly discontinuous in the presence of near-collinearity caused by errors-in-variables. Though, MP works well in the case of a singular design matrix because the null-space of the covariance matrix is exactly known in that particular case. PCR [144, 73] and partial least squares (PLS) [213, 73, 6] are other popular approaches which explicitly consider geometric structure.

Existing techniques assume a global, linear manifold, but they admit simple extensions for local, nonlinear manifolds. The regression can be posed as a weighted, linear regression problem in which the weights are chosen to localize the problem [169]. Variable selection in the nonlinear function context was studied by RODEO [126], but this tool requires a heuristic form of regularization which does not explicitly consider the manifold.

Producing sparse estimates—which can simultaneously improve estimation and do variable selection—is difficult when the predictors lie on a manifold. Lasso-type regularization, the Dantzig selector, and the RODEO perform unsatisfactorily in such situations. The EN produces sparse estimates, but it does not explicitly consider manifold structure; however, EN does provide shrinkage which is a geometric concept.

The estimators presented must learn the manifold in order to regularize the regression. As part of the manifold learning, it is important to estimate the dimension of the manifold using either dimensionality estimators [48, 131, 97] or resampling-based approaches. Though cross-validation tends to perform poorly when used with PCR [123, 152], numerical examples in Section 5.6 show that bootstrapping works well with the estimators presented in this chapter. Also, it is worth noting that these estimators only work for manifolds with integer dimensions.

Learning a local, nonlinear manifold differs from learning a global, linear manifold. In the

local case, kernels localize the estimators which (a) learn the manifold and (b) do the nonparametric regression. For simplicity, the same bandwidth is used for both; though, separate bandwidths could be used. The linear case has faster asymptotic convergence because no localization is done. The linear case considered has errors-in-variables where the noise variance is identifiable [123, 114], and this distinguishes the setup from standard ones [199, 120, 66, 223, 222].

5.1 Problem Setup

Consider prediction and coefficient estimation of a function which lies on a local, nonlinear manifold. For local regression, proving results on the pointwise-convergence of the new estimators only requires assumptions which hold locally. The number of predictors is kept fixed, and the the dimension of the manifold can vary at different points in the predictor space. In this setup, estimation cannot be done at the points where the manifold is discontinuous.

Suppose that derivative information of the function is to be estimated about the point $X_0 \in \mathbb{R}^p$, where there are p predictors. The point X_0 is the choice of the user, and varying this allows computing the derivative information at different points. Because the estimation is local, it is useful to select small portions of the predictor-space. A ball of radius R centered at X in p-dimensions is defined using the notation: $\mathcal{B}^p_{x,R} = \{v \in \mathbb{R}^p : \|v - x\|_2{}^2 < R\}$[1].

Assume that the predictors form a d-dimensional manifold $\mathcal{M}$ in a small region surrounding X_0, and there is a function which lies on this manifold $f(\cdot) : \mathcal{M} \to \mathbb{R}$. Note that $d \leq p$, and that d is in general a function of X_0; however, implicit in these assumptions is that the manifold $\mathcal{M}$ is continuous within the ball. The manifold can be formally defined at point X_0 as the image of a local chart:

$$\mathcal{M} = \{\phi(u) \in \mathcal{B}^p_{x_0,\mu} \subset \mathbb{R}^p : u \in \mathcal{B}^d_{0,r} \subset \mathbb{R}^d\}, \tag{5.1}$$

for small $\mu, r > 0$. An example of this setup for $p = 2$ and $d = 1$ can be seen in Figure 4.1.

There are n measurements of the predictors $X_i \in \mathbb{R}^p$, for $i = \{1, \ldots, n\}$, where the X_i are independent and identically distributed. There are also n noisy measurements of the function $Y_i = f(X_i) + \epsilon_i$, where the ϵ_i are independent and identically distributed with $\mathbb{E}(\epsilon_i) = 0$ and $\mathrm{Var}(\epsilon_i) = \sigma^2$. Let $\kappa, M > 0$ be finite constants, and assume the following:

1. The kernel function $K(\cdot)$, which is used to localize the new estimators by selecting points within a small region of predictor-space, is three-times differentiable and radially symmetric. These imply that $K(\cdot)$ and $K''(\cdot)$ are even functions, while $K'(\cdot)$ is an odd function.

2. The bandwidth h is the radius of predictor points about X_0 which are used by the new estimators, and it has the following asymptotic rate: $h = \kappa n^{-1/(d+4)}$.

3. The kernel $K(\cdot)$ either has exponential tails or a finite support [33]. Mathematically speaking,

$$\mathbb{E}\left[K^\gamma((X - x)/h)w(X)\mathbb{1}(X \in (\mathcal{B}^p_{x,h^{1-\epsilon}})^c)\right] = o(h^{d+4}),$$

   for $\gamma \in \{1, 2\}$, $0 < \epsilon < 1$, and $|w(x)| \leq M(1 + |x|^2)$.

4. The local chart $\phi(\cdot)$ used to define the manifold in (5.1) is invertible and three-times differentiable within its domain. The manifold $\mathcal{M}$ is a differentiable manifold, and the function $f(\cdot)$ is three-times differentiable: $\|\partial_i \partial_j \partial_k (f \circ \phi)\|_\infty \leq M$.

[1] In this notation, subscripts are denoted in lower case. For instance, the ball surrounding the point X_0 is denoted in subscripts with the lower case x_0.

5. The probability density cannot be defined in the ambient space because the Lebesgue measure of a manifold is generally zero. The probability density is defined with respect to a d-dimensional measure by inducing the density with the map $\phi(\cdot)$ [33]:

$$\mathbb{P}(X \in \mathcal{S}) = \mathbb{Q}(Z \in \phi^{-1}(\mathcal{S})),$$

where $\mathcal{S} \subseteq \mathbb{R}^p$. The density $\mathbb{Q}(\cdot)$ is denoted by $F(z)$, and assume that it is three-times differentiable and strictly positive at $(z = 0) \in \phi^{-1}(X_0)$.

6. The Tikhonov-type regularization parameter λ_n is non-decreasing and satisfies the following rates: $\lambda_n/nh^{d+2} \to \infty$ and $h\lambda_n/nh^{d+2} \to 0$. The lasso-type regularization parameter μ_n is non-increasing and satisfies the following rates: $\mu_n(nh^{d+2})^{-1/2} \to 0$ and $\mu_n(nh^{d+2})^{(\gamma-1)/2} \to \infty$.

The choice of the local chart $\phi(\cdot)$ is not unique. Suppose a different local chart $\psi(\cdot)$ is chosen. Fortunately, it can be shown that the results are invariant under the change of coordinates $\psi^{-1} \circ \phi$ as long as the measure $\mathbb{Q}(\cdot)$ is defined to follow suitable compatibility conditions under arbitrary, smooth coordinate changes. This is important because it says that the results are based on the underlying geometry of the problem.

5.2 Change in Rank of Local Covariance Estimates

Kernels, bandwidth matrices, and weight matrices are used to localize the regression problem. Define a scaled kernel $K_h(U) = h^{-p}K(U/h)$, where h is a bandwidth. Then, the weight matrix centered at X_0 with bandwidth h is given by

$$W_{x_0} = \text{diag}(K_h(X_1 - X_0), \ldots, K_h(X_n - X_0)),$$

and the augmented bandwidth matrix is given by $H = H^{1/2}H^{1/2}$, where

$$H^{1/2} = \sqrt{nh^d} \begin{bmatrix} 1 & 0 \\ 0 & h\mathbb{I}_{p\times p} \end{bmatrix}.$$

.

Defining the augmented data matrix as

$$X_{x_0} = \begin{bmatrix} 1 & (X_1 - X_0)' \\ \vdots & \vdots \\ 1 & (X_n - X_0)' \end{bmatrix}, \tag{5.2}$$

then the weighted Gram matrix of X_{x_0} is

$$\hat{C}_n \triangleq \begin{bmatrix} \hat{C}_n{}^{11} & \hat{C}_n{}^{12} \\ \hat{C}_n{}^{21} & \hat{C}_n{}^{22} \end{bmatrix} = h^p \cdot H^{-1/2} X'_{x_0} W_{x_0} X_{x_0} H^{-1/2}. \tag{5.3}$$

A formal statement is given in the appendix, but the weighted Gram matrix (5.3) converges in probability to the following population parameters:

$$\begin{aligned} C^{11} &= F(0) \int_{\mathbb{R}^d} K(d_u\phi \cdot u) du \\ C^{21} &= C^{12'} = 0 \\ C^{22} &= F(0) d_u\phi \cdot \left[\int_{\mathbb{R}^d} K(d_u\phi(0) \cdot u) uu' du \right] \cdot d_u\phi'. \end{aligned} \tag{5.4}$$

Expanding the $\hat{C}_n{}^{22}$ term from the weighted Gram matrix (5.3) into

$$\hat{C}_n{}^{22} = \frac{1}{nh^{d+2-p}} \sum_{i=1}^{n} K_h(X_i - X_0)(X_i - X_0)(X_i - X_0)', \tag{5.5}$$

it becomes apparent that $\hat{C}_n{}^{22}$ can be interpreted as a local covariance matrix. The localization associated with the kernel adds a bias, and this causes problems when doing regression. The $\hat{C}_n{}^{11}$ term does not cause problems because it is akin to the denominator of the Nadaraya-Watson estimator [68] which does not need regularization.

The bias of the local covariance estimate $\hat{C}_n{}^{22}$ causes problems in doing regression, because the bias can cause the rank of $\hat{C}_n{}^{22}$ to be different than the rank of $C_n{}^{22}$. The change in rank found in the general case of the local, nonlinear manifold causes problems with MP which is discontinuous when the covariance matrix changes rank [3]. In the special case of a global, linear manifold, a similar change in rank can happen because of errors-in-variables. It is worth noting that MP works well in the case of a singular design matrix.

5.3 Manifold Regularization

To compensate for this change in rank, a Tikhonov-type regularization similar to RR and EN can be used. The distinguishing feature of the new estimators is the particular form of the regularizing matrix used. The approach is to estimate the tangent plane at X_0 of the manifold $\mathcal{M}$ and then constrain the regression coefficients to lie close to the principal components of the manifold. The idea comes from the intuition on exterior derivatives that was discussed in Chapter 4. An advantage of this regularization is that it is easy to apply lasso-type regularization, and this combination of two types of regularization is similar to EN.

To constrain the regression coefficients to lie close to the manifold, the problem can be posed as a weighted least-squares problem with Tikhonov-type regularization:

$$\hat{\beta} = \arg\min \|W(Y - X\beta)\|_2{}^2 + \lambda \|\Pi\beta\|_2{}^2. \tag{5.6}$$

The matrix Π is a projection matrix chosen to penalize β for lying off of the manifold. Contrast this to RR and EN which choose Π to be the identity matrix. Thus, RR and EN do not fully take the manifold structure of the problem into consideration.

Stated in another way, Π is a projection matrix which is chosen to penalize the components of β which are perpendicular to the manifold. The cost function being minimized has the term $\|\Pi\beta\|_2{}^2$, and this term is large if β has components perpendicular to the manifold. Components of β parallel to the manifold are not penalized because the projection onto these directions is zero.

Since the manifold is unknown *a priori*, it must be learned from the sample local covariance matrix $\hat{C}_n{}^{22}$. This can be done by looking at the principal components of $\hat{C}_n{}^{22}$, and so the new estimators are very closely related to PCR. Consider an eigenvalue decomposition of $\hat{C}_n{}^{22}$:

$$\hat{C}_n{}^{22} = \begin{bmatrix} \hat{U}^R & \hat{U}^N \end{bmatrix} \operatorname{diag}(\lambda^1, \ldots, \lambda^p) \begin{bmatrix} \hat{U}^R & \hat{U}^N \end{bmatrix}', \tag{5.7}$$

where $\hat{U}^R \in \mathbb{R}^{p \times d}$, $\hat{U}^N \in \mathbb{R}^{p \times (p-d)}$, and $\lambda^1 \geq \lambda^2 \geq \ldots \geq \lambda^p$. Note that the eigenvalue decomposition always exists because $\hat{C}_n{}^{22}$ is symmetric. The estimate of the manifold is given by the d most relevant principal components, and the remaining principal components are perpendicular to the estimated manifold.

Because the projection matrix Π should project β onto the directions perpendicular to the estimated manifold, define the following projection matrices

$$\begin{aligned} \hat{\Pi} &\triangleq \hat{U}^N \hat{U}^{N'} \\ \hat{P} &\triangleq \operatorname{diag}(0, \hat{\Pi}) \end{aligned} \tag{5.8}$$

The choice of d is a tunable parameter that is similar to the choice in PCR. These matrices act as a regularizer because d can always be chosen to ensure that $\operatorname{rank}(\hat{C}_n{}^{22} + \lambda \hat{\Pi}_n) = p$. The following theorem regarding the full regularizing matrix $\hat{P}$ concerns the regularization in rank:

Theorem 5.3.1 (Lemma A.0.2, Part (d))**.** Under the assumptions given in Section 5.1, the following holds with probability one:

$$\operatorname{rank}\big(\hat{C}_n + \lambda_n \hat{P}_n / n h^{d+2}\big) = p + 1. \tag{5.9}$$

The new estimators can perform better than PCR because of a subtle difference. PCR requires that the estimate lies on exactly the first d most relevant principal components; however, the new estimators only penalize for deviation from the d most relevant principal components. This is advantageous because in practice d is not known exactly and because the principal components used are estimates of the true principal components. Thus, this regularization is more robust to errors in the estimates of the principal components. Also, the new regularization easily permits the addition of lasso-type regularization to potentially improve the estimation. PCR cannot be easily extended to have lasso-type regularization.

Denote the function value at X_0 as $f\big|_{x_0}$, and denote the exterior derivative of $f(\cdot)$ at X_0 as $d_x f\big|_{x_0}$. Then, the true regression coefficients are denoted by the vector

$$\beta' = \begin{bmatrix} f\big|_{x_0} & d_x f\big|_{x_0} \end{bmatrix}. \tag{5.10}$$

The nonparametric exterior derivative estimator (NEDE) is given by

$$\hat{\beta} = \arg\min_{\tilde{\beta}} \left\{ h^p \left\| W_{x_0}{}^{1/2} (Y - X_{x_0} \tilde{\beta}) \right\|_2^2 + \lambda_n \left\| \hat{P}_n \cdot \tilde{\beta} \right\|_2^2 \right\}, \tag{5.11}$$

where $\hat{P}_n$ is defined using (5.8) with $\hat{C}_n$. Similarly, the nonparametric adaptive lasso exterior derivative estimator (NALEDE) is

$$\hat{\beta} = \arg\min_{\tilde{\beta}} \left\{ h^p \left\| W_{x_0}{}^{1/2} (Y - X_{x_0} \tilde{\beta}) \right\|_2^2 + \lambda_n \left\| \hat{P}_n \cdot \tilde{\beta} \right\|_2^2 + \mu_n \sum_{j=1}^{p} \frac{1}{\hat{w}_j{}^{\gamma}} \left| \tilde{\beta}_j \right| \right\}, \tag{5.12}$$

where $\hat{P}_n$ is define in (5.8) using $\hat{C}_n$, $\hat{w}$ is the solution to (5.11), and $\gamma > 0$.

These estimators have nice statistical properties, as the following theorem says:

Theorem 5.3.2. If the assumptions in Section 5.1 hold, then the NEDE (5.11) and NALEDE (5.12) estimators are consistent and asymptotically normal:

$$H^{1/2}(\hat{\beta} - \beta) \xrightarrow{d} C^{\dagger} \mathcal{N}(B', \sigma^2 V),$$

where B and V are respectively given in (A.3) and (A.4). Furthermore, we asymptotically have that $\hat{\beta}' \in \mathbb{R} \times T_P^* \mathcal{M}$. The NALEDE (5.12) estimator has the additional feature that

$$\mathbb{P}(\operatorname{sign}(\hat{\beta}) = \operatorname{sign}(\beta)) \to 1.$$

Note that the asymptotic normality is biased because of the bias typical in nonparametric regression. This bias is seen in both the NEDE (5.11) and NALEDE (5.12) estimators, but examining B one sees that the bias only exists for the function estimate $\hat{f}_{x_0}$ and not for the exterior derivative estimate $d_x\hat{f}|_{x_0}$. This bias occurs because h is chosen to converge at the optimal rate. Choosing h to converge at a faster rate would lead to no asymptotic bias, but the estimates would converge at a slower rate.

It is worth noting that the rate of convergence in Theorem 5.3.2 has an exponential dependence on the dimension of the manifold d, and these particular rates are due to the assumption of the existence of three derivatives. As is common with local regression, it is possible to improve the rate of convergence by using local polynomial regression which assumes the existence of higher-order derivatives [169, 33]. However, the general form of local polynomial regression on manifolds would require the choice of a particular chart $\phi(\cdot)$ and domain $\mathcal{U}$. Local linear regression on manifolds is unique in that one does not have to pick a chart and domain.

As a last note, recall that the rate of convergence in Theorem 5.3.2 depends on the dimension of the manifold d and does not depend on the dimension p of the ambient space. One might mistakenly think that this means that the estimator converges in the "large p, small n" scenario, but without additional assumptions these results are only valid for when p grows more slowly than n. Analogous to other "large p, small n" settings, faster rates of convergence can be obtained by assuming and exploiting sparsity, which is the subject of the next section.

5.4 Large p, Small n

The key difference in the 'large p, small n" setting is the need to regularize the covariance matrix. The NEDE (5.11) and NALEDE (5.12) estimators use the eigenvectors of the sample covariance matrix, and it is known [128, 32] that the sample covariance matrix is poorly behaved in the "large p, small n" setting. To ensure the sample eigenvectors are consistent estimates, some form of covariance regularization [128, 224, 32] must be used.

The new estimators use the regularization technique used in [32] for ease of analysis and because other regularization techniques [128, 224] do not work when the true covariance matrix is singular. The scheme in [32] works by thresholding the covariance matrix, which leads to consistent estimates as long as the threshold is correctly chosen. Define the thresholding operator as

$$T_t(m) = m\mathbb{1}(|m| > t),$$

and by abuse of notation $T_t(M)$ is $T_t(\cdot)$ applied to each element of M.

The setup and assumptions are nearly identical to those of the fixed p case described in Section 5.1. The primary differences are that (a) d, p, n increase at different rates towards infinity, and (b) there is some amount of sparsity in the manifold and in the function. The population parameters C_n, analogous to (5.4), are functions of n and are defined in nearly the same manner, except with $[{C_n}^{21}]_k = F(0)/2 \int_{\mathbb{R}^d} K(d_u\phi \cdot u)\partial_{ij}\phi^k u^i u^j du$. Their estimates are now defined

$$\hat{C}_n = H^{-1}X'_{x_0}W_{x_0}X_{x_0};$$

compare this to (5.3). Just as C_n can be interpreted as a local covariance matrix, define a local cross-covariance matrix:

$$R_n = \begin{bmatrix} {R_n}^1 \\ {R_n}^2 \end{bmatrix} = \begin{bmatrix} {C_n}^{11} \cdot f|_{x_0} \\ {C_n}^{21} \cdot f|_{x_0} + {C_n}^{22} \cdot d_x f|_{x_0} \end{bmatrix},$$

and the estimates are given by

$$\hat{R}_n = H^{-1/2} X'_{x_0} W_{x_0} Y.$$

For the sake of brevity, the other the differences from Section 5.1 are summarized. The following things are different:

1. The manifold $\mathcal{M}_n$, local chart $\phi_n(\cdot)$, manifold dimension d_n, number of predictors p_n, and density function $F_n(\cdot)$ are all functions of n. The subscript n is dropped when it is clear from the context. These objects are defined in the same manner as in Section 5.1, and assume that the density $F(\cdot)$ is Lipschitz continuous.

2. The asymptotic rates for d, p, n are given by $d = o(\log n)$,

$$h = o((c_n^4 n / \log p)^{-1/(4+d)})$$
$$c_n \sqrt{\frac{\log p}{n h^d}} = o(1);$$

   where c_n is a measure of sparsity that describes the number of non-zero entries in covariance matrices, exterior derivative, etc.

3. The kernel $K(\cdot)$ has finite support and is Lipschitz continuous, which implies that

$$K\left(\frac{\phi(hu) - \phi(0)}{h}\right) = K(d_u\phi \cdot u) = 0,$$

   for $u \notin \mathcal{B}_{0,\Omega}^{d_n}$. Contrast this to the second assumption in Section 5.1.

4. The local chart $\phi_n(\cdot)$, function $f_n(\cdot)$, and local (cross-)covariance matrices C_n, R_n satisfy the following sparsity conditions:

$$\sum_{k=1}^{p} \mathbb{1}(Q^k \neq 0) \leq c_n \quad \text{and} \quad \left|Q^k\right| \leq M, \tag{5.13}$$

   for (derivatives of the local chart; the index k denotes the k-th component of the vector-valued ϕ) $Q^k = \partial_i \phi^k, \partial_{ij}\phi^k, \partial_{ijm}\phi^k, \partial_{ijmn}\phi^k$; for (derivatives of the function) $Q^k = [d_x f]_k, \partial_{ik}(f \circ \phi), \partial_{ijk}(f \circ \phi)$; and for (local covariance matrices) $Q^k = [C_n]_{ik}, [R_n]_{ik}$.

5. The smallest, nonzero singular value of the local covariance matrix is bounded. That is, there exists $\epsilon > 0$ such that

$$\inf_{n>0} \left(\inf_{\sigma(\cdot)>0} \sigma(C_n) \right) > \epsilon. \tag{5.14}$$

   This condition ensures that the regularized inverse of the local covariance matrix is well defined in the limit; otherwise there can be a situation with ever-decreasing non-zero singular values.

6. The Tikhonov-type regularization parameter λ_n and the lasso-type regularization parameter μ_n have the following asymptotic rates:

$$\lambda_n = O_p\left(\sqrt{c_n}\left(\frac{nh^d}{\log p}\right)^{1/4}\right)$$
$$\mu_n {c_n}^{3/2}\left(\frac{\log p}{nh^d}\right)^{1/4} = o(1)$$
$$\mu_n\left({c_n}^{3/2}\left(\frac{\log p}{nh^d}\right)^{1/4}\right)^{1-\gamma} \to \infty.$$

7. The threshold which regularizes the local sample covariance matrix is given by

$$t_n = K\sqrt{\frac{\log p}{n}}, \tag{5.15}$$

where $\frac{\log p}{n} = o(1)$. This regularization will make the regression estimator consistent in the "large p, small n" case.

5.4.1 Manifold Regularization

The idea is to regularize the local sample covariance matrix by thresholding. If the true, local covariance matrix is sparse, this regularization will give consistent estimates. This is formalized by the following theorem:

Theorem 5.4.1. If the assumptions given in Section 5.4 are satisfied, then

$$\|T_t(\hat{C}_n) - C_n\| = O_p\left(c_n\sqrt{\log p/nh^d}\right)$$
$$\|T_t(\hat{R}_n) - R_n\| = O_p\left(c_n\sqrt{\log p/nh^d}\right).$$

Given consistent estimates of the true, local covariance matrix; consistent estimators can be obtained by applying the manifold regularization scheme described in Section 5.3. The nonparametric exterior derivative estimator for the "large p, small n" case (NEDEP) is given by

$$\hat{\beta}_n = \arg\min_{\tilde{\beta}} \left\|(T_t(\hat{C}_n) + \lambda_n\hat{P}_n)\tilde{\beta} - T_t(\hat{R}_n)\right\|_2^2, \tag{5.16}$$

where $\hat{P}_n$ is as defined in (5.8) except using $T_t({\hat{C}_n}^{22})$. The nonparametric adaptive lasso exterior derivative estimator for the "large p, small n" case (NALEDEP) is given by

$$\hat{\beta} = \arg\min_{\tilde{\beta}} \left\|(T_t(\hat{C}_n) + \lambda_n\hat{P}_n)\tilde{\beta} - T_t(\hat{R}_n)\right\|_2^2 + \mu_n\sum_{j=1}^{p}\frac{1}{{\hat{w}_j}^{\gamma}}\left|\tilde{\beta}_j\right|, \tag{5.17}$$

where $\hat{P}_n$ is as defined in (5.8) except using $T_t({\hat{C}_n}^{22})$ and $\hat{w}$ is the solution to (5.16). These estimators have nice statistical properties.

Theorem 5.4.2. If the assumptions given in Section 5.4 are satisfied, then the NALEDE (5.16) and NALEDEP (5.17) estimators are consistent:

$$\|\hat{\beta} - \beta\|_2 = O_p\left(c_n{}^{3/2}\left(\frac{\log p}{nh^d}\right)^{1/4}\right).$$

The NALEDEP (5.17) estimator is also sign consistent:

$$\mathbb{P}(\text{sign}(\hat{\beta}) = \text{sign}(\beta)) \to 1.$$

A proof of this theorem is not given, because it uses essentially the same argument as Theorem 5.4.3. One minor difference is that the proof uses Theorem 5.4.1 instead of Theorem 1 from [32].

5.4.2 Linear Case

The new estimators admit simple extensions in the special case where predictors lie on a global, linear manifold and the response variable is a linear function of the predictors. Specifically considering the errors-in-variables situation with manifold structure is necessary in order to present formal results, because: In principle, the new estimators provide no improvements in the linear manifold case over existing methods when there are no errors-in-variables. In practice, the new estimators sometimes provide an improvement in this case. Furthermore, they provide another solution to the identifiability problem [78]; the exterior derivative is the unique set of regression coefficients because the predictors are only sampled in directions parallel to the manifold, and there is no derivative information about the response variable in directions perpendicular to the manifold.

Suppose there are n data points and p predictors, and the dimension of the global, linear manifold is d. Assume that d, n, p increase to infinity, and leaving d fixed is a special case of the results. Consider a linear model $\eta = \Xi\overline{\beta}$, where $\eta \in \mathbb{R}^{n\times 1}$ is a vector of function values, $\Xi \in \mathbb{R}^{n\times p}$ is a matrix of predictors, and $\overline{\beta} \in \mathbb{R}^p$ is a vector.

The Ξ are distributed according to the covariance matrix Σ_ξ, which is also a singular design matrix in this case. The exterior derivative of this linear function is given by $\beta = P_{\Sigma_\xi}\overline{\beta}$, where P_{Σ_ξ} is the projection matrix that projects onto the range space of Σ_ξ. We make noisy measurements of η and Ξ:

$$X = \Xi + \nu$$
$$Y = \eta + \epsilon.$$

The noise ν and ϵ are independent of each other, and each component of ν is independent and identically distributed with mean 0 and variance σ_ν^2. Similarly, each component of ϵ is independent and identically distributed with mean 0 and variance σ^2. In this setup, the variance σ_ν^2 is identifiable [123, 114], and an alternative that works well in practice for low noise situations is to set this quantity to zero.

The setup of errors-in-variables differs from that of existing tools [42, 146], but it is important because in practice, many of the near-collinearities might be true collinearities that have been perturbed by noise. Several formulations explicitly introduce noise into the model [123, 152, 43, 67, 114]. The setting of [123, 114] is used here, because the noise in the predictors is identifiable in this situation.

The exterior derivative estimator for the "large p, small n" case (EDEP) is given by

$$\hat{\beta} = \arg\min_{\tilde{\beta}} \left\| (T_t(X'X/n) - \sigma_\nu^2 \mathbb{I} + \lambda_n \hat{P}_n)\tilde{\beta} - T_t(X'Y/n) \right\|_2^2, \tag{5.18}$$

where $\hat{P}_n$ is as defined in (5.8) except applied to ${\hat{C}_n}^{22} = T_t(X'X/n) - \sigma_\nu^2 \mathbb{I}$. This is essentially the NEDEP estimator, except the weighting matrix is taken to be the identity matrix and there are additional terms to deal with errors-in-variables. An adaptive lasso version of this estimator can also be defined. The adaptive lasso exterior derivative estimator for the "large p, small n" case (ALEDEP) is given by

$$\hat{\beta} = \arg\min_{\tilde{\beta}} \left\| (T_t(X'X/n) - \sigma_\nu^2 \mathbb{I} + \lambda_n \hat{P}_n)\tilde{\beta} - T_t(X'Y/n) \right\|_2^2 + \mu_n \sum_{j=1}^{p} \frac{1}{\hat{w}_j^{\gamma}} \left| \tilde{\beta}_j \right|, \tag{5.19}$$

where $\hat{P}_n$ is as defined in (5.8) except applied to ${\hat{C}_n}^{22} = T_t(X'X) - \sigma_\nu^2 \mathbb{I}$ and $\hat{w}$ is the solution to (5.18). The EDE and ALEDE estimators can also be analgously defined, and they are the EDEP and ALEDEP estimators without any thresholding.

The technical conditions made are essentially the same as those for the case of the local, nonlinear manifold. The primary difference is that the conditions in 5.4 hold globally, instead of locally. This also means that kernels are not used to localize the estimators, and the W matrix in the estimators is simply the identity matrix. If the theoretical rates for the regularization and threshold parameters are compatibility redefined, then it can be shown that these estimators have nice statistical properties:

Theorem 5.4.3. If the assumptions in Sections 5.4 and 5.4.2 hold, then the EDEP (5.18) and ALEDEP (5.19) estimators are consistent. They asymptotically converge at the following rate:

$$\|\hat{\beta} - \beta\|_2 = O_p\left({c_n}^{3/2} \left(\frac{\log p}{n} \right)^{1/4} \right).$$

The ALEDE (5.19) estimator is also sign consistent:

$$\mathbb{P}(\text{sign}(\hat{\beta}) = \text{sign}(\beta)) \to 1. \tag{5.20}$$

This theoretical rate of convergence is slower than that of other techniques [42, 146] because techniques for local estimation have been applied to global estimation, and the setup of the global case has not been fully utilized. However, the rate of convergence in the global case is faster than that of the local case. Furthermore, the model here has errors-in-variables, while the model used in other techniques [42, 146] assumes that the predictors are measured with no noise. Applying the various techniques to both real and simulated data shows that the new estimators perform comparably to or better than existing techniques. It is not clear if the rates of convergence for the existing techniques [42, 146] would be slower if there were errors-in-variables, and this would require additional analysis.

5.5 Estimation with Data

Applying the new estimators requires careful usage in practice. The NEDE estimator requires choosing two tuning parameters, while the NALEDE and NEDEP estimators require choosing

three; the NALEDEP estimator requires even more: four. The extra tuning parameters—in comparison to existing techniques like MP or RR—make these methods prone to over-fitting. It is crucial to select the parameters using methods, such as cross-validation or bootstrapping, that protect against over fitting. It is also important to select from a small number of different parameter-values to protect against over fitting caused by issues related to multiple-hypothesis testing [153, 166, 165].

Bootstrapping is a good choice for parameter selection with these estimators, because discontinuity with respect to the dimension d causes some methods to be inconsistent with respect to model-selection [35, 177, 178, 179]. Additionally, we suggest selecting parameters in a sequential manner; this is to reduce over fitting caused by testing too many models [153, 166, 165]. Another benefit of this approach is that it simplifies the parameter selection into a set of one-dimensional parameter searches—greatly reducing the computational complexity of the estimators. For instance, first select the Tikhonov-regularization parameter λ for RR. Using the same λ value, pick the dimension d for the NEDE estimator. The prior values of λ and d are used to pick the lasso-regularization parameter μ for the NALEDE estimator.

MATLAB implementations of both related estimators and the new estimators can be found online[2]. The lasso-type regressions were computed using the coordinate descent algorithm [74, 214], and the "improved kernel PLS by Dayal" code given in [6] was used to do the PLS regression. The increased computational burden of the new estimators, as compared to existing estimators, is reasonable because of: improved estimation in some cases, easy parallelization, and computational times of a few seconds to several minutes on a general desktop for moderate values of p.

5.6 Numerical Examples

The first two examples use simulated data, and the third example uses real data. The examples with simulated data study the estimation accuracy of various estimators as the amount of noise and number of data points vary. The third example uses a data set with pollution data and mortality rates [145] accessed from StatLib[3]. In the example, a regression is used to explain the mortality rates in terms of the pollution data.

For examples involving linear manifolds and functions, the new estimators are compared with popular methods. The exterior derivative is locally defined, but in the linear case it is identical at every point—allowing the regression to be done globally. This is in contrast to the example with a nonlinear manifold and function where a point is picked at which to do the regression. Though MP, PLS, PCR, RR, and EN are typically thought of as global methods, these estimators can be used for local, nonparametric estimation by posing the problem as a weighted, linear regression which can then be solved using either the new or existing estimators. As a note, the MP and OLS estimators are equivalent in the examples considered.

Some of the examples involve errors-in-variables, and this suggests that using an estimator that explicitly takes this structure into account. Total Least Squares (TLS) [207] does exactly this, but it performed poorly with both the simulated data and experimental data. This was expected because standard TLS is known to perform poorly in the presence of collinearities [207]. TLS performed comparably to or worse than OLS/MP, and so the results are not included.

Based on the numerical examples, it seems that the improvement in estimation error of the new estimators is mainly due to the Tikhonov-type regularization, with lasso-type regulariza-

[2] `http://www.eecs.berkeley.edu/~aaswani/EDE_Code.zip`

[3] `http://lib.stat.cmu.edu/`

tion providing additional benefit. Thresholding the covariance matrices did not make significant improvements, partly because bootsrap has difficulty in picking the thresholding parameter. Improvements may be possible by refining the parameter selection method or by changes to the estimator. The well-known tendency of lasso to overestimate the number of non-zero coefficients [147] was also observed; using stability selection [147] to select the lasso parameter would likely lead to better results.

5.6.1 Simulated Data

Data for two different models is simulated and used to compare different estimators. One model is linear, and global estimation is done in this case. The other model is nonlinear, and hence local estimation is done in this case. In both models, there are p predictors and the dimension of the manifold is $d = \text{round}(\frac{3}{4}p)$. The predictors ξ and response η are measured with noise:

$$x = \xi + \mathcal{N}(0, {\sigma_\nu}^2)$$
$$y = \eta + \mathcal{N}(0, \sigma^2).$$

And for notational convenience, let $q = \text{round}(\frac{1}{2}p)$. Define the matrix

$$F_{ij} = \begin{cases} 0.3^{|i-j|}, & \text{if } 1 \le i, j \le d \\ 0.3, & \text{if } d+1 \le i \le p \wedge j = q + i - d \\ 0.3, & \text{if } d+1 \le i \le p \wedge j = q + i + 1 - d \\ 0, & \text{o.w.} \end{cases}$$

The two models are given by:

1. Linear Model: The predictors are distributed $\xi = \mathcal{N}(0, FF')$, and the function is

$$\eta = f(\xi) = 1 + \sum_{\substack{i=1 \\ i \text{ is odd}}}^{q} \xi_i. \tag{5.21}$$

   If $w = \begin{bmatrix} 1 & 0 & 1 & \dots \end{bmatrix}$ is a vector with ones in the odd index-positions and zeros elsewhere, then the exterior derivative of this linear function at every point on the manifold is given by the projection of w onto the range space of the matrix F.

2. Nonlinear Model: The predictors are distributed $\xi = \sin(\mathcal{N}(0, FF'))$, and the function is

$$\eta = f(\xi) = 1 + \sum_{\substack{i=1 \\ i \text{ is odd}}}^{q} \sin(\xi_i). \tag{5.22}$$

   Consider local regression about the point $x_0 = \begin{bmatrix} 0 & \dots & 0 \end{bmatrix}$. If $w = \begin{bmatrix} 1 & 0 & 1 & \dots \end{bmatrix}$ is a vector with ones in the odd index-positions and zeros elsewhere, then the exterior derivative of this nonlinear function at the origin is given by the projection of w onto the range space of the matrix F.

Table 5.1 shows the average square-loss estimation error $\|\hat{\beta} - \beta\|_2^2$ for different estimators using data generated by the linear model and nonlinear model given above, over different noise variances and number of data points n. One-hundred replications—of generating data and doing a regression—were conducted, and this helped to provide standard deviations of square-loss estimation error to show the variability of the estimators.

One curious phenomenon observed is that the estimation error goes down in some cases as the error variance of the predictors ${\sigma_\nu}^2$ increases. To understand why, consider the sample covariance matrix in the linear case $\hat{S} = X'X$ with population parameter $S = FF' + {\sigma_\nu}^2\mathbb{I}$. Heuristically, the OLS estimate will tend to $(FF' + {\sigma_\nu}^2\mathbb{I})^{-1}X'Y$, and the error in the predictors actually acts as the Tikhonov-type regularization found in RR, with lower levels of noise leading to less regularization.

The results indicate that the new estimators are not significantly more variable than existing ones, and they perform competitively against existing estimators. Though they are closely related to PCR, RR, and EN; the new estimators performed comparably to or better than these estimators. PLS also did quite well, and the new estimators did better than PLS in some cases. Increasing the noise in the predictors did not seem to significantly affect the qualitative performance of the estimators, except for OLS as explained above.

Section 5.4.2 discussed how the convergence rate of the new linear estimators is of order $n^{-1/4}$ which is in contrast to the typical convergence rate of $n^{-1/2}$ for lasso-type regression [146]. We believe that this theoretical discrepancy is because the model has errors-in-variables while the standard model used in lasso-type regression does not [146]. These theoretical differences do not seem significant in practice. As seen in Tables 5.1, the new estimators can be competitive with existing lasso-type regression.

5.6.2 Pollution Data

A well-known data set [145], accessed from StatLib[4], can be used to build a model to predict age-adjusted mortality rates in different U.S. metropolitan areas based on weather, demographic, and pollution variables. The data set is interesting because it has significant collinearity in the predictors, there are some outliers, and there is the potential to do some nonlinear transformations on the data [145, 104, 100]. The collinearity of the data suggests that there is an underlying manifold on which the predictors lie, suggesting the possibility of better estimates by exploiting this manifold structure.

To compare the different estimators, 100 replications of the following experiment were conducted: The data ($n = 60$ data points) was split into a training set $n_t = 45$ and validation set $n_v = 15$, the training set was used to calculate a model using different estimators, and then these models were used to predict the mortality rates of the validation set. Because there is some subjectivity in the removal of outliers and in the use of nonlinear transformations, this process was repeated four times. The first time included all of the untransformed data, the second time included the untransformed data with outliers removed, the third time included all of the data and with a logarithmic transformation applied to the pollution variables, and the fourth time included the transformed data with outliers removed.

The logarithmic transformation of the pollution variables was proposed by [145], but they did not use it in their analysis; this transformation was used by [100] in their models. Furthermore, [145, 100] have identified outliers. These outliers correspond to the California areas, which had

[4] http://lib.stat.cmu.edu/

	Linear Model: ${\sigma_\nu}^2 = 0.01, \sigma^2 = 1.00$ $\sigma_\nu^2 = 0.5^2$					
	$n = 10$		$n = 100$		$n = 1000$	
OLS/MP	4.048	(2.134)	7.212	(4.255)	0.526	(0.337)
RR	2.658	(1.076)	0.408	(0.250)	0.158	(0.076)
EN	2.674	(1.138)	0.147	(0.276)	0.018	(0.009)
PLS	2.522	(0.586)	0.540	(0.254)	0.030	(0.011)
PCR	3.409	(0.892)	1.680	(0.136)	1.572	(0.047)
EDE	2.658	(1.077)	0.402	(0.227)	0.158	(0.076)
ALEDE	2.694	(1.143)	0.140	(0.250)	0.015	(0.006)
	Linear Model: ${\sigma_\nu}^2 = 0.10, \sigma^2 = 1.00$ $\sigma_\nu^2 = 0.5^2$					
	$n = 10$		$n = 100$		$n = 1000$	
OLS/MP	3.481	(1.436)	1.185	(0.554)	0.168	(0.054)
RR	2.641	(1.042)	0.880	(0.348)	0.252	(0.047)
EN	2.668	(1.085)	0.648	(0.530)	0.065	(0.015)
PLS	2.558	(0.617)	0.668	(0.196)	0.139	(0.057)
PCR	3.442	(1.095)	1.690	(0.177)	1.576	(0.051)
EDE	2.642	(1.043)	0.826	(0.297)	0.247	(0.047)
ALEDE	2.669	(1.086)	0.584	(0.471)	0.051	(0.014)
	Nonlinear Model: ${\sigma_\nu}^2 = 0.01$ $\sigma_\nu^2 = 0.5^2$					
	$n = 20$		$n = 100$		$n = 1000$	
OLS/MP	885.8	(2118)	3.902	(1.976)	0.331	(0.179)
RR	2.078	(0.655)	1.197	(0.301)	0.281	(0.127)
EN	2.047	(0.685)	1.115	(0.401)	0.153	(0.087)
PLS	2.127	(0.569)	0.998	(0.281)	0.146	(0.074)
PCR	2.746	(0.398)	1.736	(0.252)	0.122	(0.043)
NEDE	2.075	(0.652)	1.146	(0.303)	0.271	(0.113)
NALEDE	2.041	(0.687)	1.047	(0.412)	0.134	(0.061)
	Nonlinear Model: ${\sigma_\nu}^2 = 0.1$ $\sigma_\nu^2 = 0.5^2$					
	$n = 20$		$n = 100$		$n = 1000$	
OLS/MP	308.4	(935.9)	2.862	(1.449)	0.442	(0.088)
RR	2.700	(2.882)	1.643	(0.258)	0.718	(0.154)
EN	2.684	(2.876)	1.635	(0.265)	0.586	(0.229)
PLS	2.434	(0.723)	1.446	(0.433)	0.753	(0.143)
PCR	2.907	(0.372)	1.967	(0.391)	0.469	(0.262)
NEDE	2.696	(2.882)	1.590	(0.293)	0.666	(0.124)
NALEDE	2.681	(2.876)	1.579	(0.309)	0.457	(0.134)

Table 5.1: Averages and standard deviations over 100 replications of square-loss estimation error for different estimators using data generated by the linear model and nonlinear model given in Section 5.6.1, over different noise variances and number of data points n.

	I		II		III		IV	
OLS/MP	2491	(1451)	2192	(823.4)	2520	(1087)	1989	(788.0)
RR	1917	(675.1)	2041	(848.5)	2332	(3801)	1948	(838.1)
EN	1926	(675.5)	2041	(849.3)	2663	(2879)	1951	(840.8)
PLS	2643	(3536)	2567	(2284)	6.7e5	(2.4e5)	2475	(1340)
PCR	2374	(810.0)	2273	(807.6)	4125	(1581)	1938	(748.9)
EDE	1901	(631.5)	2029	(821.5)	2331	(3801)	1890	(800.0)
ALEDE	1907	(632.9)	2030	(823.0)	2664	(2878)	1894	(801.6)

Table 5.2: Averages and standard deviations over 100 replications of validation set prediction error for different estimators using the pollution data [145]. I = All of the untransformed data; II = Untransformed data with outliers removed; III = All of the data and with a logarithmic transformation applied to pollution variables; IV = Transformed data with outliers removed.

abnormally high pollution levels in the 1960s. However, these outliers were not removed in [104]; so the resultant predictive power of models generated both with and without these outlier values was considered.

Table 5.2 gives the prediction error of the models generated by different estimators on the validation set. The specific quantity provided is $\|X_v\hat{\beta} - Y_v\|_2{}^2/n_v$, where X_v, Y_v are respectively the predictors (with possibly a nonlinear transformation applied) and mortality rates of the validation set. The results from this real data set shows that the new estimators can provide improvements over existing tools, because they have the lowest prediction errors in various cases. The best models are provided by removing outliers and making the nonlinear transformation of data, though the models generated by EDE and ALEDE when no data is removed or transformed is nearly as good.

5.7 Future Directions

By interpreting collinearity as predictors on a lower-dimensional manifold, a new regularization, which has connections to PCR and RR, was developed for linear regression and local linear regression. This viewpoint also allows interpretation of the regression coefficients as estimates of the exterior derivative. The consistency of these new estimators was proved in both the classical case and the "large p, small n" case and this is useful from a theoretical standpoint.

Numerical examples using simulated and real data show that the new estimators can provide improvements over existing estimators in estimation and prediction error. Specifically, the Tikhonov-type and lasso-type regularizations provided improvements, and the thresholding regularization did not provide major improvements. This is not to say that thresholding is not a good regularization, because as showed: From a theoretical standpoint, thresholding does provide consistency in the "large p, small n" situation. This leaves open the possibility of future work on how to best select this thresholding parameter value.

There is additional future work possible on extending this new set of estimators. There is some benefit provided by shrinkage from the Tikhonov-type regularization which is independent of the manifold structure. Exploring more fully the relationship between manifold structure and shrinkage will likely lead to improved estimators.

Chapter 6

Statistical System Identification in *Drosophila* embryogenesis

Inferring regulatory networks in animals is challenging because of the large number of genes and the presence of redundant and indirect interactions. To build the highest quality models, it will be necessary to use multiple data sets, including: gene expression, genome wide binding, and network perturbation data. However, combining multiple data types to infer regulatory networks is still an open problem.

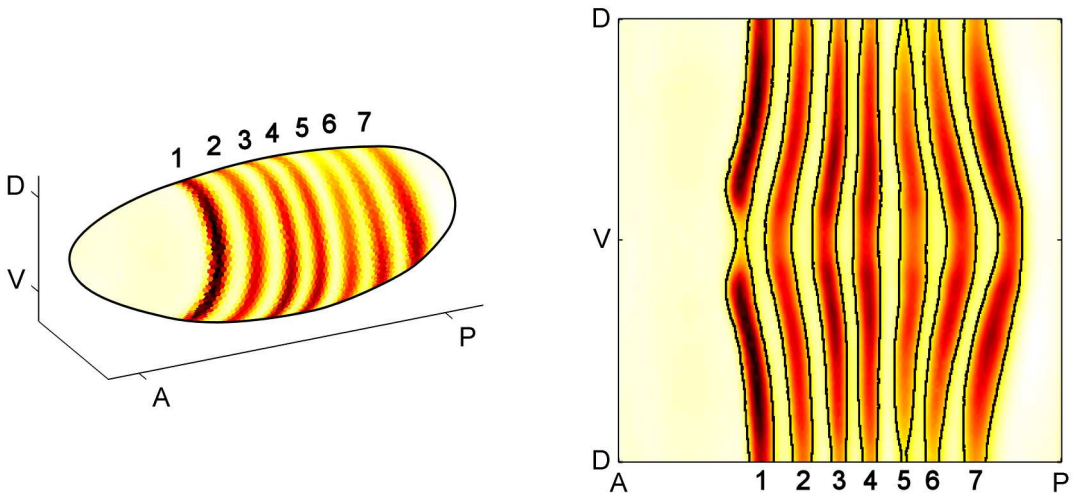
An intermediate problem is to use only gene expression data to infer regulatory networks. The relationships between the expression levels of transcription factors and target genes are used to predict which genes are regulatory. While much work has been done in this area, it is critical to understand the maximum amount of information that can be obtained about the network using this strategy.

Typical approaches for inferring regulatory networks have been to assume a model formulation and have then fitted the data to this formulation [25, 142]. This is the parametric system identification of Chapters 2 and 3. Many models have been proposed, including coexpression networks [63, 194, 31], information-theoretic [40, 191, 165], regression onto dynamical systems [59, 190, 37, 55, 160], and graphical models (including Bayesian networks) [75, 219, 215, 76, 212].

The primary differences between these models lie in the trade-off between statistical and interpretational issues. Techniques like Bayesian networks, graphical models, and information-theoretic models have protections against over-fitting (i.e., fitting models with many parameters to a small amount of experimental data); however, these techniques do not provide dynamical models which can generate new biological insights. On the other hand, techniques such as nonlinear regression networks and regression onto dynamical systems provide more biologically interpretable models, but sometimes suffer from inaccurate assumptions or over-fitting of the model to the data.

There is disagreement on the necessity of dynamical [59, 190, 37, 55, 160, 165, 75, 219, 215, 76, 212] as opposed to static [63, 40, 191, 194, 31, 72, 182, 79, 9] models. Dynamical models are more philosophically pleasing because regulatory networks contain temporal characteristics: For example, a protein binds to DNA and initiates transcription, which eventually leads to transport of the mature mRNA to the cytoplasm. Yet the argument is often made that static models provide a quasi-steady-state interpretation of the network that may provide a sufficient approximation. Rigorous comparison of the two approaches is lacking.

Dynamical modeling of animal regulatory networks has a long history [204, 24, 59, 210, 190, 112, 37, 151]. It is a powerful approach in which researchers hypothesize a set of nonlinear,



(a) (b)

Figure 6.1: (a) A three-dimensional plot of the *Drosophila* embryo showing the experimentally measured pattern of *eve* mRNA as it appears in late stage 5. There are seven distinct stripes located along the anterior-posterior axis of the embryo, the stripes being symmetric about the left-right axis. (b) A two-dimensional cylindrical projection of a stage 5 *Drosophila* embryo provides an easier visualization of the details of the *eve* mRNA patterns.

differential equations to describe the network, but it requires significant prior knowledge about the network. If there is insufficient biological knowledge about the network, then the structure of the equations can be incorrectly chosen. And if the model is not carefully chosen, it will have a large number of parameters, possibly leading to weak biological effects being erroneously identified as strong effects. Furthermore, it is sometimes shown that a wide range of different parameter values can reproduce the biological behavior of the network, which could be taken as evidence for either network robustness or over-fitting [210].

The purpose of this chapter is to discuss the application of local linearization system identification described in Chapter 4 using the NEDE estimator from Chapter 5 to a network found in *Drosophila* embryos [17]. The particular network which is studied here is the formation of *eve* mRNA stripes during stage 5 of embryogenesis, and this work was in collaboration with Mark Biggin and the Berkeley *Drosophila* Transcription Network Project (BDTNP). Their pattern during this stage is shown in Figure 6.1. Because of the ubiquity of static models in mathematical modeling of biology, a comparison between similar dynamical and static models is provided. The biological insights of the local linearization modeling is also given.

6.1 Experimental Data

The BDTNP [138, 72] experimentally measured protein and mRNA concentrations in cells of a *Drosophila melanogaster* embryo. The embryos are fluorescently stained, to label expression patterns and nuclear DNA, and mounted on slides. The embryos to be imaged are visually exam-

ined to determine their developmental stages. After staging the embryos, three-dimensional image stacks of the embryos are taken using two-photon microscopy. The image stacks are processed algorithmically to produce a virtual embryo [138, 72], which is essentially an average of factor concentrations and of the Drosophila geometry at cellular resolution.

The virtual embryo contains an average number of nuclei, factor concentrations in each cell, and embryo geometry information. The geometry data includes the location and shape of the cells in the embryo, the neighbors of each cell, and the distance between neighbors. Each of cells in the virtual embryo (there are $E = 6078$) can be thought of as a different experiment on the regulatory network, though these are not independent experiments in an absolute sense because neighboring cells are bathed in the same transcription factors. Different cells can be regarded as experiments which provide information on fluctuations around locally average effects. This data is given at six different points in time ($T = 6$) which correspond to six intervals during stage 5 of the Drosophila embryo development. To apply our technique to this data, we assume that the regulatory network does not spatially vary and that any variation in the spatial patterns is due to a difference in the species concentrations in each cell. We use the biologically accepted assumption that the original spatial variations are caused by maternal influences [127].

6.2 Dynamical Model

The locally linear ODE system identification method described in Chapter 4 was used to generate a model of *eve* mRNA pattern (Figure 6.1) formation during stage 5 of embryogenesis, and the other states in the model were the five factors which are agreed to affect the *eve* pattern: Krüppel (KR), Giant (GT), Knirps (KNI), Hunchback (HB) and Bicoid (BCD) [182, 79, 9]. The model was computed in 7 hours, and only data from the first two time points were used to generate the model. Denoting *eve* mRNA concentration as $[eve]$, it is worth noting that the time derivative of *eve* mRNA concentration does not depend on $[eve]$; more specifically, the local model is assumed to be

$$\frac{d[eve]}{dt} = a_{[bcdP]}\left([bcdP] - \overline{[bcdP]}\right) + a_{[gtP]}\left([gtP] - \overline{[gtP]}\right) + a_{[kniP]}\left([kniP] - \overline{[kniP]}\right) + a_{[hbP]}\left([hbP] - \overline{[hbP]}\right) + a_{[KrP]}\left([KrP] - \overline{[KrP]}\right), \quad (6.1)$$

where the overline notation indicates the operating point of the linearization. This model is a set of nonparametric ordinary differential equations (ODE) which are fit using the nonparametric exterior derivative estimator (NEDE) [12, 14]. For these reasons, we call the resulting model the NODE (an amalgamation of NEDE and ODE) model. The NODE model specifically refers to the collection of models given in (6.1) for operating points defined by the concentrations of factors in each cell at each point in time.

The *eve* mRNA pattern generated by the model in (6.1) matches the behavior of the experimental pattern well. The experimental and simulated eve patterns are compared in Figure 6.2a. The black lines on each of the maps in Figure 6.2a show the boundaries of the experimental measurements of the eve mRNA stripes, and how they change location during Stage 5. The stripes narrow in extent, and *eve* concentration in the stripes becomes stronger. The stripes also shift anteriorly. The simulation matches this experimental behavior, and captures the changing boundaries of the eve stripes particularly well.

To quantify the accuracy of the model, the simulation error is also shown in Figure 6.2a. The model is able to accurately predict the *eve* pattern at time points corresponding to Stages

5:9-25, 5:26-50, and 5:51-75. Its predictions are less accurate for Stage 5:76-100 in some regions, especially in stripe 1. This is not unexpected because at the end of Stage 5 a new set of factors begin to regulate *eve* expression [94], something which could not be learned using only data taken from early Stage 5. If expression data from all time points is used to learn the model, the simulation better matches the experimental data (Figure 6.3).

6.3 Comparison to Static Model

Static models are often used to model biological systems. The most common of these is the spatial-correlation model [72, 141, 182, 79, 9], and they are often implicitly used to interpret experiments. Mathematically speaking, a locally linear variant of the spatial-correlation model is

$$[eve] = a_{[bcdP]}\left([bcdP] - \overline{[bcdP]}\right) + a_{[gtP]}\left([gtP] - \overline{[gtP]}\right) + a_{[kniP]}\left([kniP] - \overline{[kniP]}\right) + a_{[hbP]}\left([hbP] - \overline{[hbP]}\right) + a_{[KrP]}\left([KrP] - \overline{[KrP]}\right), \quad (6.2)$$

and the coefficients of this model can be fit in a manner analogous to that described in Chapters 4 and 5.

There has been little work on understanding the applicability of static versus dynamic models for specific systems. In this particular instance, this is the question of whether the spatial-correlation model is more accurate and useful. To understand this, a spatial-correlation model was generated using the first two time points of data; the model was then used to predict the *eve* pattern at later points in time. The experimental and simulated patterns, along with simulation error, is shown in Figure 6.2b. This model much more poorly predicts the pattern at later stages (compare to Figure 6.2a). The dynamical ODE model predicts an *eve* pattern that is 22% more accurate than the pattern predicted by the spatial-correlation model. Thus, in a direct comparison of a static (spatial-correlation) model and a dynamical (NODE) model, the dynamic model is superior.

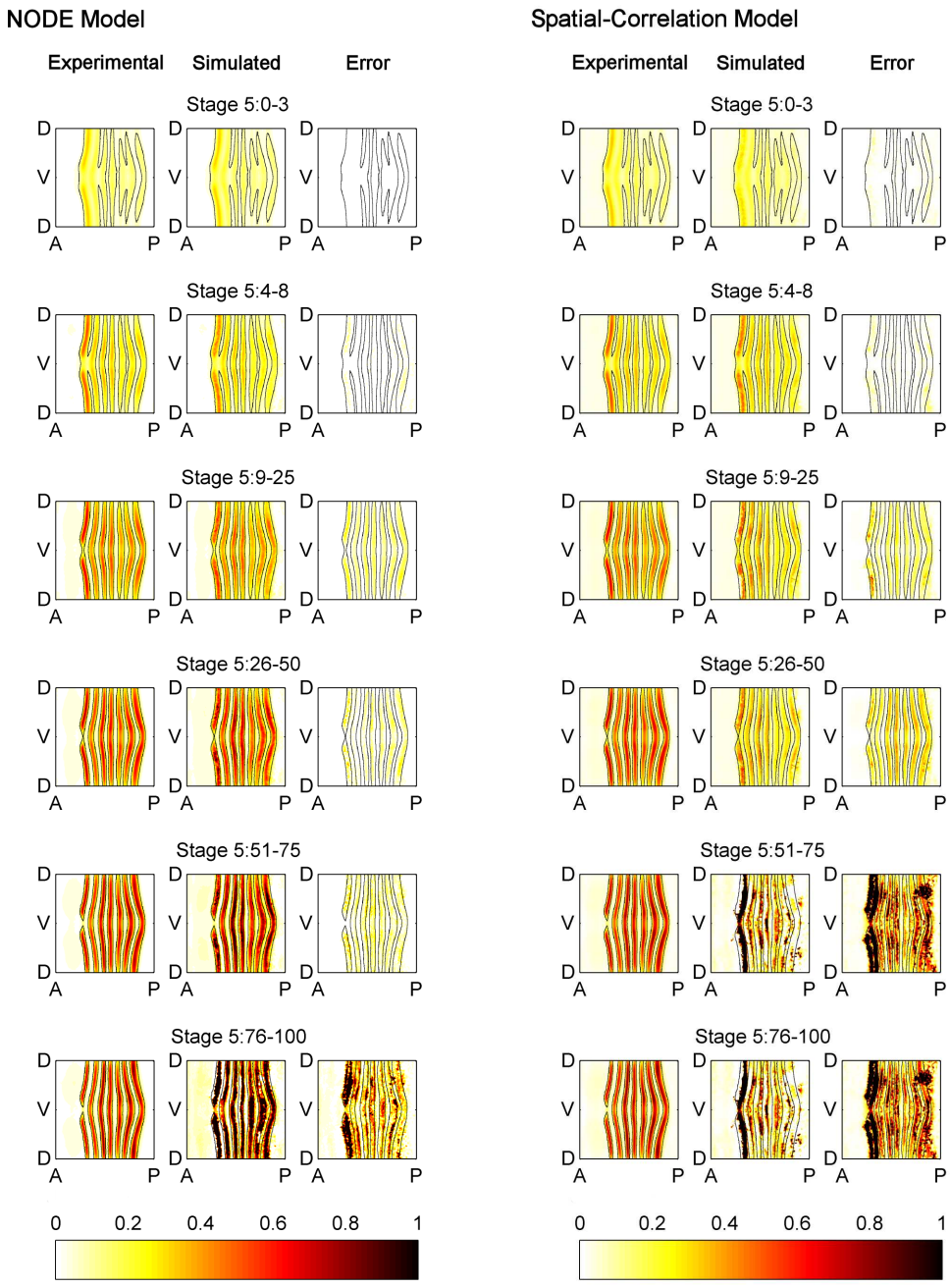
6.4 Factor Activity

Factor activity is a quantitative measure of the impact of a factor on the target gene expression, and it is a particular scaling of the coefficients of the model. It takes into account the concentration of the factors and the coefficients of (6.1), which describe the amount of influence of the factors on the target expression. Without loss of generality, consider the equation for factor activity of GT on the expression of *eve* mRNA

$$a_{[gtP]}\left(\frac{1}{n}\left[X_x' W X_x\right]_{[gtP]}\right)^{1/2}, \quad (6.3)$$

where X_x is given in (5.2).

The first term is the coefficient from (6.1), and the second term in parenthesis is a measure of average GT concentration within cells whose factor concentrations are similar to the operating point x_0. The second term in parenthesis is a measure of average concentrations because it is a measure of the mean difference about the baseline concentration of x_0. To clarify the notation, suppose the i-th value of the state vector x denotes: $x_{[gtP]}$, which is GT concentration. Then the term $[X_x' W X_x]_{[gtP]}$ denotes the i-th value along the diagonal of the matrix $X_x' W X_x$.



(a) Dynamical Model

(b) Static Model

Figure 6.2: Cylindrical projections of the measured pattern of eve mRNA concentrations (left column), the NODE model simulated pattern of eve mRNA (center column), and the simulation error (right column) at six successive time points during blastoderm stage 5 (rows). The eve mRNA concentration values have been normalized to range from 0 to 1 and the simulation error shown is the absolute value of the difference between experimental and simulated eve concentration in the embryo. Plots are given for both a dynamical model (a) and a static model (b).

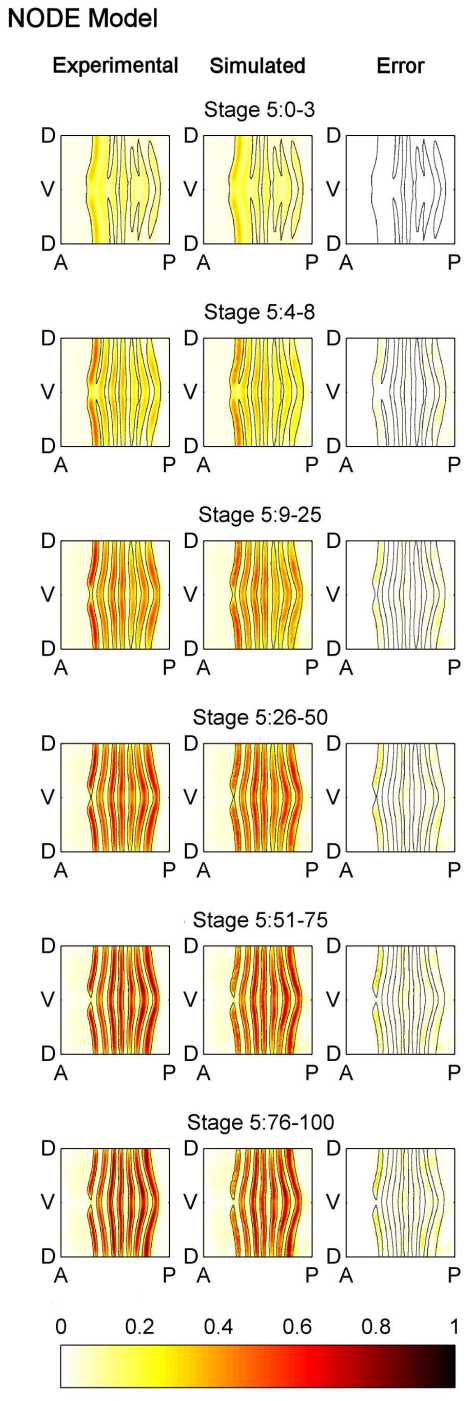


Figure 6.3: A comparison is provided between the experimental *eve* pattern and the simulated pattern generated by a locally linear ODE model which uses expression data from all time points to generate the model. The labels are the same as in Figure 6.2, and the match is better than that shown in Figure 6.2a which was generated using only the first two time points of data.

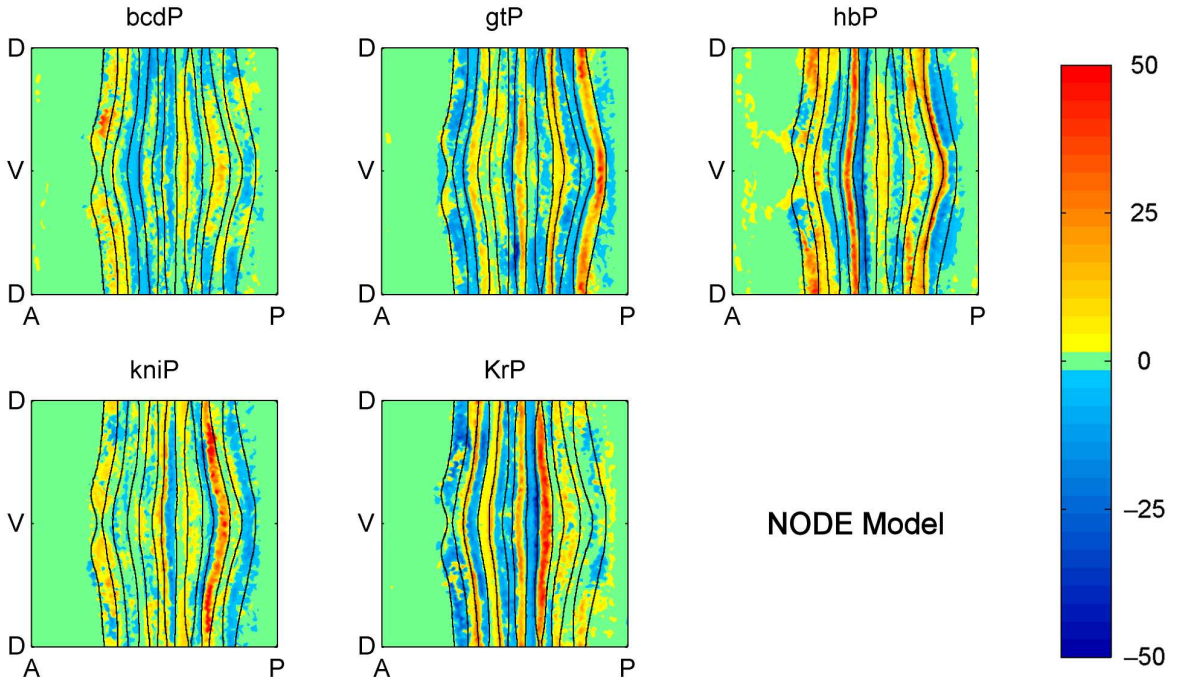


Figure 6.4: Cylindrical projections of the correlation between each factor and the change in target expression over time. The intensity of the factor activity values is the product of the coefficients of the model in Equation 4 and the average, local factor concentration. The mathematical definition of factor activity is given in Methods and Models.

The model generated by the technique in in Chapter 4 can be visualized as spatio-temporal maps of factor activities. An example of a spatial map for Stage 5:9-25 is shown in Figure 6.4, and it shows how the five factors (directly or indirectly) affect *eve* mRNA pattern formation. Blue values correspond to predicted repression (i.e., an anticorrelation between factor expression and the rate of change of target expression) and yellow/red values correspond to predicted activation (i.e., a positive correlation between factor and the change in target).

6.5 Concentration-Dependent Effects

It is known that individual stripes are often controlled via a single *cis*-regulatory module (CRM). Furthermore, current computational models generally assume that a given factor acts only as an activator or a repressor on a given CRM [210, 112, 113, 176, 65]. However, both our NODE model and our variant of the spatial-correlation model predict concentration-dependent effects in which a factor has both repressing and activating effects around a single stripe. For example, KR is predicted to repress at the posterior portion of eve stripe 2, but the model shows that it is an activator just anterior of this in cells where KR concentrations are lower (Figure 6.4). Such cases could represent spurious correlations, perhaps due to other factors having dominant effects on targets in cells where the factor under study is expressed at lower levels. However, there are

	$sign\left(a_{[gtP]}\right)$	$sign\left(d[eve]/dt\right)$
Type I Repression	$-$	$-$
Type II Repression	$-$	$+$
Type I Activation	$+$	$+$
Type II Activation	$+$	$-$

Table 6.1: Mathematical Definition of Factor Activity Classification

a number of cases where factors, including KR, have been shown to switch from activating to repressing the same target as their concentrations increase [162, 171]. Thus, the predictions of both our NODE model and our variant of the spatial-correlation model suggest that gene regulation might involve multiple mechanisms of factor action that should be considered.

The NODE model can distinguish between multiple regions of the embryo where target mRNA either increases or decreases over time, whereas spatial-correlation models, by definition, cannot. This allows the NODE model to provide more subtle distinctions of factor activity. The factor activities can be subdivided into four categories of behavior. Without loss of generality, mathematical definitions for four categories of GT activity on *eve* mRNA are given. At a given concentration x_0, if the GT coefficient from (6.1) is negative (i.e., $a_{[gtP]} < 0$) and *eve* concentration is decreasing (i.e., $d[eve]/dt < 0$), then GT is formally a Type I repressor when the factor concentrations are x_0. A summary of the other mathematical definitions is given in Table 6.1.

Chapter 7

Learning Global-Sparsity Structure

Lasso-type regularization [199, 120, 222] has been immensely successful with inducing sparsity in a multitude of applications which can be reformulated as linear regression. These regularization procedures assume that different models have decoupled sparsity structures. But, this can be a restrictive framework. In some applications, such as optical character recognition and genetic network inference, there is a coupling between the models, and there is interest in learning this global-sparsity structure.

In the biological context, this global-sparsity structure corresponds to learning the topology of the network. Biological networks (and many engineering systems) are known to be quite sparse and have a hierarchical organization with little feedback between different hierarchical levels. For the system identification of biological networks and the resultant target identification problem, learning the sparsity structure and associated topology of the network is extremely important. It is particularly important because not only can it help improve the quality of identified models, but as will be discussed in the second part of this thesis: The topology of the network can be used to intelligently select pharmaceutical drug targets.

As an application of learning global-sparsity structure, consider the problem of variable selection in local linear regression. Local linear regression is a nonparametric approach which does regression in small regions of the predictors (see Figure 7.1), because the response looks linear in small regions [169, 126]. One can do variable selection for each of these blocks of predictors using lasso-type regularization or exploiting properties of local linear regression [126].

However, there is a coupling between the regression coefficients of each block of the predictor space. One expects that globally, over all chunks of predictor space, there is a common sparsity structure. This corresponds to a situation in which a set of predictors are globally unrelated to the response variable. This is despite the fact that locally, in each individual block, the sparsity structure might deviate by being a superset of the global-sparsity structure.

It has been recognized that one can gain improvements in learning this global-sparsity structure by posing the problem as a multi-task learning problem [202, 221, 155, 8]. Unfortunately, these techniques are computationally expensive. One class of techniques require solving non-convex optimization problems [202, 221]. A set of convex optimization based approaches [155, 8] are essentially variations of group-lasso-type regularization [220] for calculating the regression coefficients $\beta(u)$.

These group-lasso approaches [155, 8] are computationally difficult when there are a large number of coupled tasks. They require the solution of a linear regression problem with group-lasso-type regularization in which the matrix has m^2p^2 entries, where m is the number of tasks and p

is the number of features. These problems involve sparse matrices, but they are still difficult to solve because of the coupling between the different problems [220]. Coordinate descent approaches [220, 74] and the algorithms of [155, 8] seem to be susceptible to a growth in complexity by a factor of m^2.

In light of these difficulties, two new approaches to learning the global-sparsity structure are proposed in this chapter [13]. The first approach is to solve a regression problem which can be thought of as an average of the regression problems at $u_1, \ldots, u_m$, with a lasso-type regularization term to induce sparsity. The second approach is to solve a regression problem which can be thought of as an average of the square of the regression problems at $u_1, \ldots, u_m$, with a group-lasso-type regularization term to induce sparsity. The first approach makes more assumptions about the distribution of $\beta(u)$ and is computationally easier, when compared to the second approach.

First, the global-sparsity problem in the context of linear regression is formally defined. The two approaches are posed as optimization problems, and the problems are made tractable by making convex relaxations. Algorithms are given which can efficiently solve these convex optimization problems, and these algorithms are used to do variable selection for simulated data taken from a nonlinear function.

7.1 Global-Sparsity Structure Problem

Consider a collection of models $\mathcal{A}$, that are indexed by $u \in \mathcal{A}$. Furthermore, suppose that u is distributed with mixed density $f_U(u)$. Then, for each $u \in \mathcal{A}$, there is a linear model

$$y(u) = x(u)'\beta(u) + \epsilon(u), \tag{7.1}$$

where $\epsilon(u), y(u) \in \mathbb{R}$, $x(u), \beta(u) \in \mathbb{R}^p$, and $\epsilon(u)$ is a zero-mean random variable. In the example of piecing together local linear regressions: The u are points in a metric space, and the separate linear models for each u correspond to different local linear regressions.

Assume that covariances are finite. The noise variance $\sigma^2(u)$, covariance matrix $C(u)$, and the cross-covariance matrix $R(u)$ are given by:

$$\sigma^2(u) = \mathbb{E}(\epsilon^2) \tag{7.2}$$

$$C(u) = \mathbb{E}\Big((x(u) - \overline{x}(u))'(x(u) - \overline{x}(u))\Big) \tag{7.3}$$

$$R(u) = \mathbb{E}\Big(y(u)(x(u) - \overline{x}(u))\Big), \tag{7.4}$$

where vectors are taken to be column-vectors. Without loss of generality, assume that $C(u) \succ 0$ is positive definite. If this is not the case, then Tikhonov-type regularization [105, 223] can be used to ensure this rank condition holds. For local linear regressions, the $C(u)$ is a local covariance matrix that can vary as u changes.

T sparsity structure of a model is defined as a vector whose i-th entry is: 0, if the i-th predictor is always irrelevant; and 1, if the i-th predictor is potentially relevant. Also, assume that the regression coefficients obey a local-sparsity structure $K(u) \in \mathbb{R}^p$, that is

$$K_i(u) = 1(\beta_i(u) \neq 0). \tag{7.5}$$

Additionally, assume that the regression coefficients obey a global-sparsity structure $K \in \mathbb{R}^p$, that is

$$K_i = 1(\exists S_i : \mu(S_i) > 0 \wedge \beta_i(u) \neq 0, \forall u \in S_i), \tag{7.6}$$

where $\mu(S_i)$ denotes the measure of set S_i. For local linear regressions, there is additional structure which couples the $\beta(u)$, and this is not captured by the global-sparsity structure K. Though the global-sparsity structure is adequate for the purposes of variable selection, it might be possible to get better performance by fully considering this additional structure.

Global-Sparsity Structure Problem: Consider the collection of models $\mathcal{A}$ defined in Section 7.1. Suppose that the following quantities have been calculated: the sample covariance matrix $\hat{C}(u)$ and the sample cross-covariance matrix $\hat{R}(u)$ at points $u = u_1, \ldots, u_m$, where the u_i are independent and identically distributed with mixed density $f_U(u)$. The problem is: Learn the global-sparsity structure K.

7.2 Problem Solutions

Two approaches to solving this problem are given, and these approaches can be posed as optimization problems. The first approach makes additional assumptions on the distribution of $\beta(u)$, but is considerably simpler from a computational standpoint when compared to the second approach. The second approach does not make additional assumptions on the distribution of $\beta(u)$, but it requires a relaxation of the optimization problem to make it tractable.

7.2.1 Averaged Lasso

The idea is to calculate the average of the regression vectors, meaning calculate

$$\overline{\beta} = \mathbb{E}(\beta(u)). \tag{7.7}$$

The intuition behind this is that if $K_i = 0$, then $\overline{\beta}_i = 0$; whereas, if $K_i \neq 0$, then it is expected that $\overline{\beta}_i \neq 0$. Assuming that $K_i = 0$ if and only if $\overline{\beta}_i = 0$, then the intuition becomes exact. This additional assumption is need because, in general, a particular coordinate of the regression vector might have a non-zero distribution but have a zero-mean.

Recall that under the additional assumption, solving for K is equivalent to solving for $\overline{\beta}$. So one approach would be to solve for $\hat{\beta}(u)$ at $u = u_1, \ldots, u_m$ and then average them over u. The difficulty with this approach is that the average will typically not be sparse. Sparsity can be induced by thresholding the average, using cross-validation to choose the threshold level. It can be shown that this estimator of $\overline{\beta}$ is both sign- and norm-consistent.

However, it is known that when doing regression problems with sparse regression vectors, using lasso-type regularization can provide better performance than simply thresholding [199, 120, 223, 222]. This can be used to construct a better estimator of $\overline{\beta}$. Begin by solving the normal equations

$$\hat{\beta}(u) = \arg\min_{\beta} \|\hat{C}(u)\beta - \hat{R}(u)\|^2, \tag{7.8}$$

where $\|\cdot\|$ is the standard ℓ_2 norm. It is well-known that (7.8) has an analytic solution:

$$\hat{\beta}(u) = (\hat{C}(u))^{-1}\hat{R}(u). \tag{7.9}$$

Next, consider the following problem

$$\beta^{ave} = \arg\min_{\beta} \left\| \frac{1}{m}\sum_{i=1}^{m} \hat{C}(u_i)\beta - \frac{1}{m^2}\sum_{i=1}^{m}\sum_{j=1}^{m} \hat{C}(u_i)\hat{\beta}(u_j) \right\|^2. \tag{7.10}$$

By construction, the solution to (7.10) is given by

$$\beta^{ave} = \frac{1}{m}\sum_{i=1}^{m}\hat{\beta}(u_i), \tag{7.11}$$

and make the following identifications

$$C^{ave} = \frac{1}{m}\sum_{i=1}^{m}\hat{C}(u_i) \tag{7.12}$$

$$R^{ave} = \frac{1}{m^2}\sum_{i=1}^{m}\sum_{j=1}^{m}\hat{C}(u_i)\hat{\beta}(u_j) \tag{7.13}$$

$$= C^{ave}\beta^{ave}. \tag{7.14}$$

Thus, (7.10) represents a regression problem whose covariance matrix C^{ave} is an average of the sample covariance matrices. The cross-covariance matrix R^{ave} of (7.10) is simply the product of C^{ave} and β^{ave}. More importantly, it is useful to think of (7.10) as an averaged regression problem.

By posing the problem of estimating $\overline{\beta}$ as a regression problem, lasso-type regularization can be introduced to induce sparsity in the solution:

$$\beta^{lasso} = \arg\min_{\beta}\left\|C^{ave}\beta - R^{ave}\right\|^2 + \lambda\sum_{i=1}^{p} w_i|\beta_i|, \tag{7.15}$$

where $w_i = |\beta_i^{ave}|^{-\gamma}$ are weights [222], $\lambda \geq 0$, and $\gamma > 0$. Then, $\hat{K}_i = 1(\beta^{lasso} \neq 0)$.

7.2.2 Square-Averaged Lasso

Suppose that the assumption—$K_i = 0$ if and only if $\overline{\beta}_i = 0$—does not hold. Then, the averaged Lasso approach (7.15) cannot be used. However, the problem can be solved in an alternative manner. Define the quantity

$$\overline{\Pi} = \mathbb{E}\Big(\mathbb{E}[\beta(u)(\beta(u))^T|u]\Big). \tag{7.16}$$

The intuition for the importance of this term is that if $\overline{\Pi}_{ii} = 0$, then by definition of expectation: $\beta_i(u) = 0$ almost everywhere and $K_i = 0$; alternatively, if $\overline{\Pi}_{ii} > 0$, then $K_i = 1$.

Note that (7.8) is equivalent to solving the linear equation: $\hat{C}(u)\hat{\beta}(u) = \hat{R}(u)$. This linear equation can be "squared" and rewritten, similar to the idea of the lifting procedure [137, 51], as a quadratic equation

$$\hat{C}(u)\hat{\beta}(u)(\hat{\beta}(u))^T(\hat{C}(u))^T = \hat{R}(u)(\hat{R}(u))^T. \tag{7.17}$$

This can be rewritten as a semidefinite program (SDP) with nonconvex constraints:

$$\begin{aligned}\hat{\Pi}(u) = \arg\min_{\Pi} \quad & \left\|\hat{C}(u)\Pi(\hat{C}(u))^T - \hat{R}(u)(\hat{R}(u))^T\right\|^2 \\ \text{subject to:} \quad & \Pi \succeq 0 \\ & \text{rank}(\Pi) = 1,\end{aligned} \tag{7.18}$$

where $\|\cdot\|$ is the Frobenius norm. These constraints are equivalent to saying that there exists $\hat{\beta}(u)$ such that $\hat{\Pi}(u) = \hat{\beta}(u)(\hat{\beta}(u))^T$. It is important to note that (7.17) is a linear equation in Π, and it has a unique, analytic solution [198]:

$$\text{vec}(\Pi) = (\hat{C}(u) \otimes \hat{C}(u))^{-1}\text{vec}(\hat{R}(u)), \tag{7.19}$$

where $\otimes$ denotes the Kronecker product. It is interesting to note that this linear problem is nearly identical to the discrete-time Lyapunov equation [41], which is also a linear problem.

Whereas (7.15) was the average of linear equations, similarly consider an average of quadratic equations with

$$\begin{aligned} \Pi^{ave} = \arg\min_{\Pi} \quad & \left\| C^{ave}\Pi(C^{ave})^T - C^{ave}\Pi^{ave}(C^{ave})^T \right\|^2 \\ \text{subject to:} \quad & \Pi \succeq 0 \\ & \Pi = \sum_{i=1}^{p} \Pi_i \\ & \text{rank}(\Pi_i) = 1, \quad \text{for } i = 1, \ldots, p. \end{aligned} \tag{7.20}$$

By construction, $\Pi^{ave} = \frac{1}{m}\sum_{i=1}^{m} \hat{\Pi}(u_i)$. This optimization problem is nonconvex, but a convex relaxation can be made by dropping the rank constraint [51]. Since the goal is to induce sparsity, solve the following problem:

$$\begin{aligned} \Pi^{sql} = \arg\min_{\Pi} \quad & \left\| C^{ave}(\Pi - \Pi^{ave})(C^{ave})^T \right\|^2 + \lambda \sum_{i=1}^{p} w_i \sqrt{S(i)} \\ \text{subject to:} \quad & \Pi \succeq 0 \\ & S(i) = \sum_{j=1}^{p} \left((\Pi_{ji})^2 + (\Pi_{ij})^2 \right), \end{aligned} \tag{7.21}$$

where

$$w_i = \left| \sum_{j=1}^{p} \left((\Pi_{ji}^{ave})^2 + (\Pi_{ij}^{ave})^2 \right) \right|^{-\gamma} \tag{7.22}$$

are weights [211], $\gamma > 0$, and $\lambda \geq 0$. Then, $\hat{K}_i = 1(\Pi_{ii}^{sql} \neq 0)$.

The summation term is actually a group-lasso-type regularization term. The reason for using group-lasso is that there is a relationship between the different variables in the matrix Π. If $\Pi_{ii} = 0$, then $\beta_i(u) = 0$ almost everywhere. Therefore,

$$\Pi_{ij} = \Pi_{ji} = \mathbb{E}(\beta_i(u)\beta_j(u)) = 0, \tag{7.23}$$

where $j = 1, \ldots, p$. This is a situation for which group-lasso is well-suited. One can compare this use of group-lasso to the use of lasso in a similar situation [51]. The lasso is computationally simpler, but does not preserve the full structure of the situation.

7.3 Consistency Results

There are two levels of measurements that are made. The first measurement is a value of $u = u_1, \ldots, u_m$, for $u_i \in \mathcal{A}$. The second level of measurement is that for each value of u_i, a total of E_i measurements of y and x are used to calculate the value of the sample covariance and cross-covariance matrix. Thus, a total of $n = \sum_i E_i$ measurements are made.

Consequently, computing expectations has to average given u_i and then average over u. Averaging over u means

$$\overline{\Xi}^{ave} = \frac{1}{m} \sum_{i=1}^{m} \Xi(u_i), \tag{7.24}$$

for $\Xi = C, \beta, \Pi$, and $\overline{R}^{ave} = \overline{C}^{ave} \overline{\beta}^{ave}$. Additionally, define the limits as: $\overline{C} = \mathbb{E}(C(u))$ and $\overline{R} = \mathbb{E}(C(u))\mathbb{E}(\beta(u))$, and recall the definitions of $\overline{\beta}$ and $\overline{\Pi}$.

Averaging given u_i means that for increasing E_i:

$$\|\hat{C}(u_i) - C(u_i)\| = O_p(a_n) \tag{7.25}$$

$$\|\hat{R}(u_i) - R(u_i)\| = O_p(a_n), \tag{7.26}$$

for all $u_i \in \mathcal{A}$, where $a_n = o_p(1)$. This operational definition for average given u_i allows flexibility to consider different techniques for computing the sample (cross-)covariance matrix, such as the standard approach or a sparsity-inducing approach applicable to "large p, small n" scenarios [224, 51, 32]. Furthermore, assume that the linear-least squares estimate is consistent:

$$\|\hat{\beta}(u_i) - \beta(u_i)\| = O_p(a_n). \tag{7.27}$$

This has been proved under many situations.

This setup of making two levels of measurements can be used to prove both norm- and sign-consistency. To simultaneously achieve both types of consistency, use a two-step regression procedure [146]. Adaptive lasso and adaptive group-lasso [222, 211] are such a two-step procedure that seems to work well in practice. To prove consistency, it is first shown that the linear models of the averaged equations are a consistent estimator of the averaged regression vectors. Then, the triangle inequality and existing theorems on the consistency of adaptive lasso and group-lasso can be applied. The theorems are given below, and proofs can be found in the appendix.

Theorem 7.3.1 (Consistency of Averaged Lasso)**.** Recall the assumptions of Sections 7.1, 7.2.1, and 7.3. The estimator given in (7.15) is norm-consistent: $\beta^{lasso} \xrightarrow{p} \overline{\beta}$ and sign-consistent: $\mathbb{P}(\text{sign}(\hat{K}_i) = \text{sign}(K_i)) \to 1$.

Theorem 7.3.2 (Consistency of Square-Averaged Lasso)**.** Recall the assumptions of Sections 7.1 and 7.3. The estimator given in (7.21) is norm-consistent: $\Pi^{sql} \xrightarrow{p} \overline{\Pi}$ and sign-consistent: $\mathbb{P}(\text{sign}(\hat{K}_i) = \text{sign}(K_i)) \to 1$.

7.4 Numerical Algorithms

Implementing the averaged lasso (7.15) is straight-forward, because the averaged lasso is simply a linear regression problem with lasso-type regularization. So, the averaged lasso can be implemented using any of the existing techniques to do lasso regression [62, 74]. These techniques are known to be computationally efficient.

Techniques for lasso regression can be used to solve the square-averaged lasso (7.21), if additional relaxations are made to the problem. By removing the constraint $\Pi \succeq 0$ and replacing the group-lasso-type term with the lasso-type term $1^T|\Pi|1$ [51], the problem is equivalent to a lasso regression. So, the relaxation can be solved fairly easily.

The unmodified, square-averaged lasso (7.21) is somewhat more difficult to implement. Coordinate descent cannot be used because coordinate descent algorithms are not guaranteed to converge in the situation [74] where the variables Π_{ij} appear in multiple group-lasso-type regularization terms.

To solve the unmodified, square-averaged lasso problem (7.21), consider the related problem,

$$\begin{aligned} \Pi^{sql} = \arg\min_{\Pi} \quad & \left\| C^{ave}\Pi(C^{ave})^T - R^{ave}(R^{ave})^T \right\|^2 \\ \text{subject to:} \quad & \Pi \succeq 0 \\ & \sum_{i=1}^{p} w_i \sqrt{S(i)} \leq \mu \\ & S(i) = \sum_{j=1}^{p} \left((\Pi_{ji})^2 + (\Pi_{ij})^2 \right), \end{aligned} \tag{7.28}$$

where w_i, γ are as defined for (7.21), and $\mu > 0$. As shown in Section 2.3.g of [4], the constraints involving the summations can be rewritten as linear and conic inequalities. Thus, we can solve this problem using existing semi-definite programming approaches. However, the solver must be carefully chosen because this reformulation introduces on the order of p^2 constraints. Interior-point solvers might have difficulties with the problem size, and so one might consider using a bundle method [99], though these often convergence slowly.

There is another way to solve the rewritten form of the square-averaged lasso problem (7.28). Dropping the constraint $\Pi \succeq 0$ leads to a much simpler problem that can be solved using a Majorization-Minimization (MM) algorithm [110]. These algorithms are first-order methods and have very simple iterations. Though they require more iterations to converge to the solution than second-order methods, the increased simplicity of each iteration leads to performance improvements for large problems such as (7.28).

7.5 Numerical Experiments

These approaches are used to solve the variable selection problem, which can be reformulated as a global-sparsity problem. To make the variable selection problem interesting, here the focus is on response variables which are nonlinear functions of the predictor variables. Variable selection for linear functions is well-studied, and the global-sparsity approach would not perform better.

Consider the predictor variables: $X_k \sim \text{Uniform}[0, 2\pi]$, for $k = 1, \ldots, 11$. Define the response variable

$$Y = \sum_{k=1}^{5} \frac{5\sin(X_k)}{k} + \epsilon, \tag{7.29}$$

where $\epsilon \sim \mathcal{N}(0, 0.1)$. The variable selection problem is: Given measurements of X_i and Y, choose the X_i upon which Y (or more accurately $\mathbb{E}[Y|X]$) depends.

As shown in Figure 7.1: Doing local linear regressions [169, 126] at several points in the support of the X_k's, the variable selection problem is equivalent to learning the global-sparsity structure of the local linear regressions. There is a subtle issue regarding whether the regression problems for different blocks of the predictor are independent. Of course, in the finite sample regime, there will be some dependence between the regression problems. However, this variable selection procedure can be theoretically justified by noting that in the limit of increasing data points, the windowing separates the regression problems and makes them independent.

One-hundred sample points were generated, and 22 local linear regressions were performed at randomly chosen points within the range of the predictors. The tunable parameters for the local linear regressions were chosen using V-fold cross-validation [32]. Then, the averaged lasso (7.15) and square-averaged lasso (7.21) were used to learn the global sparsity structure. The $\beta(u_i)$ used in (7.15) and (7.21) were calculated without the use of lasso-type regularization; it may be possible to improve the performance of the global-sparsity learning by using lasso-regularized values of $\beta(u_i)$. Fixing $\gamma = 5$, the remaining tunable parameters were chosen using V-fold cross-validation with empirical loss function defined as:

$$\ell = \frac{1}{N} \sum_{\nu=1}^{N} \|\beta_{train} - \beta^{ave}_{validate}\|. \tag{7.30}$$

The results of 50 trials using the averaged lasso and square-averaged lasso are shown in Figures 7.2c and 7.2d, respectively. Pure black corresponds to not selecting the variable 100% of the time, whereas pure white corresponds selecting the variable selected 100% of the time. Shades of gray indicate intermediate rates of selection. Also, note that β_0 refers to the value $\mathbb{E}[Y|X]$; it is an artifact of local linear regression [169, 126].

It is instructive to examine the results of one trial in detail, and a typical example is shown in Figure 7.3. Two methods were used to compute β^{ave}: using local linear regression and local linear regression with lasso-type regularization. Thresholding the β^{ave} values, using the empirical loss function given in (7.30), yields the results shown in Figures 7.3c and 7.3d. These results are not as good as those produced by the averaged lasso or the square-averaged lasso, shown in Figures 7.3e and 7.3f. Part of this discrepancy is due to the empirical loss function: It would be fairer to choose the thresholding levels by cross-validating against prediction error in each block of predictors, but this requires significant computational effort. The global-sparsity structure methods use a computationally simpler cross-validation scheme that seems to give good results.

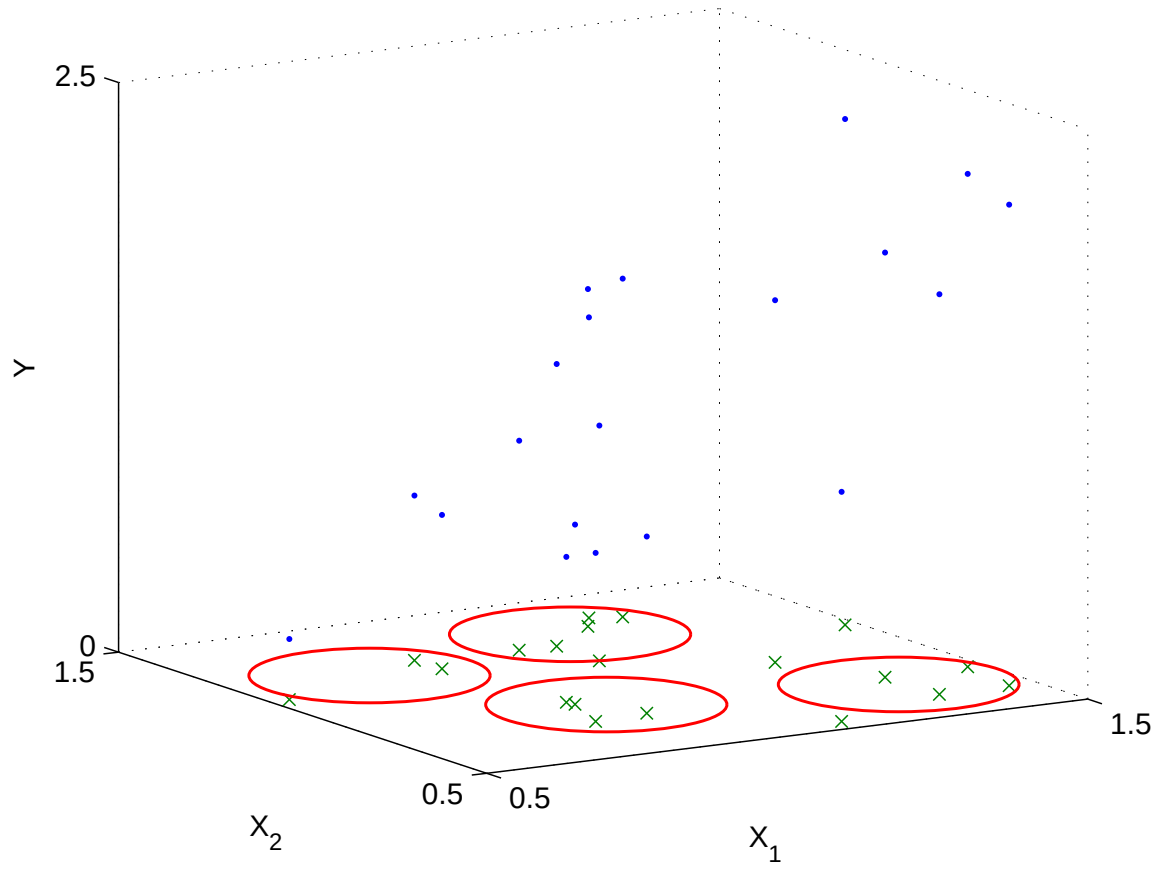
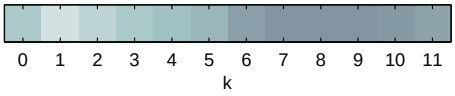
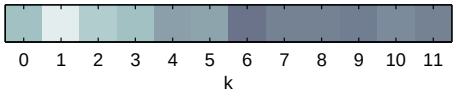


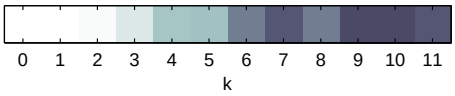
Figure 7.1: One expects that the regression vectors computed using local linear regression at the centers of the balls—in which the 'x' markers indicate predictor values and the '.' markers indicate corresponding response values—obey a global-sparsity structure corresponding to the relevant and irrelevant predictors. Consequently, the problem of variable selection for a nonlinear function can be solved by learning the global-sparsity structure of the regression vectors.



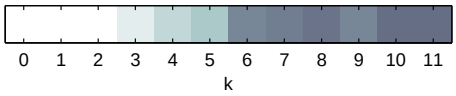
(a) $\hat{K}_k$ from thresholding regular β^{ave}



(b) $\hat{K}_k$ from thresholding lasso β^{ave}

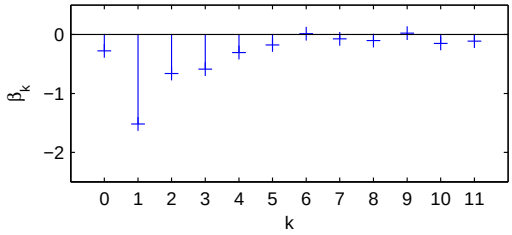


(c) $\hat{K}_k$ from averaged lasso

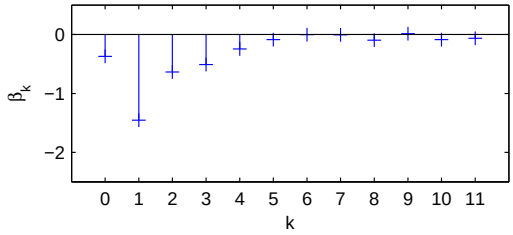


(d) $\hat{K}_k$ from square-averaged lasso

Figure 7.2: We calculated $\hat{K}$ using different approaches over 50 trials and averaged the results. Note that "regular β^{ave}" means that the values of $\hat{\beta}(u_i)$ were calculated using regular local linear regression, whereas "lasso β^{ave}" means that the values of $\hat{\beta}(u_i)$ were calculated using a local linear regression with lasso-type regularization.

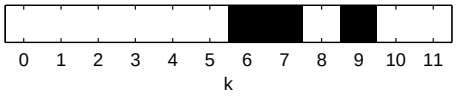


(a) Values of regular β_k^{ave}



(b) Values of lasso β_k^{ave}

(c) $\hat{K}_k$ from thresholding regular β^{ave}



(d) $\hat{K}_k$ from thresholding lasso β^{ave}

(e) $\hat{K}_k$ from averaged lasso

(f) $\hat{K}_k$ from square-averaged lasso

Figure 7.3: For a typical trial, we calculated $\hat{K}$ using different approaches. Note that "regular β^{ave}" means that the values of $\hat{\beta}(u_i)$ were calculated using regular local linear regression, whereas "lasso β^{ave}" means that the values of $\hat{\beta}(u_i)$ were calculated using a local linear regression with lasso-type regularization.

Part II

Drug Multi-Target Selection

Chapter 8

Topology-Based Control

Control theory has traditionally focused on a core group of goals: to stabilize a plant, to improve plant performance, to robustify a plant, to track a reference, or to perform motion planning. In engineering systems, these goals have been achieved through an analysis-design flow; this flow is rarely linear—there is often a need to go back to previous steps and incorporate things that were missed in earlier attempts. Beginning with design specifications, we write mathematical models for the engineering system, analyze these models, and devise a controller. We also implement the controller on actual hardware.

The traditional control scheme has been to input a signal into a plant, using either an open-loop or a closed-loop controller. Such a control strategy is possible if the plant is able to accept inputs or can be modified to do so. However, this situation is not always true in biological genetic networks; in these systems, there is often no input or obvious modification to allow inputs. Instead of inputs, genetic networks are more easily influenced through large-scale modifications. Genetic networks are different from traditional engineering systems and require a new paradigm for control.

It is often easier to change the topology of a genetic network than it is to either change the states or elements of the network. For instance, a state could be the concentration of a protein within a cell, something which is difficult to affect to within any order of precision. Additionally, it is sometimes difficult or not feasible to modify or insert pathways by adding elements [168, 164, 56]. Thus, for genetic networks it is important to develop a theory of control based on making large-scale changes (e.g. genetic changes or pharmaceutical drugs) to the topology of the genetic network. Fundamentally, medical treatments seek to change how a cell operates and go beyond modifying the cellular environment.

Genetic networks can be modified in a variety of ways. The most basic is the use of pharmaceutical drugs, many of which prevent certain reactions from occurring or remove a state from a network. Biotechnology techniques allow for the insertion of genetic material into bacteria, and are commonly used for alternative energy and pharmaceutical applications [85, 197]. In another technique, the genetic material of a virus is replaced with useful, genetic material. Next, the host is infected with the virus, and this inserts the useful, genetic material into the host. This control technique is being studied for use in pharmaceutical applications such as cystic fibrosis [85, 197]. Biologists continue to develop new techniques, amongst which include the use of microRNA and single interfering RNA.

Though many of these techniques are established and used in practice, there is a lack of a systematic theory or methodology to determine which modifications to make or what to target with

pharmaceutical drugs. Biological research often involves the use of intuition or trial-and-error to determine which changes are or are not beneficial for the purposes of controlling a biological system. This chapter proposes the idea of abstracting the effect of pharmaceutical drugs as modifying the topology of the biological network, and it also proposes how this abstraction might be used to do control by identifying drug targets [19].

Piecewise-affine (PWA) hybrid systems and ordinary differential equation (ODE) models of biological systems are considered in this chapter. Two different types of models are used for reasons of analysis: The simpler, hybrid systems models are easier to analyze for global behavior, and the more detailed, ODE models are easier to analyze for local behavior of small components of the network. Results related to hybrid systems theory are discussed, controllers using ODE theory are defined and analyzed, and these approaches are used to analyze and build a controller for the p53 pathway—a pathway that is related to cancer.

This topological control changes the topology of the network by applying a pharmaceutical drug or other chemical, and the topology remains changed only in the presence of this pharmaceutical. As soon as it degrades away, the topology of the network goes back to an uncontrolled, unchanged state. Since the control is topological, it is crucial to have a correctly identified network. The approach described is unable to deal with latent variables that are unidentified, because the presence of latent variables can drastically change the behavior of the system.

8.1 Preliminaries

The PWA hybrid systems considered here have rectangular guards, and are a simplification of general hybrid systems [140, 201]. In order to define them, consider the following preliminary definitions. Define some operations on sets; formal definitions can be found in [103]. Denote the boundary of set $\mathcal{S}$ as $\partial\mathcal{S}$. The closure of a set is the union of a set and its boundary, and the closure of $\mathcal{S}$ is denoted $\overline{\mathcal{S}}$. The interior of set $\mathcal{S}$ is denoted $\text{int}(\mathcal{S})$.

Define a hyperrectangle as

$$\mathcal{C} = \{x : l_i < x_i < u_i, \forall i \in \{1, \ldots, n\}\}, \tag{8.1}$$

where l_i, u_i are constants, n is the dimension of the state-space, and x_i denotes the i-th component of x. Similarly, define a hyperedge as

$$\mathcal{E} = \{x : l_i < x_i < u_i, \forall i \in \mathcal{I} \wedge x_j = \gamma_j, \forall j \in \{1, \ldots, n\} \setminus \mathcal{I}\}, \tag{8.2}$$

where γ_j is a constant and $\mathcal{I} \subseteq \{1, \ldots, n\}$ is a set of indices. Let e_i refer to the standard basis vector in $\mathbb{R}^n$. If $\dim(\mathcal{E}) = n - 1$, then the normal to $\mathcal{E}$ is the vector e_j, where $j \notin \mathcal{I}$.

By construction, the hyperrectangle is an embedded submanifold of dimension n and the hyperedge is an embedded submanifold of dimension $|\mathcal{I}|$. For the q-th hyperrectangle $\mathcal{C}^q$, associate a total of $(3^n - 1)$ hyperedges ${\mathcal{E}^q}_k$, by defining l_i, u_i, γ_j, and $\mathcal{I}$ for each k—such that $\bigcup_{i=1}^{3^n-1} {\mathcal{E}^q}_i = \partial\mathcal{C}^q$ and ${\mathcal{E}^q}_i \cap {\mathcal{E}^q}_j = \emptyset$ for $i \neq j$. The set ${\mathcal{E}^q}_k$ is the k-th hyperedge of the q-th hyperrectangle. Examples in $\mathbb{R}^2$ of these sets are shown in Figure 8.1.

8.1.1 Definition of PWA Hybrid System

Define a PWA system on a domain $\mathcal{D}$ as a collection of hyperrectangles $\mathcal{C}^i$, with $\dot{x} = A^i x + b^i$ for $x \in \mathcal{C}^i$, such that $\mathcal{D} = \text{int}(\bigcup_{i>0} \overline{\mathcal{C}^i})$ and $\mathcal{C}^i \cap \mathcal{C}^j = \emptyset$ for $i \neq j$. A trajectory of this system is a

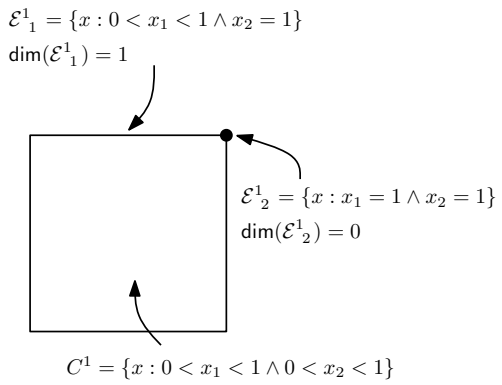


Figure 8.1: A simple example of a hypercube and its associated hyperedges in $\mathbb{R}^2$ is shown. Only two of the associated hyperedges are labeled; there are eight hyperedges associated to C^1.

solution of the vector field $f(x)$ in the sense of Filippov [71], where $f(x) = A^i x + b^i$ if $x \in \mathcal{C}^i$ and is undefined otherwise. Specifically, a trajectory of this system with initial condition $x \in \mathcal{D}$ is given by an absolutely continuous function $\psi_t(x) \in \mathcal{D}$ such that

$$\frac{d\psi_t(x)}{dt} \in \bigcap_{\delta>0} \bigcap_{\mu(N)=0} \overline{\text{co}}(f(\mathcal{B}(x,\delta) \setminus N)) \tag{8.3}$$

almost everywhere, where $\mathcal{B}(x,\delta) = \{y : \|x-y\|_2{}^2 < \delta\}$ and the intersection is taken over all sets N with measure zero. For initial condition $x \in \mathcal{D}$, define $\mathcal{T}_x = [0, t_f)$ as the maximal interval such that $\psi_t(x) \in \mathcal{D}$. Note that t_f can be $+\infty$. And, t_f can be interpreted as the escape time at which $\psi_{t_f}(x) \notin \mathcal{D}$.

8.1.2 Trajectory Cycles

The notion of trajectory cycles will be defined to describe the type of trajectories possible in a system. Intuitively, a *forward trajectory cycle* is defined as a forward trajectory of the continuous states of the hybrid system, such that the trajectory makes an infinite number of visits to a particular hyperrectangle. Similarly, a *backwards trajectory cycle* is defined as a backwards trajectory of the continuous states, such that the trajectory makes an infinite number of visits to a particular hyperrectangle. Note that implicit in both intuitive definitions is the inclusion of Zeno behavior.

8.1.3 Promotion-Inhibition Networks

A promotion-inhibition network is a signed, directed graph $N = (V, E, S)$; where $V = \{v_1, \ldots, v_n\}$ is the set of vertices, $E \subseteq \{(u,v) : u, v \in V\}$ is the set of directed edges, and $S : E \to \{-1, +1\}$ is a function that gives the sign of an edge. For an edge $e = (u,v)$: u is the direct predecessor of v, and v is the direct successor of u. A *feedback loop* is a directed cycle $L = \{e_1 = (u_1, v_1), \ldots, e_m = (u_m, v_m)\}$, where $u_i = v_{i-1}$ for $i = 2, \ldots, m$; and $v_m = u_1$. A *negative feedback loop* is a feedback loop L such that $\prod_{i=1}^m S(e_i) = -1$. A *monotone loop* is an undirected cycle $M = \{e_1 = (u_1, v_1), \ldots, e_m = (u_m, v_m)\}$, where either $v_i = u_{i-1}$ or $u_i = v_{i-1}$ for $i = 2, \ldots, m$; and $v_m = u_1$ or $v_1 = u_m$. A *negative monotone loop* is a monotone loop such that $\prod_{i=1}^m S(e_i) = -1$.

8.1.4 Relation to Biological Genetic Networks

Genetic networks are often elucidated in the form of a promotion-inhibition network, and examples are shown in Figure 8.2. Intuitively, a positively (negatively) signed edge between two vertices means that an increase in the direct predecessor leads to an increase (decrease) in the direct successor, and biologists term this as promotion (inhibition). These networks do not describe the underlying biological mechanism of an edge, but this information will be important when designing controllers.

A PWA hybrid system model can be generated from a promotion-inhibition network, using the techniques of [54, 84, 185, 90, 11, 83, 29]. Similarly, an ordinary differential equation model can be generated [125, 187] from these networks. There are various advantages and disadvantages to the two types of models, and this is discussed this below.

8.2 Existence of Trajectory Cycles

An important class of results in the hybrid and monotone systems theories relates the topological structure of a system to the global behavior of trajectories of the system, independent of any coefficients in the system. One useful theorem concerns a PWA hybrid system derived from a promotion-inhibition network, in the manner of [54, 84, 185, 90, 11, 83, 29]. Under technical assumptions on the PWA hybrid system, if there are no negative feedback loops in the promotion-inhibition network, then there are no trajectory cycles [18].

Under the given conditions, these systems are stable and trajectories converge to an equilibrium point in a node-like manner; that is, trajectories qualitatively look like the trajectories of a stable, linear system with purely real eigenvalues. Moreover, the presence of negative feedback is a necessary condition for the presence of limit cycles, centers, and foci. The shortcoming of this theorem is that it does not apply to many systems with self-inhibition, and this is common in biological systems.

An similar result holds for ODE systems. Under technical conditions, if the promotion-inhibition network has no negative monotone loops, then the system is a monotone system [125, 7]. Consequently, the trajectories of the system converge to an equilibrium and there are no stable oscillations [102]. These theorems apply to systems with self-inhibition, but they are stricter because negative monotone loops are stricter than negative feedback loops.

8.3 Controllers

Through the use of results stated in Section 8.2, several different topology based controller schemes become apparent. The basic idea is to use operations, such as removing edges or vertices in the promotion-inhibition network, to change the topology of the genetic network. The genetic network needs to be forced into a situation such that it has no negative feedback. Because this will ensure that the system does not oscillate and has simple dynamics, it will be easier to move the system into a desired state. Such control is quite crude in relation to traditional control techniques, but it can be used to achieve useful results in certain situations.

It is important to keep in mind that the fundamental ideas of the controllers are contained within the basic topological examples given below. Though these are examples, the examples are general since the equations model two broad classes of reactions. Based on existing biological

techniques [85, 197], the controller examples that are given are hypothetically feasible. However, it is not always possible to implement the controller.

Despite this, the reason that the fundamental ideas are topological is that there are multiple, biological ways to achieve the described effects. For instance, a vertex removal can be accomplished via compounds or pharmaceuticals that bind to a protein and remove its function, instead of the antisense RNA example that is given [85]. The controller used to remove an edge or a vertex must correspond to the biological mechanism behind the edge or vertex, otherwise the controller will fail.

8.3.1 Inhibition Edge Removal Example

Gene therapy techniques can add genetic material, but they cannot remove genetic material. Competitive binding can be used to remove an inhibition edge. The basic idea for the control is to add a high number of copies of a gene and its promoter region to a cell. This negates the effect of any inhibitors.

Suppose that the promotion-inhibition network has an edge $e = (A, B)$ with label $S(e) = -1$, and the control enforces $[D_0] \gg [A_0]$, where the subscript 0 is used to denote initial concentrations and the square brackets $[\cdot]$ denote concentration. One model for this is the set of reactions

$$A + D \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} AD \tag{8.4}$$

$$D + E_i \underset{k_{-2,i}}{\overset{k_{2,i}}{\rightleftharpoons}} DE_i \xrightarrow{k_{4,i}} B + D + E_i \tag{8.5}$$

$$D + F_i \underset{k_{-3,i}}{\overset{k_{3,i}}{\rightleftharpoons}} DF_i, \tag{8.6}$$

where E_i are proteins which promote the production of protein B, D is the DNA which codes for protein B and has the accompanying promoter region, and F_i is a protein which binds to DNA D but does not begin transcription. Note that i indexes over multiple proteins and complexes. In these reactions, AD, DE_i, AP, and DF_i are intermediate complexes. The effect of the inhibitors A and F_i is to prevent the formation of the DE_i complex, that is either A or F_i cannot simultaneously bind with either E_i or D. Note that (8.5) describes the aggregate process of activators and enzymes producing a protein and (8.4) and (8.6) describe an inhibitor binding to DNA.

These reactions can be written as a set of fractal reaction equations [172, 173, 216] as

$$\frac{d[AD]}{dt} = k_1[A]^{\alpha_1}[D]^{\alpha_2} - k_{-1}[AD] \tag{8.7}$$

$$\frac{d[DE_i]}{dt} = k_{2,i}[D]^{\alpha_{3,i}}[E_i]^{\alpha_{4,i}} - k_{-2,i}[DE_i] \tag{8.8}$$

$$\frac{d[DF_i]}{dt} = k_{3,i}[D]^{\alpha_{5,i}}[F_i]^{\alpha_{6,i}} - k_{-3,i}[DF_i] \tag{8.9}$$

$$\frac{d[B]}{dt} = \sum_i k_{4,i}[DE_i]^{\alpha_{7,i}}, \tag{8.10}$$

with the following constraints:

$$[A_0] = [A] + [AD] \tag{8.11}$$

$$[D_0] = [D] + [AD] + \sum_i [DE_i] + \sum_i [DF_i] \tag{8.12}$$

$$[E_{i,0}] = [E_i] + [DE_i] \tag{8.13}$$

$$[F_{i,0}] = [F_i] + [FE_i]. \tag{8.14}$$

The constraints assume fixed initial concentrations. In these fractal equations, each k is a reaction rate, each α is an exponent that relates concentration to the speed of the reaction, and the brackets denote concentration.

From the positivity of concentrations and (8.11), it is clear that $0 \leq [AD] \leq [A_0]$, which implies that

$$[D_0] - [A_0] - [DE] \leq [D] \leq [D_0] - [DE]. \tag{8.15}$$

However, since $[D_0] \gg [A_0]$, it approximately holds

$$[D] = [D_0] - [DE]. \tag{8.16}$$

Consequently, if $[F_{i,0}] \sim [D_0]$ then the system is approximately the same as if reaction (8.4) did not occur. In the special case of $\alpha_{3,i} = \alpha_{5,i} = 1$, solving for the rate of production, using standard assumptions, gives

$$\frac{d[B]}{dt} = \sum_i k_{4,i} \left(\frac{k_{2,i}}{k_{-2,i}} ([D_0]/H)[E_i]^{\alpha_{4,i}} \right)^{\alpha_{7,i}}, \tag{8.17}$$

where

$$H = 1 + \sum_i \frac{k_{2,i}}{k_{-2,i}} [E]^{\alpha_{4,i}} + \sum_i \frac{k_{3,i}}{k_{-3,i}} [F]^{\alpha_{6,i}}. \tag{8.18}$$

Through the use of the controller, the inhibitory effect of A on B was eliminated. This effectively breaks the inhibition edge in the promotion-inhibition network.

8.3.2 Vertex Removal Example

In a vertex removal controller, a vertex is removed from the promotion-inhibition network. Antisense RNA can be used to prevent translation of protein through competitive binding with mRNA[85, 197]. The control is to add a high number of copies of the antisense RNA. This removes the effect of the vertex.

Suppose that the promotion-inhibition network has edge $e = (A, B)$ with label $S(e) = +1$, and the control enforces $[P_0] \gg K$, where $[mAP] \leq K$. One model for this is the set of reactions

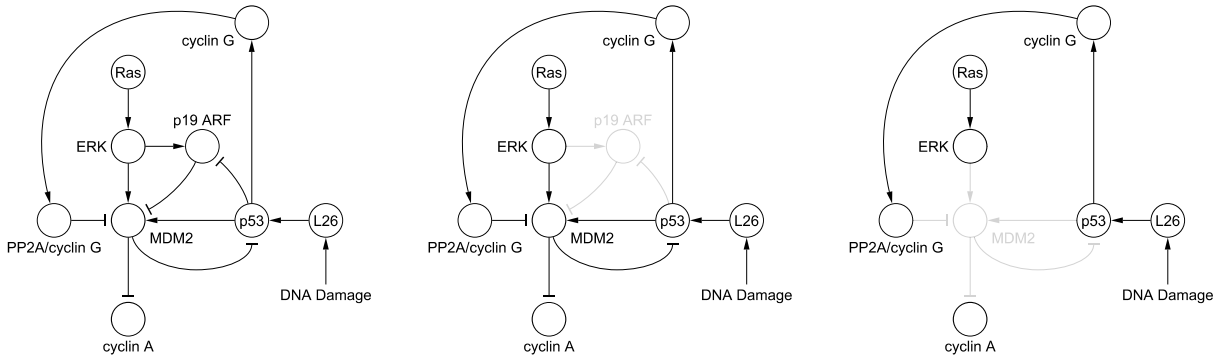
$$D + E_i \underset{k_{-1,i}}{\overset{k_{1,i}}{\rightleftharpoons}} DE_i \xrightarrow{k_{3,i}} DE_i + mA \tag{8.19}$$

$$mA \xrightarrow{k_4} mA + A \tag{8.20}$$

$$mA \xrightarrow{k_5} Z_1 \tag{8.21}$$

$$A \xrightarrow{k_6} Z_2 \tag{8.22}$$

$$mA + P \underset{k_{-2}}{\overset{k_2}{\rightleftharpoons}} mAP \tag{8.23}$$



(a) Normal p53 Pathway (b) Abnormal p53 Pathway (c) Abnormal p53 Pathway with Controller

Figure 8.2: When a portion of the normal p53 pathway [95, 167, 196, 130] becomes abnormal, such as loss of p19 ARF function [167], the system behaves unfavorably by underexpressing p53. Using a controller, certain edges and vertics of the pathway can be removed to make the system behavior more favorably.

where E_i are proteins which promote the production of protein A, D is the DNA which codes for protein A and has the accompanying promoter region, P is the control of added antisense RNA. Also, Z_1 and Z_2 are the aggregate products of degradation. Note that i indexes over multiple proteins and complexes. In these reactions, mA is the translated mRNA for protein A, DE_i is an intermediate complex, and mAP is the complex of mRNA bound with the antisense RNA. Note that (8.19) describes the aggregate process of activators and enzymes translating the DNA for a protein into mRNA, (8.20) describes the aggregate process of translation of mRNA into protein, and (8.23) describes the binding of a mRNA with the added antisense RNA. Since P is the antisense RNA of the mRNA, this prevents translation of the mRNA into protein when P is bound to mA. Also, (8.21) and (8.22) describe the degradation of mA and A, respectively.

These reactions can be written as a set of fractal reaction equations [172, 173, 216] as

$$\frac{d[DE_i]}{dt} = k_{1,i}[D]^{\alpha_{1,i}}[E]^{\alpha_{2,i}} - k_{-1,i}[DE_i] \tag{8.24}$$

$$\frac{d[mA]}{dt} = k_{3,i}[DE_i]^{\alpha_{3,i}} - k_4[mA] + \\ - k_2[mA]^{\alpha_5}[P]^{\alpha_6} + k_{-2}[mAP] \tag{8.25}$$

$$\frac{d[A]}{dt} = k_5[mA]^{\alpha_4} - k_6[A] \tag{8.26}$$

$$\frac{d[mAP]}{dt} = k_2[mA]^{\alpha_5}[P]^{\alpha_6} - k_{-2}[mAP] \tag{8.27}$$

with the following constraints:

$$[D_0] = [D] + \sum_i [DE_i] \tag{8.28}$$

$$[E_{i,0}] = [E_i] + [DE_i] \tag{8.29}$$

$$[P_0] = [P] + [mAP] \tag{8.30}$$

$$[DE_i] \leq K_i \tag{8.31}$$

$$[mA] \leq K. \tag{8.32}$$

The last two inequalities come about through standard arguments involving nullclines. In these fractal equations, each k is a reaction rate, each α is an exponent that relates concentration to the speed of the reaction, and the brackets denote concentration.

Typically, the reversible reactions are much faster than the irreversible reaction, that is the reactions corresponding to $k_{1,i}, k_2, k_{-1,i}, k_2$ are much faster than those corresponding to $k_{3,i}, k_4, k_5$ [172]. Under this assumption, the quasi-steady state assumption can be applied to get

$$[DE_i] = \frac{k_1}{k_{-1}}[D]^{\alpha_{1,i}}[E_i]^{\alpha_{2,i}} \tag{8.33}$$

$$[mAP] = \frac{k_2}{k_{-2}}[mA]^{\alpha_5}[P]^{\alpha_6}. \tag{8.34}$$

Combining (8.28) and (8.34) gives

$$[mA] = \left(\frac{k_{-2}[mAP]}{k_2[P]^{\alpha_6}}\right)^{1/\alpha_5} \tag{8.35}$$

$$\leq \left(\frac{k_{-2}K}{k_2([P_0] - [mAP])^{\alpha_6}}\right)^{1/\alpha_5}. \tag{8.36}$$

Because the controller enforces that $[P_0] \gg K$, it roughly means that $[mA] = 0$. Consequently, it approximately holds that

$$\frac{d[A]}{dt} = k_5[mA]^{\alpha_4} - k_6[A] \leq 0. \tag{8.37}$$

Through the use of the controller, the concentration of protein A was reduced to zero. Thus, vertex A we effectively removed from the promotion-inhibition network.

8.4 p53 Pathway

The p53 protein is an important tumor suppressor, which reacts to stress signals and induces an appropriate cellular response [95, 80, 10, 208]. These stress signals include DNA damage, heat shock, cold shock, and spindle damage. These stress signals lead to a post-translational modification of p53, causing the p53 to trigger downstream pathways involved with cell cycle arrest, cell senescence, or apoptosis [95]. The inactivation of p53 can lead to tumor development [10].

A promotion-inhibition network for a portion of the p53 pathway is shown in Figure 8.2a. In roughly 10% of human tumors, p53 is inactivated through overexpression of MDM2 [80]. MDM2

can be overexpressed through an inactivation of p19 [167], and this is shown in Figure 8.2b. MDM2 works to reduce expression of p53 [95, 80, 10, 208] by increasing the degradation rate of p53 and facilitating the nuclear export of p53 [10, 208]. Thus, inhibition of MDM2 has been considered as a possible strategy for cancer treatment [208, 80].

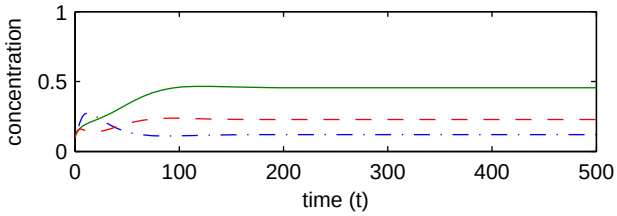
In designing a controller for the abnormal p53 pathway, the underlying biological mechanisms for the edges in the network must be considered. Here, the inhibition edge between MDM2 and p53 is due to protein-protein interaction, and so the controller given in Section 8.3.1 cannot be used. However, the controller given in Section 8.3.2 can be used to remove the vertex corresponding to MDM2. The controller is shown in Figure 8.2c, and is implemented through the addition of antisense RNA that binds with the mRNA for MDM2. Based on existing techniques [85, 197], the controller is hypothetically feasible.

Time course concentrations of p53, cyclin A, and MDM2 are shown in Figure 8.3a for the normal p53 pathway, Figure 8.3b for the abnormal p53 pathway, and Figure 8.3c for the abnormal p53 pathway with controller. These simulations come from an ODE model of the network, and in the simulations the edges between either MDM2 and p53 or MDM2 and cyclin A are not removed. In the normal p53 pathway, concentrations of p53 and cyclin A are high, and concentrations of MDM2 are low. In the abnormal p53 pathway, p53 and cyclin concentrations are low, whereas MDM2 is in high concentration. In the abnormal p53 pathway with controller, the controller is used at times $t = 200$, $t = 250$, and $t = 300$. The controller causes cyclin A and p53 concentrations to increase to higher levels, and reduces MDM2 concentrations. The controller must be used at multiple times, because the cyclin A promoter is modeled to decay. So, the effect of the controller wains as time goes on. If the controller is not applied again, the system returns to an abnormal state.

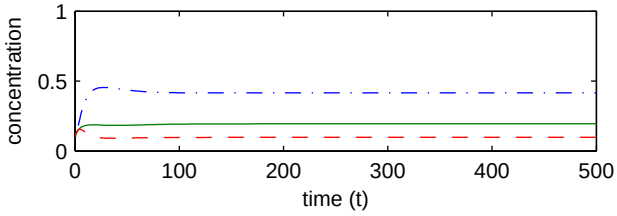
8.5 Future Directions

Steps towards a new paradigm for the control of biological genetic networks through topology based controllers was presented in this chapter. Such techniques may also be useful for understanding the effects of pharmaceuticals. The basic idea of the controller is two-fold. First, the use of pharmaceuticals was abstracted to a graph-theoretical interpretation. Secondly, the dynamics of the system was simplified by removing negative feedback, and then the simplified dynamics steered the system towards a desirable state. Derivations were given for two possible controllers to remove edges or vertices of a network, and one of these was used to treat abnormalities in the cancer-related p53 pathway.

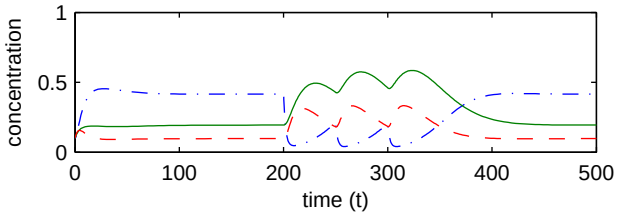
What is needed is an understanding of the biological mechanisms behind interactions, and a library of controllers to deal with and eliminate such interactions. Additionally, algorithms are needed to identify what the optimal edges or vertices to remove are. This is the topic of the next chapters, though there is still more work to be done in this area. Reach set algorithms from hybrid systems may also be useful for this. In fact, PWA hybrid systems were used to analyze global system behavior because efficient algorithms for system analysis exist for such systems [29, 45, 54, 18, 11].



(a) Normal p53 Pathway



(b) Abnormal p53 Pathway



(c) Abnormal p53 Pathway with Controller

Figure 8.3: The time course plots for the different pathways displays the effect of the abnormality and the controller. Note that p53 is solid, cyclin A is dashed, and MDM2 is dash-dotted.

Chapter 9

Monotone Piecewise-Affine Systems

Monotone systems are order-preserving systems: given a partial order on any two initial conditions, the trajectories of the monotone system preserve this partial order through time. There is a rich theory of strong results about the dynamics and stability of monotone systems with continuous vector fields [183, 101, 64]. Though the previous work on monotone systems has largely been theoretical, from an applications point of view there is growing interest in monotone systems due to the realization that many systems are monotone [187, 58, 57].

Piecewise affine (PWA) systems have found application in systems biology [90, 54, 84, 185], traffic engineering [88, 57], and multi-robot systems [58]. Understanding which conditions are sufficient for a PWA system to be monotone is useful, both for understanding the dynamics and for designing controllers for qualitative, reference tracking [19, 57]. Unfortunately, the existing results on monotone systems do not apply to PWA systems, which have discontinuous vector fields.

In the biological context, sufficient conditions for monotonicity can be used to design controllers, as discussed in Chapter 8. Yet, many system identification procedures, such as the one discussed in Chapters 4 and 5 or in [55, 160], identify PWA models. Therefore, the prevalence of such PWA models makes it important to develop tools and analyses which handle such PWA models. Extending monotone systems theory to PWA systems allows for the formal extension of the topological control techniques described in Chapter 8.

This chapter contains sufficient conditions for monotonicity of PWA systems with hyper-rectangular invariants, which are a particular restriction on the form of the vector field [20]. In particular, proofs are given for analogs of the Kamke-Müller theorem [183, 101] and the graph theoretical theorem of [125], both of which provide sufficient conditions for a system with continuous vector field to be monotone. Though the definitions and sufficient conditions described herein bear a strong similarity to those in the existing literature [183, 101], the analysis is modified to deal with the discontinuous vector fields. The main difference with PWA systems is that solutions of the vector field are not unique on certain sets of measure zero; so to have a well-defined notion of monotonicity, the focus needs to be on the behavior of the system almost everywhere. Finally, these results are used to aid with drug discovery by designing controllers for genetic regulatory networks.

9.1 Preliminaries

Recall the definitions of set operations, hyperrectangles, and hyperedges given in Section 8.1. Next, consider the affine ordinary differential equation (ODE)

$$\dot{x} = Ax + b \tag{9.1}$$

with an embedded submanifold of final conditions $\mathcal{X} \subset \mathbb{R}^n$ of dimension $d \leq n-1$. Assume d strictly less than n, because in later sections the interest will be in computing the backward reach sets of hyperedges. Let $\mathcal{M}$ be an open subset of $\mathbb{R}^d$, and define a smooth embedding $h : \mathcal{M} \to \mathcal{X}$. The reach map $R : \mathcal{M} \times \mathbb{R} \to \mathbb{R}^n$ is defined as

$$R(u, t; A, b) = e^{At}h(u) + \int_0^t e^{A(t-\tau)}bd\tau, \tag{9.2}$$

and the backward reach set of $\mathcal{X}$ is the image of the reach map R applied to the whole manifold $\mathcal{M} \times \mathbb{R}_-$: $R(\mathcal{X}; A, b) \triangleq R(\mathcal{M}, \mathbb{R}_-; A, b)$. Note that, depending on the properties of A and b, it is possible for the Lebesgue measure of the backward reach set to be non-zero when $d = n-1$. If $d \leq n-2$, then the Lebesgue measure of the backward reach set can be characterized:

Lemma 9.1.1 (Lee [129, Lemma 10.3])**.** For the affine ODE defined in (9.1) with embedded submanifold of final conditions $\mathcal{X}$, if $d \leq n-2$, then $\mu(R(\mathcal{X}; A, b)) = 0$.

9.2 Defining Monotonicity for Piecewise Affine Systems

Definition 9.2.1. Define a PWA system on a domain $\mathcal{D}$ as a collection of hyperrectangles $\mathcal{C}^i$, with $\dot{x} = A^i x + b^i$ for $x \in \mathcal{C}^i$, such that $\mathcal{D} = \text{int}(\bigcup_{i>0} \overline{\mathcal{C}^i})$ and $\mathcal{C}^i \cap \mathcal{C}^j = \emptyset$ for $i \neq j$. A trajectory of this system is a solution of the vector field $f(x)$ in the sense of Filippov [71], where $f(x) = A^i x + b^i$ if $x \in \mathcal{C}^i$ and is undefined otherwise. Specifically, a trajectory of this system with initial condition $x \in \mathcal{D}$ is given by an absolutely continuous function $\psi_t(x) \in \mathcal{D}$ such that

$$\frac{d\psi_t(x)}{dt} \in \bigcap_{\delta>0} \bigcap_{\mu(N)=0} \overline{\text{co}}(f(\mathcal{B}(x, \delta) \setminus N)) \tag{9.3}$$

almost everywhere, where $\mathcal{B}(x, \delta) = \{y : \|x - y\|_2{}^2 < \delta\}$ and the intersection is taken over all sets N with measure zero. A solution in the sense of Filippov is not necessarily unique; this property is unfortunate, because the non-uniqueness of solutions can lead to a lack of global monotonicity. For initial condition $x \in \mathcal{D}$, define $\mathcal{T}_x = [0, t_f)$ as the maximal interval such that $\psi_t(x) \in \mathcal{D}$. Note that t_f can be $+\infty$, and t_f can be interpreted as the escape time at which $\psi_{t_f}(x) \notin \mathcal{D}$.

It is interesting to derive conditions under which such PWA systems are monotone. Fix $m = (m_1, \ldots, m_n)$, with $m_i \in \{0, 1\}$ for $i \in \{1, \ldots, n\}$, and define

$$K_m = \{x \in \mathbb{R}^n : (-1)^{m_i} x_i \geq 0, \forall i \in \{1, \ldots, n\}\}. \tag{9.4}$$

Note that K_m is a cone that defines a partial ordering: $x \leq_{K_m} y \Leftrightarrow y - x \in K_m$. Also, denote $x \ll_{K_m} y \Leftrightarrow y - x \in \text{int}(K_m)$.

Definition 9.2.2. A PWA system is a "type-K_m almost everywhere" (type-K_ma.e.) monotone system if there exists $\mathcal{Z} \subset \mathcal{D}$ with $\mu(\mathcal{Z}) = 0$, such that for all $x, y \in \mathcal{D} \setminus \mathcal{Z}$, where $x \leq_{K_m} y$, it follows that $\psi_t(x) \leq_{K_m} \psi_t(y)$ for all $t \in \mathcal{T}_x \cap \mathcal{T}_y$.

Remark 9.2.1. This definition is nearly identical to that given in [183], except the definition says that the system is monotone except on a set of points of measure zero. Both definitions are global in nature, and so the existing results for monotone systems [183, 101] can easily be extended to type-K_ma.e. systems.

Definition 9.2.3. A PWA system is a "type-Locally K_m almost everywhere" (type-LK_ma.e.) monotone system if there exists $\mathcal{Z} \subset \mathcal{D}$ with $\mu(\mathcal{Z}) = 0$, such that for all $x, y \in \mathcal{D} \setminus \mathcal{Z}$, where $x \leq_{K_m} y$, there exists $\delta > 0$ such that $\psi_t(x) \leq_{K_m} \psi_t(y)$ for all $t \in [0, \delta)$.

Remark 9.2.2. This definition is local in nature: it asks that the system be monotone for short intervals. This distinction between local and global monotonicity does not occur with continuous vector fields and is unique to PWA systems. Consequently, not all existing results for monotone systems [183, 101] can be extended to type-LK_ma.e. systems.

These are natural definitions to make because the non-uniqueness of Filippov solutions leads to a destruction of monotone properties; these definitions are made in order to have a meaningful notion of a monotone PWA system. This destruction of monotone properties because of non-uniqueness of solutions can be demonstrated with a simple example: $\mathcal{C}_1 = (0,1) \times (0,1)$, $b_1 = [-1 \;\; 0]^T$; $\mathcal{C}_2 = (0,1) \times (1,2)$, $b_2 = [1 \;\; 0]^T$; where $A_1 = A_2 = \text{diag}(0,0)$. Choosing $m = (0,0)$, it follows that for all fixed $i \in \{1,2\}$: if $x, y \in \mathcal{C}_1 \cup \mathcal{C}_2$, $x_i = y_i$, and $x \leq_{K_m} y$, then $f(x)_i \leq f(y)_i$. By analogy to monotone system theory for continuous vector fields [183, 101], it is expected this system to be type-K_m monotone. Yet choosing $x = (0.25, 1)$ and $y = (0.75, 1)$, then a set of feasible trajectories is $\psi_t(x) = (0.25 + t, 1)$ and $\psi_t(y) = (0.75 - t, 1)$. At $t = 0.3$, the monotone property is violated because $\psi_{0.3}(x) \geq_{K_m} \psi_{0.3}(y)$. This occurs because initial conditions $x, y \in \overline{\mathcal{C}_1} \cap \overline{\mathcal{C}_2}$ do not have a unique trajectory.

9.3 Reach Sets of Measure Zero

In general, PWA systems do not have unique solutions along hyperedges of dimension $d \leq n-2$ [90]; trajectories along these hyperedges are unique only under special cases [71]. Under suitable conditions, a PWA system will be monotone on a set of points which excludes the set of initial conditions with non-unique trajectories. If the trajectories are locally unique, then the system is type-LK_ma.e. monotone; whereas, if the trajectories are globally unique, then the system is type-K_ma.e. monotone.

This section contains sufficient conditions that distinguish between a) locally unique solutions and b) globally unique solutions. Also given are sufficient conditions for the set of initial conditions with globally non-unique solutions to have zero measure. The following lemma characterizes the intersections of backward reach sets with hyperedges of dimension $n-1$.

Lemma 9.3.1. Consider a set of final conditions $\mathcal{X}$ with dimension $d \leq n-2$, and recall that $\mathcal{X} = h(\mathcal{M})$. Suppose that for some hyperedge $\mathcal{E}$ it holds that $\mathcal{X} \cap \overline{\mathcal{E}} \neq \emptyset$. If $\dim(\mathcal{E}) = n-1$ and $e_k^T(A \cdot h(u) + b) \neq 0$ for all normals e_k to $\mathcal{E}$, then either $R(\mathcal{X}; A, b) \cap \mathcal{E} = \emptyset$ or $R(\mathcal{X}; A, b) \cap \mathcal{E}$ is an embedded submanifold of dimension d.

For brevity, define two hypotheses. Note that these hypotheses are not the full set of sufficient conditions for a PWA system to be monotone; there are more general conditions that are given in Corollary 9.4.1.

Hypothesis 9.3.1. For all fixed $k \in \{1, \ldots, n\}$, let $m = (0, \ldots, 0)$ and assume that $\forall x, y \in \bigcup_{q>0} \mathcal{C}^q$, where $x \leq_{K_m} y$ and $x_k = y_k$, it holds that $f(x)_k \leq f(y)_k$.

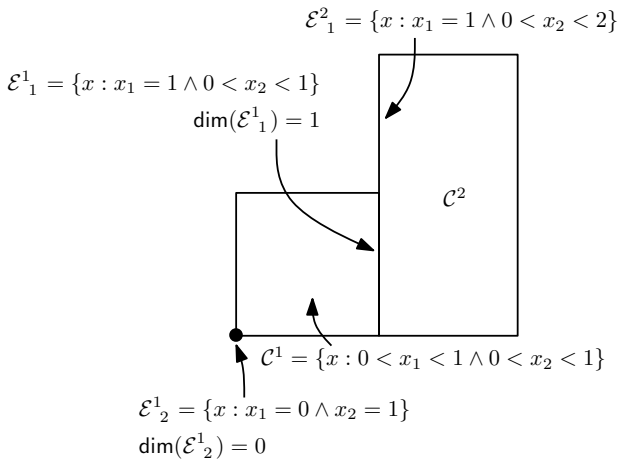


Figure 9.1: A simple example of hyperrectangles and the associated hyperedges in $\mathbb{R}^2$ is shown. Only two of the associated hyperedges of $\mathcal{C}^1$ are labeled; there are eight hyperedges associated to $\mathcal{C}^1$. Also, in this example we have that $\mathcal{E}^1{}_1 <_1 \mathcal{E}^2{}_1$.

Remark 9.3.1. For this particular choice of m, the cone K_m is the positive-orthant $\mathbb{R}^n_+$, and so $x \leq_{K_m} y$ means that $x_i \leq y_i$, for all $i \in \{1, \ldots, n\}$. The hypothesis applies to vector fields that are monotone with respect to the positive-orthant.

Denote $\mathcal{E}^q{}_i <_j \mathcal{E}^r{}_k$, if for two $(n-1)$-dimensional hyperedges with normal e_j, the following holds

$$\sup_{x \in \mathcal{C}^q} (e_j^T x) < \sup_{y \in \mathcal{C}^r} (e_j^T y). \tag{9.5}$$

A example of this notation is shown in Figure 9.1.

Hypothesis 9.3.2. Assume that $\forall x \in \mathcal{E}^q{}_i \cap \mathcal{E}^r{}_k$, where $\mathcal{E}^q{}_i <_j \mathcal{E}^r{}_k$, $\dim(\mathcal{E}^q{}_i) = \dim(\mathcal{E}^r{}_k) = n-1$, and for the common normal e_j to the hyperedges, it does not hold that: a) $e_j^T(A^q x + b^q) \geq 0$ and $e_j^T(A^r x + b^r) \leq 0$; b) $e_j^T(A^q x + b^q) = 0$; or c) $e_j^T(A^r x + b^r) = 0$.

Remark 9.3.2. The hypothesis applies to vector fields that do not have sliding modes, as this is prohibited by assumption (a). Additionally, degenerate vector fields that are parallel to the hyperedges of dimension $n-1$ are prohibited by assumptions (b) and (c).

Define a series of sets on which the dynamics have the same qualitative behavior. The first set is the union of the hyperrectangles, $\mathcal{W}_I = \bigcup_{i>0} \mathcal{C}^i$. The next sets involve $(n-1)$-dimensional

hyperedges,

$$\mathcal{W}_{II}{}^{qr} = \{x \in \mathcal{E}^q{}_i \cap \mathcal{E}^r{}_k : \mathcal{E}^q{}_i <_j \mathcal{E}^r{}_k \wedge \\ e_j^T(A^q x + b^q) > 0 \wedge e_j^T(A^r x + b^r) > 0\} \tag{9.6}$$

$$\mathcal{W}_{III}{}^{qr} = \{x \in \mathcal{E}^q{}_i \cap \mathcal{E}^r{}_k : \mathcal{E}^q{}_i <_j \mathcal{E}^r{}_k \wedge \\ e_j^T(A^q x + b^q) < 0 \wedge e_j^T(A^r x + b^r) < 0\} \tag{9.7}$$

$$\mathcal{W}_{IV}{}^{qr} = \{x \in \mathcal{E}^q{}_i \cap \mathcal{E}^r{}_k : \mathcal{E}^q{}_i <_j \mathcal{E}^r{}_k \wedge \\ e_j^T(A^q x + b^q) \geq 0 \wedge e_j^T(A^r x + b^r) \leq 0\} \setminus \mathcal{W}_V \tag{9.8}$$

$$\mathcal{W}_V = \bigcup_{q>0} \bigcup_{r>0} \{x \in \mathcal{E}^q{}_i \cap \mathcal{E}^r{}_k : \mathcal{E}^q{}_i <_j \mathcal{E}^r{}_k \wedge \\ e_j^T(A^q x + b^q) \leq 0 \wedge e_j^T(A^r x + b^r) \geq 0\}. \tag{9.9}$$

Note that the sets (9.6)—(9.9) do not depend on (i,k), because there is only one pair (i,k) that produces a nonempty set for a given pair (q,r). The last set is a union of hyperedges with dimension $d \leq n-2$, $\mathcal{W}_{VI} = \bigcup_{q>0} \bigcup_{i>0} \mathcal{E}^q{}_i$. It will be shown that the set of points on which uniqueness of solutions is not guaranteed is given by

$$\mathcal{Z}_0 = \mathcal{W}_V \cup \mathcal{W}_{VI}. \tag{9.10}$$

As alluded to, the dynamics have the same qualitative behavior in each of these sets. The dynamics are PWA on the set of the hyperrectangular invariants $\mathcal{W}_I$. The sets $\mathcal{W}_{II}{}^{qr}$ and $\mathcal{W}_{III}{}^{qr}$ are sometimes called *transparent walls* [90], because trajectories pass through the hyperedges contained in these edges. The sets $\mathcal{W}_{IV}{}^{qr}$ can be referred to as *black walls* [90], because trajectories that hit the hyperedges in this set become stuck and move along the hyperedges. The set $\mathcal{W}_V$ is sometimes known as the union of *white walls* [90], because trajectories emanate from the hyperedges in this set. Figures with examples of these sets can be found in [90].

Note that the vector field $f(x)$ is only defined for $x \in \mathcal{W}_I$. Extend this vector field by defining:

$$\hat{f}(x) \triangleq \begin{cases} f(x), & \text{if } x \in \mathcal{W}_I \\ A^r x + b^r, & \text{if } x \in \mathcal{W}_{II}{}^{qr} \\ A^q x + b^q, & \text{if } x \in \mathcal{W}_{III}{}^{qr} \\ \alpha(A^q x + b^q) + & \\ \quad (1-\alpha)(A^r x + b^r), & \text{if } x \in \mathcal{W}_{IV}{}^{qr} \end{cases}, \tag{9.11}$$

where $\alpha = e_j^T(A^q x + b^q)/(e_j^T(A^q x + b^q) + e_j^T(A^r x + b^r))$, and $\hat{f}(x)$ is defined for $x \in \mathcal{D} \setminus \mathcal{Z}_0$.

Lemma 9.3.2. If Hypothesis 9.3.1 holds, then $\forall x, y \in \mathcal{D} \setminus \mathcal{Z}_0$, where $x \leq_{K_m} y$ and $x_k = y_k$, it holds that $\hat{f}(x)_k \leq \hat{f}(y)_k$.

It is said that *right uniqueness* holds at point x_0 if for $x = x_0$ and $y = x_0$, if for any two solutions of the PWA system, there exists a δ such that $\psi_t(x) \equiv \psi_t(y)$, for $t \in [0, \delta]$. Similarly, *left uniqueness* holds at point x_0 if for $x = x_0$ and $y = x_0$, it holds that for any two solutions of the PWA system with modified vector field $\tilde{f}(x) = -f(x)$, there exists a δ such that $\tilde{\psi}_t(x) \equiv \tilde{\psi}_t(y)$, for $t \in [0, \delta]$. Note that left uniqueness corresponds to uniqueness of trajectories running backwards in time, because time reversal of a time-invariant system is identical to negating the vector field of the system and then solving the equations forward in time.

With these definitions, the following lemma characterizes the set of points which exhibit left and right uniqueness.

Lemma 9.3.3. If Hypothesis 9.3.1 holds, then right uniqueness holds for $x \in \mathcal{D} \setminus \mathcal{Z}_0$, and also $\hat{f}(\psi_s(x)) = \frac{d}{dt}\psi_t(x)|_{t=s^+}$. If Hypothesis 9.3.2 also holds, then left uniqueness also holds for $x \in \mathcal{D} \setminus \mathcal{Z}_0$.

Define the set $\mathcal{Z} = \{x : \exists t \in \mathcal{T}_x, \text{ s.t. } \psi_t(x) \in \mathcal{Z}_0\}$. Uniqueness of solutions is not guaranteed by Lemma 9.3.3 for points in $\mathcal{Z}_0$. So, $\mathcal{Z}_0$ can be thought of as an unsafe set and $\mathcal{Z}$ as the backward reach set of this unsafe set [149]. The following lemma characterizes the Lebesgue measure of the set $\mathcal{Z}$.

Lemma 9.3.4. If Hypothesis 9.3.1 and Hypothesis 9.3.2 both hold, then $\mu(\mathcal{Z}) = 0$.

9.4 Sufficient Conditions for Monotone PWA Systems

Theorem 9.4.1 and Corollary 9.4.1 are analogs of the classical Kamke-Müller theorem [183, 101]. Theorem 9.4.2 is an analog of the graph theoretical sufficient conditions of [125]. Our results generalize this previous work to PWA systems.

9.4.1 Kamke-Müller Analogs

This section provides sufficient conditions for a system to be monotone for $m = (0, \ldots, 0)$. With this established, a change of variables can be used to relate sufficient conditions for arbitrary m to sufficient conditions for $m = (0, \ldots, 0)$.

Theorem 9.4.1. If Hypothesis 9.3.1 holds, then the PWA system is type-LK_ma.e. monotone. If Hypothesis 9.3.2 also holds, then the PWA system is also type-K_ma.e. monotone.

For $m = (m_1, \ldots, m_n)$, with $m_i \in \{0, 1\}$ for $i \in \{1, \ldots, n\}$, associate the change of variables matrix $P_m = \text{diag}(-1^{m_1}, \ldots, -1^{m_n})$.

Corollary 9.4.1 (Smith [183, Lemma 2.1])**.** If there exists $m = (m_1, \ldots, m_n)$, with $m_i \in \{0, 1\}$ for $i \in \{1, \ldots, n\}$, such that

(i) $\tilde{f}(x)$ satisfies Hypothesis 9.3.1 for $x \in P_m\mathcal{D}$, then the PWA system is type-LK_ma.e. monotone;

(ii) $\tilde{f}(x)$ also satisfies Hypothesis 9.3.2 for $x \in P_m\mathcal{D}$, then the PWA system is also type-K_ma.e. monotone;

where $\tilde{f}(x) = P_m f(P_m x)$.

9.4.2 Graphical Conditions

Theorem 2 provides an easy, graphical method to check if a PWA system satisfies Corollary 9.4.1. The conditions of this theorem are stricter than the conditions of the corollary, precisely because this theorem is a set of sufficient conditions which implies the hypothesis of the corollary.

Defining

$$x(\xi; j, \nu) = (\xi_j + \nu)e_j + \sum_{i \neq j} \xi_i e_i, \tag{9.12}$$

the expression $f(x(\xi; j, \nu))_i$ can be intuitively thought of as an analog of the partial derivative of the i-th component of f with respect to x_j. Furthermore, if $\forall i, j \in \{1, \ldots, n\}$ and $\forall \xi \in \mathcal{W}_I$ it holds that $f(x(\xi; j, \nu))_i$ does not both increase and decrease for increasing ν, then a signed, directed graph (V, E, S) can be associated to the PWA system. An example of a PWA system and its associated graph is given in Section 9.5.1.

Here, the vertices $V = \{v_1, \ldots, v_n\}$ correspond to the state variables $x_1, \ldots, x_n$; a directed edge from v_i to v_j is given by $(v_i, v_j) \in E$; and the edges are signed $S((v_i, v_j)) \in \{-1, +1\}$. Construct the associated signed, directed graph as follows:

- $\forall \xi \in \mathcal{W}_I$, $f(x(\xi; j, \nu))_i$ is constant for all ν, then there is no edge from v_j to v_i;
- $\forall \xi \in \mathcal{W}_I$, $f(x(\xi; j, \nu))_i$ is non-decreasing for increasing ν, then $(v_j, v_i) \in E$ and $S((v_j, v_i)) = +1$;
- $\forall \xi \in \mathcal{W}_I$, $f(x(\xi; j, \nu))_i$ is non-increasing for increasing ν, then $(v_j, v_i) \in E$ and $S((v_j, v_i)) = -1$;

Moreover, define an undirected cycle:

$$\mathcal{L} = \{u_1, \ldots, u_m, u_{m+1} = u_1; e_1, \ldots, e_m\}, \tag{9.13}$$

where for $i = 1, \ldots, m$ it holds that $u_i \in V$ and either $e_i = (u_i, u_{i+1}) \in E$ or $e_i = (u_{i+1}, u_i) \in E$. A negative, undirected cycle is defined as a cycle $\mathcal{L}$ such that $\prod_{i=1}^{m} S(e_i) = -1$.

Theorem 9.4.2. Assume that $\forall i, j \in \{1, \ldots, n\}$ and $\forall \xi \in \mathcal{W}_I$, $f(x(\xi; j, \nu))$ does not both increase and decrease for increasing ν. If the associated signed, directed graph (V, E, S) has no negative, undirected cycles, except for negative, self-cycles which are allowed, then the PWA system is type-LK_ma.e. monotone. If additionally the graph (V, E, S) has no negative, self-cycles and conditions (b) and (c) of Hypothesis 9.3.2 hold, then the PWA system is also type-K_ma.e. monotone.

Remark 9.4.1. The results in [69, 84] concern the existence of a stable periodic orbit in PWA systems whose associated graph consists of a negative, simple cycle. Monotone systems do not have stable periodic orbits [101], but [69, 84] do not contradict these results: these sufficient conditions for type-K_ma.e. monotonicity prohibit the case of a negative, simple cycle.

9.5 Example: Genetic Regulatory Networks

Monotone PWA systems arise in a number of systems of interest to controls engineers; such areas include genetic regulatory networks (GRNs) and traffic engineering. This section focuses on biological applications.

9.5.1 Associated Graph of a PWA System

As a simple example of a GRN, consider the following PWA system: $\mathcal{C}_1 = (0, 1) \times (0, 1)$, $b_1 = [2 \;\; 0]^T$; $\mathcal{C}_2 = (0, 1) \times (1, 2)$, $b_2 = [0 \;\; 0]^T$; $\mathcal{C}_3 = (1, 2) \times (0, 1)$, $b_3 = [2 \;\; 2]^T$; $\mathcal{C}_4 = (1, 2) \times (1, 2)$, $b_4 = [0 \;\; 2]^T$; where $A_1 = A_2 = A_3 = A_4 = \text{diag}(-1, -1)$, and the states x_1, x_2 correspond to the concentrations of protein A and protein B. Using the procedure of Section 9.4.2, a graph can be associated to this PWA system. The associated graph is shown in Figure 9.2a. This GRN can be thought of as a negative feedback loop between protein A and protein B, because the associated graph is a negative, simple cycle. Consequently, the sufficient condition of Theorem 9.4.2 is not met. In fact, the results of [69, 84] can be used to show that this system is not monotonic.

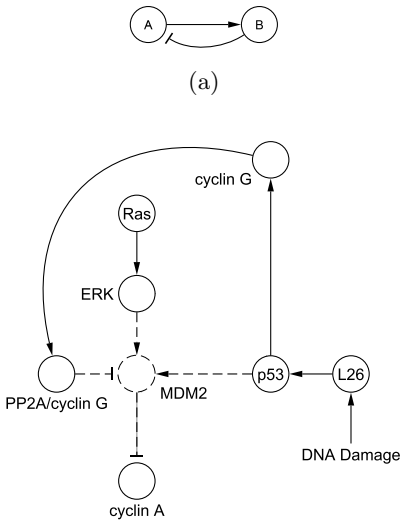


Figure 9.2: (a) The graph associated to the system given in Sect. (9.5.1) is a negative feedback loop. Note that a $+1$ (-1) signed edge has a "$\rightarrow$" ("$\dashv$"). (b) The effect of applying a pharmaceutical, which removes the dashed elements of the network, to the cancer-related p53 pathway [95, 167, 196, 130] can be viewed as converting the pathway into a network whose associated PWA system is type-K_ma.e. monotone.

9.5.2 Qualitative Reference Tracking

A promotion-inhibition network is a signed, directed graph $N = (V, E, S)$, and it is a common, qualitative model for GRNs. An example is shown in Figure 9.2b. Intuitively, a positively (negatively) signed edge between two vertices means that an increase in the direct predecessor leads to an increase (decrease) in the direct successor. There are methods that convert a promotion-inhibition network into a PWA system by suitably choosing a vector field [90, 54, 84, 185].

If N has no negative, undirected cycles—except for negative, self-cycles—then by Theorem 9.4.2 the associated PWA system is type-LK_ma.e. monotone. If N additionally has no negative, self-cycles, then the vector field can be chosen to enforce Hypothesis 9.3.2 [90, 54, 84, 185]; consequently, the associated PWA system is type-K_ma.e. monotone by Theorem 9.4.2. Further results, such as convergence of trajectories to equilibrium points, can be shown under additional assumptions, such as the positive-invariance of $\mathcal{D}$ [183].

These results can be used for the control of GRNs [19] whose dynamics are given by PWA systems. These controllers are open-loop and apply to both classical and PWA monotone systems. In [19], it was shown that the chemical effect of pharmaceuticals and gene therapy on a GRN can be abstracted as the removal of edges or vertices from N. Consequently, qualitative, reference tracking control objectives can be posed as a related problem: remove edges and vertices of N to make the system into a type-K_ma.e. monotone system [19]. This can be done because monotone systems have well-defined input-output behaviors [187]. The control action is then described in terms of potential targets for pharmaceuticals or gene therapy to act on.

The protein p53 has important cellular functions, and low concentrations of p53 are correlated with cancerous cells. An example of a mutated p53 pathway which leads to low p53

concentrations, in the presence of DNA damage, is shown in Figure 9.2b; the mutated pathway consists of both the black and dashed elements. A qualitative, reference tracking objective would be to have high p53 concentration in the presence of DNA damage. Removing the dashed elements of Figure 9.2b from the network via a pharmaceutical causes the system to become type-K_ma.e. monotone. Furthermore, this system satisfies the qualitative, control objectives: p53 concentration is high in the presence of DNA damage.

9.6 Traffic Engineering

The Lighthill-Whitham-Richards (LWR) model is a partial-differential equation model of traffic flow [88, 195]. The Godunuv scheme can be used to convert an LWR model of a single freeway into the cell transmission model (CTM), a discrete time model for N sections of a freeway. In the CTM, each section has one on- and one off-ramp. Let x be a N-dimension vector of vehicle counts on each section of the freeway, then the CTM model is given by $x[n+1] = g(x[n])$, where g is a PWA function and satisfies our Hypothesis 9.3.1 [88], since this results also hold for $\mathcal{E}$ which are arbitrary, embedded submanifolds which divide $\mathcal{D}$. One can also use a semi-discrete Godunuv scheme to spatially, but not numerically, discretize the LWR model. This gives the model $\dot{x} = g(x)$, and by Theorem 9.4.1 this system is type-LK_ma.e. monotone. Showing that the system is type-K_ma.e. monotone requires extending the results presented in this chapter and then performing additional computations. We suspect that it is indeed type-K_ma.e. monotone, but this needs to be checked.

Chapter 10

Graph-Theoretic Topological Control

A common theme repeated throughout the second part of this thesis is that biological genetic networks contain features that make it difficult to do traditional control. Measuring the states of a system for the purposes of feedback control can be prohibitively difficult or even not feasible with current technologies. Moreover, such genetic networks do not typically have inputs that can be changed to do control. In light of these difficulties, a new framework was proposed in [19] for doing the topological control for such networks. A related line of research that has been developed concurrently is [150].

In the framework considered here, the affect of drugs, pharmaceuticals, and gene therapy are abstracted to having a graph theoretic interpretation. It is common for biologists to abstract genetic networks, which are dynamical systems, to a signed, directed graph which qualitatively describes the influence of a state on another state. This graph is often referred to as a promotion-inhibition network. Using a quasi-steady-state approximation, it can be shown that drugs and pharmaceuticals can be interpreted as modifying the signed, directed graph by removing vertices or edges of the graph [19].

If pharmaceuticals are interpreted as modifying a graph, then control can be accomplished by intelligently modifying the topology of the genetic networks. The topology of a network will remain modified only in the presence of the pharmaceutical: As soon as the pharmaceutical degrades, the topology of the network will return to an unmodified nature, and the system will go back to being uncontrolled. The control that is chosen also depends on having correct knowledge of the genetic network.

Topological control can be done by using theorems that relate the topology of the network to the dynamical behavior of the network. A big class of results concerns monotone systems [184, 187, 125], systems with no undirected, negative cycles within the graph of the network. These systems do not have any stable oscillations, and all trajectories converge to equilibrium points. A related class of results concerns systems with no directed, negative cycles within the graph of the network. If the graph is also strongly connected, then all trajectories of the system converge to equilibrium points and there are no stable oscillations [121]. Similar results are found in [159, 89, 47, 188]. These results can be extended to prove that all systems with no directed, negative cycles have the same behavior. The particular case of piecewise-affine hybrid systems with no self-inhibition was proved in [18, 11], and the more general case of arbitrary smooth vector fields was proved in [187].

Directed, negative cycles correspond to the intuitive notion of negative feedback in a system. Undirected, negative cycles, which are a superset of directed, negative cycles, do not

always match the intuitive notion of negative feedback. Certain control objectives can be satisfied by removing the negative feedback of the system, which removes the oscillations of the system. This is the goal of the current chapter: How can negative feedback be removed from the system, so that the system trajectories do not have stable oscillations and converge to equilibrium points? The goal is: What should pharmaceuticals be designed to target, so that the concentrations in the genetic network converge to equilibria?

This is a crude level of control, but returning to the biological example in Chapter 8 of the p53 pathway—which is implicated in cancer—shows that it can generate useful controllers. This type of control is related to the work in [108, 50]; however, the difference is that removing undirected, negative cycles is a more restrictive condition, because negative feedback is a subset of undirected, negative cycles.

An influence graph (also known as a promotion-inhibition network) is a signed, directed graph $G = (V, E, S)$, where $V = \{v_1, \ldots, v_n\}$ is the set of vertices, $E \subseteq \{(u, v) : u, v \in V\}$ is the set of directed edges, and $S : E \to \{-1, +1\}$ is a function that gives the sign of an edge. For an edge $e = (u, v)$: u is the direct predecessor of v, and v is the direct successor of u. Edges labeled -1 are called *inhibition* edges, while edges labeled maps to 1 are called *promotion* edges. A simple directed cycle $l = (e_1, e_2, \cdots e_n)$ with all $e_i \in E$ is called a *negative feedback loop* if and only if it contains an odd number of inhibitory edges; in other words:

$$\prod_{e \in l} \gamma(e) = -1.$$

Consider the problem of modifying an influence graph to eliminate such cycles: Given an influence graph G and a weighting function $\omega : E \cup V \mapsto \mathbb{R}$, find the minimum weight subsets (possibly empty) $E' \subset E$ and $V' \subset V$ such that removal of both of these subsets from the influence graph causes the graph to have no negative feedback. As mentioned before, this problem, while similar to the problem of balancing signed directed graphs discussed in [108, 50], is different in that directed, as opposed to undirected, feedback cycles are considered here. This problem is also similar to that in [186], but there are differences in: the weights used in the problem, the application of the problem, and the algorithm used to solve the problem.

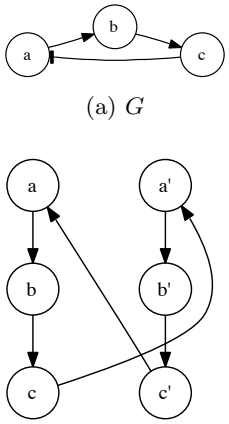
The weights ω can be interpreted as the cost of removing an edge or vertex, and there are many biological interpretations of the weight. For instance, if an existing drug can remove an edge, then the weight of that edge can be set low, because there is a lower cost to removing that edge.

This chapter begins by proving that the decision version of the negative feedback edge-vertex deletion problem is in NP by demonstrating a polynomial time algorithm to check whether a given influence graph has negative feedback or not. Then, using results from [132], it is shown that the node deletion problem—and thus the node/vertex deletion problem—for negative feedback is NP-hard. An integer linear program (ILP) for eliminating negative feedback in an arbitrary graph is proposed [15]. Trivial modifications of efficient approximation algorithms for the directed multicut problem [44, 93] are used to solve this ILP. Lastly, this approach is used on a few biological examples and on the p53 pathway, a pathway involved in human cancers.

10.1 Negative Feedback Edge/Vertex Removal is NP-Hard

10.1.1 Decision Version of Negative Feedback Edge/Vertex Removal is in NP

To show that the decision version of the negative feedback problem lies in NP, a polynomial time algorithm is demonstrated which determines if a given influence graph has negative feedback



(a) G

(b) $\gamma(G)$

Figure 10.1: Influence graph G and $\gamma(G)$

cycles. Though this fact was stated without proof in [186], it is useful to formally prove this, because the proof will provide the intuition behind the algorithm presented in this chapter. To do this, first define the operation $\delta : G \to G'$ that maps an influence graph $G = (V, E, S)$ to a directed graph $G' = (V', E')$. Define the operation $\delta(G) = (\eta(V), \tau(E, S)) = (V', E')$ by defining the two functions $\eta(\cdot)$ and $\tau(\cdot)$. Specifically, define a bijection $\eta : V \to P$ that makes a clone of every vertex. Here, $P = \{\eta(v) : v \in V\}$ is a clone of every vertex V. Also, define the vertices of the graph G as $V' = V \cup P$.

Next, define a function that doubles every edge. If the edge is an promotion edge, make two edges: one of which stays within V and one of which stays within P. If the edge is an inhibition edge, make an edge that crosses from V to P and an edge that crosses from P to V (see Figure 10.1). This is done using a one-to-two correspondence τ which maps each edge in E to two edges in $\delta(G)$. In particular,

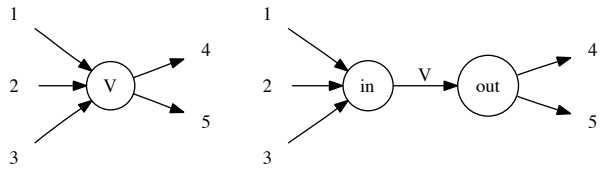
$$\tau((u,v)) = \begin{cases} \{(u,v),(\eta(u),\eta(v))\} & \text{if } \gamma((u,v)) = 1 \\ \{(u,\eta(v)),(\eta(u),v)\} & \text{if } \gamma((u,v)) = -1 \end{cases}$$
$$E' = \bigcup_{e \in E} \tau(e).$$

Note that τ^{-1} is a function. Note that this construction, illustrated in Figure 10.1, superficially resembles the embedding construction in [187]. A closer examination reveals that these two constructions are quite different in terms of operations, and they serve different purposes.

For a path in $\delta(G)$, it has a preimage in G. The preimage of a path is obtained by mapping τ^{-1} onto each edge in the path. Additionally, for every path p in G, a corresponding path in $\delta(G)$ can be constructed by choosing an appropriate edge from $\tau(e)$—for each edge e in p—for the new path.

The existence of a path in G' from any vertex x to its duplicate $\eta(x)$ implies the existence of a negative feedback loop in the original influence graph G, and that any negative feedback loop in G that contains x implies the existence of a path from x to $\eta(x)$ in G'. This can be formalized with the theorem given below. A proof can be found in the appendix.

Theorem 10.1.1. The existence of a negative feedback cycle in G at vertex v implies the existence



(a) Before (b) After

Figure 10.2: Vertex before and after splitting

of a path in $\delta(G)$ from v to $\eta(v)$, and the existence of a path in $\delta(G)$ from v to $\eta(v)$ implies the existence of a negative feedback cycle in G.

Using these ideas, the question of whether or not G has negative feedback cycles can be rephrased as a question about connectivity in $\delta(G)$. This allows checking for the existence of negative feedback cycles by checking if there exists a path between any $v \in V$ and $\eta(v) \in P$ in $\delta(G)$. One easy way to do this is by performing $n = |V|$ depth first searches in $\delta(G)$—a polynomial time operation.

10.1.2 Negative Feedback Edge/Vertex Removal is NP-Hard

Using the results of [132], it can be shown that the node-deletion problem for negative feedback is NP-hard. In [132], the authors prove that for any graph property Π, which is "nontrivial" and "hereditary", the node-deletion problem is NP-hard. *Nontrivial* properties are true for infinitely many graphs and false for infinitely many graphs, while a *hereditary* property is true on all vertex-induced subgraphs of a satisfying graph. For purposes of the presentation here, take $\Pi(G)$ to mean that G has no negative feedback cycles.

Theorem 10.1.2. $\Pi =$ "no negative feedback", is a nontrivial property.

Proof. It is easy to see that Π is nontrivial. Consider the directed cycle graphs C_n (C_n is an n-vertex graph that consists of a single, directed cycle) with each edge inhibitory; when n is odd, C_n is clearly a negative feedback loop, and when n is even, C_n must have no negative feedback. Both of these sets are infinite, so Π must be nontrivial. □

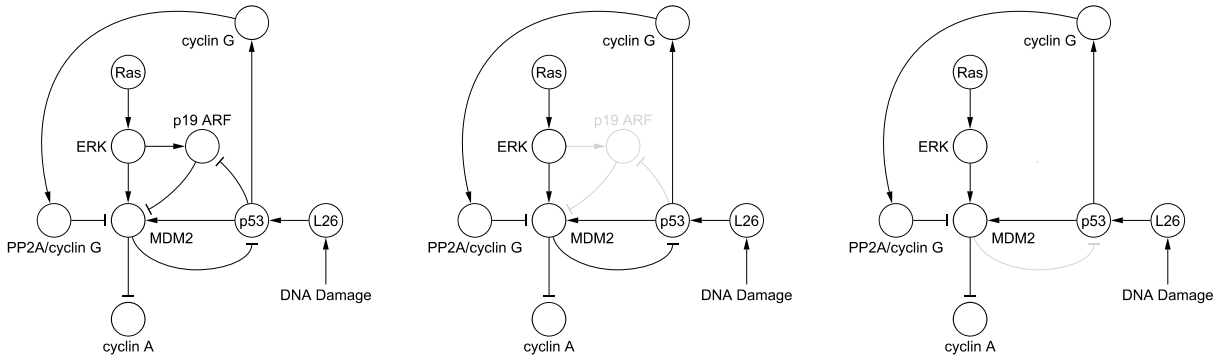
Theorem 10.1.3. $\Pi =$ "no negative feedback", is a hereditary property.

Proof. Proof by contradiction. Assume $\Pi(G)$. If Π is not hereditary, then for some G, there is some vertex-induced subgraph S of G that contains a negative feedback loop. If S is a subgraph of G, any feedback loop in S is also in G, thus G has a negative feedback loop. Contradiction. □

Appealing to Theorem 7 of [132], the conclusion is that the node deletion problem for negative feedback is NP-hard. By appropriate choice of the weighting function ω, the edge-node deletion problem for negative feedback can also be shown to be NP-hard.

10.2 Heuristic Algorithm

The operation $\delta(G)$ provides an interesting way to pose the problem of negative feedback removal: By Theorem 10.1.1, the problem is equivalent to disconnecting each pair v, $\eta(v)$ in $\delta(G)$.



(a) Normal p53 Pathway (b) Abnormal p53 Pathway (c) Abnormal p53 Pathway with Controller

Figure 10.3: When a subsegment of the normal p53 pathway [95, 167, 196, 130] becomes abnormal, such as loss of p19 ARF function [167], the system behaves unfavorably by underexpressing p53. Using a controller, the grayed edge can be removed to make the system behavior more favorably.

This would ensure no negative feedback in the original influence graph. This rephrased problem is very nearly an instance of the *directed multicut* problem discussed in [46].

The directed multicut problem takes a directed graph and a list of source/sink pairs ($[(s_1, t_1)...(s_k, t_k)]$) to be separated. An optimal solution to multicut finds the minimum weight subset of edges that must be cut in order to separate each source from its corresponding sink. The directed multicut problem is NP-hard, and a short description of the multicut integer linear program (ILP) solution is given here.

Each edge is assigned a variable x_e that is either zero or one. If x_e is high in the solution, then cut e. In formulating the constraints, these variables are interpreted as lengths on each edge. Intuitively, the constraints specify that the minimum distance between a source and its corresponding sink be at least one. This ensures that they cannot be connected once a cut is made.

To achieve this, each vertex is assigned k variables $d_{v,i}$, with $i \in [1, k]$. The optimization variable $d_{v,i}$ is the distance between the i-th source and the vertex v. Each distance is required to be consistent with the edge lengths. Specifically, if vertex u is connected by an edge with zero length to vertex v, each $d_{v,i}$ is at most $d_{u,i}$. If they are connected by an edge with length one, each $d_{v,i}$ is at most $d_{u,i} + 1$. Formally, the solution is given by:

Directed Multicut ILP

$$\begin{aligned}
&\min \sum_{e \in E} \omega(e) \cdot x_e \\
&\text{s.t. for all } i \in [1, k] \\
&\quad d_{t_i,i} - d_{s_i,i} \geq 1 \\
&\quad d_{v,i} \leq d_{u,i} + x_e && \text{for all } e = (u, v) \in E \\
&\quad x_e \in \{0, 1\} && \text{for all } e = (u, v) \in E
\end{aligned}$$

Unfortunately, there are two problems with simply running multicut on $\delta(G)$ with each v, $\eta(v)$ pair as a source/sink pair. The first problem is that the multicut problem deals exclusively with

cutting edges. Therefore, the edge/node deletion problem must be reduced into the edge deletion problem in order to take advantage of known approximations of multi-terminal cuts [44, 93]. The reduction is fairly intuitive and can be applied to the unmodified influence graph G (with some modification to the sign function) or to the altered digraph $\delta(G)$. To do the reduction, split each vertex in two to create an in-terminal and an out-terminal for the node. All incoming edges to the vertex are connected to the in-terminal, and all outgoing edges are connected to the out-terminal. Then connect the two terminals with an edge that is equivalent to the original vertex (see Figure 10.2). More formally, transform $\delta(G) = (V, E)$ into $G' = (V', E')$ where

$$V' = \{v_{in}, v_{out} : v \in V\}$$
$$E' = \{(u_{out}, v_{in}) : (u, v) \in E\} \cup \{(v_{in}, v_{out}) : v \in V\}.$$

Then, reassign the weights for the vertices to the new edges that connect each in/out vertex pair. The new edges can be considered equivalent to the original vertices. Coincidentally, this manipulation can be used to show that the edge-deletion problem is also NP-hard. From here on, only the edge-deletion variant of the problem will be considered because, as shown, the node/edge deletion problem is equivalent.

The second problem with directed multicut stems from the fact that τ relates each edge in G to two edges in $\delta(G)$. So, the two edges could be separately cut. To get around this problem, the directed multicut ILP is modified slightly by having each edge variable correspond to two edges instead of one,. This leads to the final formulation of the edge deletion negative feedback problem:

Negative Feedback ILP

$$\min \sum_{e \in E} \omega(e) \cdot x_e$$
$$\text{s.t. for all } i \in [1, |V|]$$
$$d_{v_i,i} - d_{\eta(v_i),i} \geq 1$$

$$d_{v_i,i} \leq d_{u_i,i} + x_e \qquad \text{for all excitatory edges,}$$
$$d_{\eta(v_i),i} \leq d_{\eta(u_i),i} \qquad e = (u, v) \in E$$

$$d_{v_i,i} \leq d_{\eta(u_i),i} + x_e \qquad \text{for all inhibitory edges,}$$
$$d_{\eta(v_i),i} \leq d_{u_i,i} + x_e \qquad e = (u, v) \in E$$

$$x_e \in \{0, 1\} \qquad \text{for all } e \in E$$

The approximation algorithms for directed multicut given in [44, 93] can be used to solve the negative feedback ILP. As a substep, these algorithms require the solution of the linear program (LP) formed by relaxing the integer constraints in the directed multicut ILP. These approximation algorithms can be trivially modified to solve the negative feedback ILP by relaxing its integer constraints to form a LP. If this change is made to the approximation algorithms in [44, 93], then this gives a heuristic algorithm for solving the negative feedback ILP. We conjecture that is an approximation algorithm for the negative feedback ILP, but the necessary calculations have not been made.

Table 10.1: Number of edge deletions for monotonicity and number of edge deletions required to eliminate negative feedback

Network	Vertices	Edges	Monotone	No Negative Feedback	Time (min.)
EGFR	330	885	210	45	6.5
Yeast	690	1082	41	1	6
Macrophage	678	1582	374	74	12.5

10.3 Results

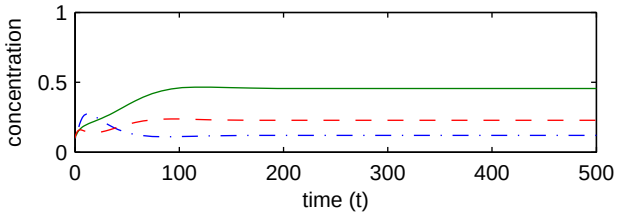
This algorithm was implemented in Python, using the PuLP library to interface with the COIN LP solver [136]. All tests were run on a 2.4 GHz Intel Core 2 Duo MacBook Pro with 2 GB of RAM. The edge-deletion variant of this algorithm was evaluated on the three regulatory networks used in [108]: Macrophage, EGFR, and Yeast. Table 10.1 provides the size of each network, the number of edges that need to be deleted in order to make the system monotone, and the results of the LP relaxation of the negative feedback ILP. For the EFGR and Yeast networks, the optimal LP solutions were integer, but for the Macrophage network the solution was fractional. Rounding up increased the value of the objective function from 66 to 74.

It is interesting to compare the results of removing negative feedback versus the results of removing monotone feedback. Since negative feedback is a subset of monotone feedback, it is expected that less edges would be removed than the approach of [108]. This is what is seen in Table 10.1. The number of edges required to remove negative feedback is significantly less than the number of edges required to remove monotone feedback. This is also interesting, because this approach is a heuristic approach that is not guaranteed to give the true minimum number of edges to remove; [108] uses an algorithm that computes the optimal solution.

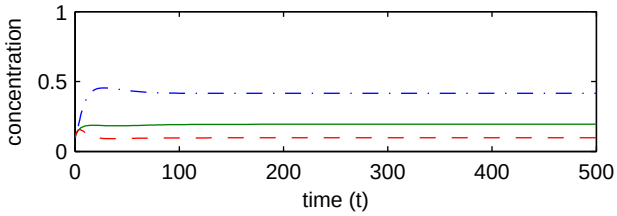
10.3.1 p53 Pathway

Returning to the p53 pathway considered in Chapter 8, it is interesting to see the results of this algorithm on this network. After using the heuristic algorithm on the mutated p53 pathway 10.3b, the algorithm suggests cutting the grayed edge shown in Figure 10.3c. The results are simple, but they are interesting. This is because the edge told to be cut is an edge that biologists have studied in detail and devised chemicals to cut [208, 80]. It is also interesting to compare the results of this algorithm to the control of the same system in [19], because the two controllers superficially look the same but have different behaviors and modalities. In the present chapter, the control removes an edge; whereas in [19], the control removes a vertex. The behaviors of the controlled system are also different, and this can been seen by comparing Figure 10.4 to Figure 8.3.

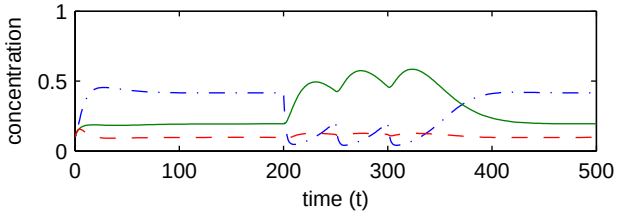
The effect of the control of cutting the grayed edge shown in Figure 10.3c can be seen in Figure 10.4. Time course concentrations of p53, cyclin A, and MDM2 are shown in Figure 10.4a for the normal p53 pathway, Figure 10.4b for the abnormal p53 pathway, and Figure 10.4c for the abnormal p53 pathway with controller. These simulations come from an ODE model of the network, and in the simulations the edge between MDM2 and p53 is removed, but the edge between MDM2 and cyclin A is not removed. In the normal p53 pathway, concentrations of p53 and cyclin A are high, and concentrations of MDM2 are low. In the abnormal p53 pathway, p53 and cyclin concentrations are low, whereas MDM2 is in high concentration. In the abnormal p53 pathway



(a) Normal p53 Pathway



(b) Abnormal p53 Pathway



(c) Abnormal p53 Pathway with Controller

Figure 10.4: The time course plots for the different pathways displays the effect of the abnormality and the controller. Note that p53 is solid, cyclin A is dashed, and MDM2 is dash-dotted.

with controller, the controller is used at times $t = 200$, $t = 250$, and $t = 300$. The controller causes p53 concentrations to increase to higher levels, and reduces MDM2 concentrations. The cyclin A concentration stays at a reduced level. The controller must be used at multiple times, because the controlling drug is modeled to decay. So, the effect of the controller wains as time goes on. If the controller is not applied again, the system returns to an abnormal state.

10.4 Future Problems

A heuristic algorithm for removing the negative feedback from a promotion-inhibition network was presented. This is an important problem because it is an abstraction of the question of which drugs to design to remove stable oscillations from a biological genetic network to a graph-theoretic problem of edge and vertex deletions. Such a method of control is crude, that is it cannot perform specific control actions. However, as seen in the p53 example, the results of the algorithm can be interesting and biologically relevant.

The heuristic algorithm is required because the original problem of edge and vertex removal to remove negative feedback from a promotion-inhibition network is NP-hard. Solving the problem in a reasonable amount of time for large networks requires either the use of heuristic or approximation algorithms. By recognizing that this problem is similar to the directed multicut

problem (which is also NP-hard), existing approximation algorithms [44, 93] — for solving the directed multicut problem — could be modified to solve this problem. We conjecture that this heuristic is an approximation algorithm because it is based on approximation algorithms for a similar problem.

Future works include two aspects. First of all, it would be interested to formally prove whether this algorithm is an approximation algorithm. It will likely be a straight-forward extension of the results of [44, 93]. Second of all, it will be interesting to use this algorithm to study other interesting biological networks.

Chapter 11

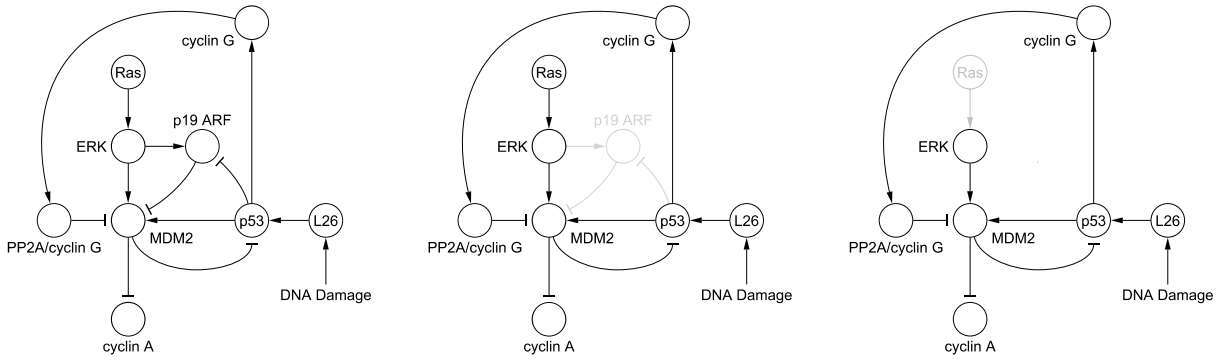
Computer-aided Drug Discovery for Pathway and Genetic Diseases

When designing pharmaceuticals for a disease, researchers currently use their intuition and knowledge about the particular disease modalities to choose which cellular components to target with pharmaceuticals. Such an approach is limited by intuition, and only a few cellular components can be considered at a time. In recent years, there has been a growing recognition on the importance of using multiple drugs, so called "drug cocktails". Unfortunately, simultaneously choosing and designing drugs to target multiple components of a disease modality is even more difficult because of the difficulty in predicting the simultaneous affect of multiple drugs on a large network of cellular components.

This has prompted work on systems theoretic tools for doing drug multi-target selection, such as those presented throughout this second part of the thesis. An important problem facing the health-care industry is how to systematically do drug-discovery. The first aspect of this, is: How should the targets for drugs be chosen? A related problem is: How can drugs be designed to minimize their adverse effects on healthy cells?

Attempts to aid with drug-discovery using a systems-view are relatively recent. Some existing techniques [2, 150, 163] use a biological modeling approach known as flux balance analysis [116]. These approaches identify genes to knock out using gene-therapy, and they typically involve solving the problem of either ensuring viability or ensuring non-viability [2, 163]. The approach in [150] tries to make reactions in a treated, mutated cell occur at the same rate as reactions in an untreated, healthy cell. Other techniques focus on fixing the steady-state behavior of the network [181, 52, 217] by knocking out genes.

There are certain trade-offs inherent in the approaches of [2, 150, 163, 181, 52, 217]. They do not simultaneously consider the impact of the drug treatment on both mutant- and wild-type cells. This is important because real treatments cannot distinguish between healthy and diseased cells. Additionally, these techniques focus on the steady-state behavior of the pathways. Some biological pathways are known to display complicated transient and oscillatory behaviors. Moreover, these approaches cannot deal with the effects of removing a reaction between species without having to manually modify the models; it cannot be done in an automated manner. Though biologically accomplishing this is difficult, some pharmaceutical drugs do this. The approaches in [2, 150, 163] consider reaction rates and not species concentrations. For the function of the cell, the latter is often more important than the former. It can be overly restrictive to ask that the reaction rates between the two types of cells be the same.



(a) Healthy p53 Pathway (b) Diseased p53 Pathway (c) Diseased p53 Pathway with Controller

Figure 11.1: When a subsegment of the healthy p53 pathway [95, 167, 196, 130] becomes diseased, such as loss of p19 ARF function [167], the system behaves unfavorably by underexpressing p53. Using a drug (abstracted as a controller), the grayed portions of the network can be removed to make the system behave more favorably.

In parallel to the efforts of [2, 150, 163, 181, 52, 217], there has been work on similar problems but for pathway-dependent diseases. In [19], it was shown that pharmaceuticals and genetic-therapy are mathematically equivalent to either removing a species from the cell or removing an interaction/reaction between two species. Mathematical theory relating the topology of the pathway with the behavior of the system was used to develop methods to tell which species and interactions to target with drugs in order to make the network behave in a specific manner [19, 12, 17]. The level of control afforded by these techniques is crude, and this has prompted the development of a newer technique which gives a finer level of control on the pathway [21].

This approach requires a researcher to choose what the inputs to and the outputs from a pathway are. The outputs of the pathway can be biomarkers for disease; for instance, in the p53 network shown in Figure 11.1, high MDM2 and low p53 concentration are correlated with cancerous cells [95, 167, 196, 130]. The inputs of the pathway can be triggers of diseases or a direct signal of a negative event; as shown in the p53 pathway in Figure 11.1, L26 is an indicator of DNA damage.

Having chosen inputs and outputs, the desire is to make the diseased pathway with drugs behave similarly to the healthy pathway. The notion of "closeness" is defined by asking that the healthy and diseased pathways have similar input-output behaviors. The benefit of this definition of closeness is that it can be rigorously quantified using a variant of the $\mathcal{L}_2$ norm of the error dynamics. Additionally, there is a desire to make the healthy network with drugs behave similarly to the healthy network with no drugs. In this manner, drugs can be chosen to make a trade-off between adverse effects on healthy cells and positive effects on diseased cells.

This chapter begins with a description of this new computational drug discovery approach. The approach is based on mathematical theory from the field of multivariate control and optimization, and it consequently requires a good mathematical model of the pathway. Choosing the optimal drug-targets can be posed as a combinatorial optimization problem. Scalable approaches for solving this problem, through the use of Monte-Carlo based resampling techniques, are presented. Lastly, examples are given of this approach used on biological examples.

11.1 Problem Setup

11.1.1 Preliminaries

Let $x \in \mathbb{R}^p$, where p is the dimension of the state-space, and let $y \in \mathbb{R}^w$, where w is the number of outputs. The outputs of the pathway are given by the relationship

$$y = Cx, \tag{11.1}$$

where $C \in \mathbb{R}^{w \times p}$. Also, $u(t) \in (\mathcal{L}_2[0,T])^v$ is the set of v-dimensional, square integrable functions with support on $[0,T]$. Furthermore, refer to the i-th component of a vector x as x^i.

Define a slight variation on the notion of the $\mathcal{L}_2$ norm [206, 117]. In particular, define the *T-time $\mathcal{L}_2$ gain* of the following nonlinear system

$$\begin{aligned} \dot{x} &= f(x,u) \\ y &= h(x), \end{aligned} \tag{11.2}$$

as the solution of the following (nonconvex, in general) optimization problem: $\mathcal{L}_{2,T,\epsilon} = \inf \gamma$, subject to

$$\inf_{\substack{\|u\|_2 \geq \epsilon \\ x_0 \in \mathcal{X}}} \int_0^T \left(\gamma^2 \|u\|^2 - \|y\|^2\right) dt \geq 0, \tag{11.3}$$

where $\mathcal{X}$ is the set of initial conditions. The difficulty with the standard $\mathcal{L}_2$ norm [206, 117] is that if a system is not stable then its $\mathcal{L}_2$ norm is infinite. By (a) defining a cutoff time T (granted this cutoff is somewhat arbitrary), (b) defining a minimum input norm ϵ, and (c) not defining a system operating point, the T-time $\mathcal{L}_2$ gain can distinguish between different levels of instability. The notion of decay rate [38] can be used to distinguish between different levels of instability, but it is a cruder notion for the purposes of drug discovery: Two systems with the same decay rate can have different T-time $\mathcal{L}_2$ gains.

11.1.2 Pathway and Pharmaceutical Model

Let $x \in \mathbb{R}^p$ be a vector of species concentrations and $u \in \mathbb{R}^u$ be a vector of inputs. The i-th component of this vector is denoted x^i, and it always refers to the same species. Subscripts are used to differentiate amongst different conditions (e.g., healthy, diseased, etc.) on the pathway. Recall that the output of the pathway is given by $y = Cx$, and the output is the same regardless of the conditions on the pathway. This is because C is a matrix which reads out certain species concentrations, and it has the form that each row of C is all zeros except for one column (this column is different for different rows) which is one. An example of a C matrix is

$$C = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{11.4}$$

Differentially weighting the importance of certain outputs is a simple extension that requires weighting the rows of C.

Assume that the healthy pathway obeys the dynamics

$$\dot{x}_h = f_h(x_h, u), \tag{11.5}$$

where the subscript h denotes that the quantities are those of the healthy pathway. Similarly, suppose that the diseased pathway obeys the dynamics $\dot{x}_d = f_d(x_d, u)$, where the subscript d

denotes that the quantities are those of the diseased pathway. Denote the dynamics of the diseased pathway with drug treatments as $\dot{x}_s = f_s(x_s, u)$, and the error is given by

$$e_s(t) = y_h(t) - y_s(t). \tag{11.6}$$

Similarly, the dynamics of the healthy pathway with drug treatments is denoted as $\dot{x}_r = f_r(x_r, u)$, and the error is given by

$$e_r(t) = y_h(t) - y_r(t). \tag{11.7}$$

Following the approach in [19], abstract the effect of genetic therapy and pharmaceuticals as removing species or reactions from the network. From a biochemistry standpoint, most pharmaceuticals remove species from the network. This is because it is easier to prevent a species from being created than it is to prevent a reaction between two species from occurring. This is not to say that there are no drugs which prevent reactions, just that their biochemical design could be exceedingly difficult.

Mathematically speaking, a drug that removes a species from the network can be modeled in two ways. The first way is to enforce an algebraic constraint on the system. For instance, removing the i-th species using a pharmaceutical is algebraically represented as ${x_r}^i = 0$. The second way is to modify the dynamics of the network by adding a new term to the vector field:

$$\dot{{x_r}}^i = f_d(x_r, u)^i - \lambda {x_r}^i, \tag{11.8}$$

where λ is a very large value. In this equation, the vector field $\dot{x}_r = f_r(x_r, u)$ is represented in terms of the vector field $f_d(x_r, u)$.

The second way of modeling the removal of a species from the network is essentially the same as enforcing an algebraic constraint on the system, but it enables posing the drug-discovery problem in a more natural optimization framework. Let v be a p-dimensional vector whose entries are either 0 or 1. For a p-dimensional vector-field $f(x, u)$, define the operator $R(\cdot)$ as:

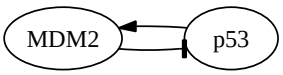
$$R(f(x, u))^i = f(x, u)^i - \lambda x^i, \tag{11.9}$$

where λ is a very large value. The drug-discovery problem can be posed as selecting the values of v subject to a minimization objective. Note that species i can be disallowed from removal by forcing the constraint $v^i = 0$.

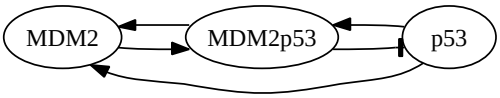
11.1.3 Interaction Removal

Removing an interaction (typically a reaction) between two species is biologically more difficult. It is not surprising that it is also difficult to do this in a general norm minimization framework. The most general method for doing this is to add more states to the system and then accordingly modifying the ODE model. By carefully adding states to the system, allowing reactions to be removed is equivalent to removing species from the network. This can be best explained with an example. In this example, the added state has a biological basis; however, it is not necessary to have a biological basis for adding these additional states.

Consider the p53 pathway from Chapters 8 and 10, which is shown in Figure 11.1. It is known [208] that MDM2 and p53 bind to form a complex (for the purposes of the discussion here, this will be called MDM2p53). This complex then signals p53 to be degraded by the cell. Instead of directly modeling this inhibition of p53 by MDM2, an additional state can be added. Figure 11.2a shows the simplest model in which these interactions are modeled as being direct. On the



(a) Direct Interaction



(b) Appended Interaction

Figure 11.2: The interaction between MDM2 and p53 can be modeled either by a direct interaction or with an appended interaction which adds a state to model the reaction that forms the MDM2p53 complex.

other hand, Figure 11.2b shows a model in which the complex MDM2p53 is added to model the interaction between MDM2 and p53. If the species MDM2p53 is removed from the network, then this removes the interaction going from MDM2 to p53; the interaction from p53 to MDM is not removed by removing MDM2p53.

11.1.4 $\mathcal{L}_2$ Norm Minimization Problem

The problem of choosing the optimal targets for drugs can be written as a non-convex optimization problem. For a given weight $\alpha \in [0,1]$ and value of $T > 0$, solve:

$$\inf_{v} \quad \alpha\gamma_r + (1-\alpha)\gamma_s, \tag{11.10}$$

subject to:

1. $\dot{x}_h = f_h(x_h, u)$ and $y_h = Cx_h$
2. $\dot{x}_d = f_d(x_d, u)$ and $y_d = Cx_d$
3. $\dot{x}_r = R(f_h(x_r, u))$ and $y_r = Cx_r$
4. $\dot{x}_s = R(f_d(x_s, u))$ and $y_s = Cx_s$
5. $x_0 \in \mathcal{X}$
6. $u(t) \in (\mathcal{L}_2[0,T])^v$ and $\|u(t)\| > \epsilon$
7. $\inf \int_0^T \left(\gamma_r^2 \|u\|^2 - \|y_h - y_r\|^2\right) \geq 0$
8. $\inf \int_0^T \left(\gamma_s^2 \|u\|^2 - \|y_h - y_s\|^2\right) \geq 0$
9. $v^i \in \{0,1\}$ for $i \in V$
10. $v^i = 0$, for $i \notin V$

where $V \subseteq \{1, \ldots, p\}$ are the indices of the species that can be removed by a pharmaceutical and $\|\cdot\|$ is the ℓ_2 norm.

The optimization problem given in (11.10) means minimizing a weighted sum of (a) T-time $\mathcal{L}_2$ norm of the error between a healthy pathway and a diseased pathway with drug-treatment, and (b) T-time $\mathcal{L}_2$ norm of the error between a healthy pathway and the healthy pathway with drug treatment. This provides a trade-off between adverse effects on healthy cells and positive effects on diseased cells, because the $\mathcal{L}_2$ norm of the error dynamics gives a measure of the deviation between the input-output behavior of the healthy cell and the diseased cell with drug-treatment. As stated earlier, the minimization in (11.10) is achieved by selection of the values of the vector v which indicate which species to remove from the network.

11.2 Solving the $\mathcal{L}_2$ Norm Minimization Problem

The optimization problem defined in (11.10) is a difficult problem to solve because it is a combinatorial optimization problem with non-convex constraints. Because biological pathways can be large, it is important that any method used to solve the problem be scalable. In order to solve the problem efficiently, Monte-Carlo-based heuristics can be used; there are two approaches to take. The first approach is a brute-force, exhaustive-search approach, and this is best-suited for the "optimal drug cocktail selection" problem when the cardinality of V is small. The second approach is a greedy-search approach, and it is best suited for the "drug-discovery" problem when the cardinality of V is large.

11.2.1 Infeasibility of Linearization-Based Approaches

Before discussing heuristic methods for solving $\mathcal{L}_2$ norm minimization problem, it is worth discussing one approach which will not work well. Imagine trying to linearize the error dynamics $\dot{e}_r = C(\dot{x}_h - \dot{x}_r)$, and then using linear theory to compute a norm of the error-dynamics. The most common norms for linear systems are the $\mathcal{H}_2$ and $\mathcal{H}_{\text{inf}}$ (for linear systems this is equivalent to the $\mathcal{L}_2$ norm), but a problem is that both norms require that the system be asymptotically stable. This is not true in general because the operator V changes the dynamics to

$$R(f(x,u))^i = a_i x^i + b^i - \lambda(x^i - {x_0}^i), \tag{11.11}$$

where x_0 is the operating point of the linearization.

The second approach is to use the notion of decay rate [38] (as opposed to $\mathcal{L}_2$ norm) to compute the size of the error. If $e_r(t)$ is the error term, then the decay rate α is defined as

$$\inf \alpha, \text{such that } \lim_{t\to\infty} e^{-\alpha t} e_r(t) = 0. \tag{11.12}$$

This is a quasi-convex optimization for systems with linear dynamics, and efficient algorithms exist for solving this [38]. Imagine using this notion of distance in the norm minimization problem, and the optimal v of species to remove with drugs could then be computed using an algorithm similar to the $\mu - K$ iterations ubiquitous to multivariate linear control. The problem with this approach is that the nonlinear system is stable for all values of v (this can be shown using an argument involving positively invariant regions). Thus, the decay rate for all values of v is identically $\alpha = 0$, even though the size of the error could be uniformly smaller for some values of v. This situation is even worse for the linearized case where the decay rate calculated will be an upper bound on the true value: The decay rate will have no relation to the size of the error.

Algorithm 1: Brute-Force Search for Optimal Drug Cocktail Selection

input : V,α, $\mathcal{X} \subset \mathbb{R}^p$, $\mathcal{U} \subset \mathbb{R}^v$, $n \in \mathbb{R}_+$
output: γ, $I \subseteq V$

foreach $j = 1, \ldots, n$ **do**
  set x_j :=random vector from $\mathcal{X}$;
  set u_j :=random vector from $\mathcal{U}$;
end
foreach $I \subseteq V$ **do**
  $v^i = 1, \forall i \in I$;
  calculate $\gamma_r(I)$ **subject to**
    $\gamma_r = \max_{j=\{1,\ldots,n\}} \int_0^T \|y_{h,j} - y_{r,j}\|_2{}^2 / \|u_j\|_2{}^2 dt$;
  end
  calculate $\gamma_s(I)$ **subject to**
    $\gamma_s = \max_{j=\{1,\ldots,n\}} \int_0^T \|y_{h,j} - y_{s,j}\|_2{}^2 / \|u_j\|_2{}^2 dt$;
  end
end
set $\gamma := \min_I \alpha\gamma_r + (1-\alpha)\gamma_s$;
set $I := \arg\min_I \alpha\gamma_r + (1-\alpha)\gamma_s$;
return γ, I

11.2.2 Optimal Drug Cocktail Selection Problem

Recall that in the "optimal drug cocktail selection" problem, the cardinality of V is small. A biological example of this problem is the treatment of a cancer in which there are 10 drugs whose effects on a pathway are known, and a researcher would like to choose the subset of drugs that provides the best clinical outcome. Since the cardinality of V is small in such problems, we can use an exhaustive-search to choose the optimal set of drugs as shown in Algorithm 1.

The basic idea is to randomly sample points from $\mathcal{X} \subset \mathbb{R}^p$ which contains all possible initial conditions and randomly sample from a subset $\mathcal{U} \subset \mathbb{R}^u$ which contains all possible input values. These randomly sampled points u_j for $j = 1, \ldots, n$ are used as inputs to the system, and the T-time $\mathcal{L}_2$ gain of the system errors are computed for each value of u_j.

11.2.3 Drug Discovery Problem

Recall that in the "drug discovery selection" problem, the cardinality of V is large. A biological example of this problem is the treatment of a cancer in which a researcher would like to determine a best set of targets for drugs, such that the clinical outcome of the cancer is positive. Since the cardinality of V is large in such problems, exhaustive-search cannot be used. However, a backwards greedy-search can be used. This is shown in Algorithm 2. The basic idea is to remove one species or interaction at a time, until there is no improvement in the weighted sum of the T-time $\mathcal{L}_2$ norms. This heuristic method allows for the computations to scale up on very large pathways.

Algorithm 2: Greedy-Search for Drug Discovery Problem

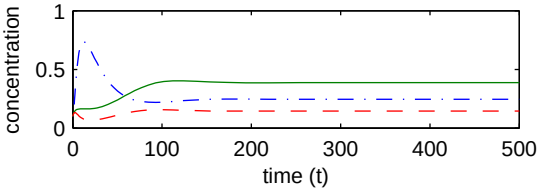
input : V,α, $\mathcal{X} \subset \mathbb{R}^p$, $\mathcal{U} \subset \mathbb{R}^v$, $n \in \mathbb{R}_+$
output: γ, $I \subseteq V$

foreach $j = 1, \ldots, n$ **do**
  set x_j :=random vector from $\mathcal{X}$;
  set u_j :=random vector from $\mathcal{U}$;
end
set $I := \{\emptyset\}$;
repeat
  set $\tilde{V} := V \setminus I$;
  foreach $a \in \tilde{V} \cup \{\emptyset\}$ **do**
    $v^i = 1, \forall i \in I$;
    $v^a = 1$;
    calculate $\gamma_r(a)$ **subject to**
      $\gamma_r = \max_{j=\{1,\ldots,n\}} \int_0^T \|y_{h,j} - y_{r,j}\|_2{}^2 / \|u_j\|_2{}^2 dt$;
    end
    calculate $\gamma_s(a)$ **subject to**
      $\gamma_s = \max_{j=\{1,\ldots,n\}} \int_0^T \|y_{h,j} - y_{s,j}\|_2{}^2 / \|u_j\|_2{}^2 dt$;
    end
    set $\gamma := \min_a \alpha\gamma_r + (1-\alpha)\gamma_s$;
    set $a := \arg\min_a \alpha\gamma_r + (1-\alpha)\gamma_s$;
    set $I := I \cup a$;
  end
until $a = \{\emptyset\}$;
return γ, I

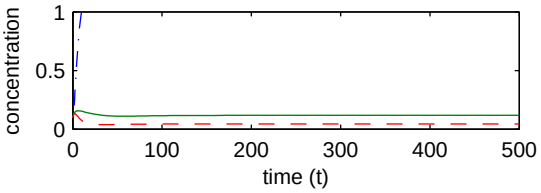
11.3 Example: p53 Pathway

Returning to the p53 pathway considered in Chapters 8 and 10, it is interesting to see the results of this algorithm on this network. This subsystem of the p53 pathway is small enough that Algorithm 1 can be used to generate new strategies for the treatment of the mutated pathway. Algorithm 2 can also be used to do drug discovery. In the presence of no constraints on V, the results of Algorithm 1 will always be better than or equal to those of Algorithm 2. In these experiments, $\alpha = 0.1$ was picked to ensure that the healthy, normal pathway with drug treatment did not deviate too much from the normal pathway with no treatment. The input controls the species concentration of L26, and the outputs are the concentrations of MDM2, p53, and cyclin A.

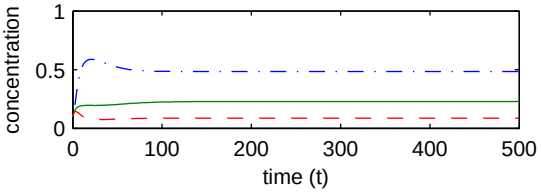
The effect of the control of cutting the grayed edge shown in Figure 11.1c can be seen in Figure 11.3. Time course concentrations of p53, cyclin A, and MDM2 are shown in Figure 11.3a for the normal p53 pathway, Figure 11.3b for the abnormal p53 pathway, Figure 11.3c for the abnormal p53 pathway with controller, and Figure 11.3d for the normal p53 pathway with controller. These simulations come from an ODE model of the network, and in the simulations with controllers, the species Ras is removed from the pathway through the use of a hypothetical, pharmaceutical drug. In the normal p53 pathway, concentrations of p53 and cyclin A are high, and concentrations of MDM2 are low. In the abnormal p53 pathway, p53 and cyclin A concentrations are low, whereas



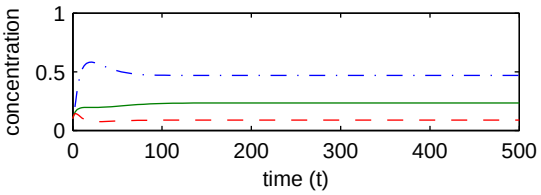
(a) Healthy p53 Pathway



(b) Diseaesd p53 Pathway



(c) Diseased p53 Pathway with Controller



(d) Healthy p53 Pathway with Controller

Figure 11.3: The time course plots for the different pathways displays the effect of the disease and the pharmaceutical drug controller. Note that p53 is solid, cyclin A is dashed, and MDM2 is dash-dotted.

MDM2 is in high concentration. The controller causes p53 concentrations to increase to higher levels, and reduces MDM2 concentrations. The cyclin A concentration stays at a reduced level. In the normal p53 pathway with controller, the network is not as well behaved as the normal p53 pathway; however, it is also not as poorly behaved as the abnormal p53 pathway.

Chapter 12

Conclusion

This thesis presents systems theoretic tools for doing pharmaceutical drug discovery. The first class of tools is used to identify mathematical models, and these models are then used in conjunction with the second class of tools which selects potential drug targets in order to achieve good therapeutic outcomes for patients.

The technical implications of the tools and potential future works were, for the most part, presented at the end of each chapter, including a fairly clear path for the refinement of the technical aspects of this systems theory. Yet, these tools will need to be extended to deal with larger and more complex data sets. Newer techniques of statistical system identification will need to be developed which can incorporate different types of experimental data (e.g., expression levels, protein-DNA binding, etc.) and varying levels of prior knowledge about the biological network. Drug multi-target identification tools will need to be extended to be able to handle more realistic models of the effects of drugs on their targets, and this will itself involve doing system identification. A tighter integration between the identification and target selection is also needed, meaning the target selection tools should be able to somehow compensate for the limitations of identified models.

The tools presented in this thesis also have applications to engineering systems, and this is another avenue for future work. The statistical system identification work and the estimators presented have applications to diverse areas such as the social sciences, computer vision, robotics, and autonomous systems. The area of robotics and autonomous systems is particularly interesting to a systems theorist, and discovering how to handle the limitations of learning techniques, as identified by our theoretical analysis, will be necessary for being able to push learning in robotics significantly beyond the state of the art. On the other hand, the topology control tools used for drug multi-target identification might have applications to engineering systems such as in traffic engineering or multi-robot systems. Extending these tools to engineering systems will be challenging because of the specialized nature of topological control, but it seems promising because of the similarity in mathematical structure and organization of these systems with respect to biological networks.

Going beyond this, the real challenge will be in applying these techniques to real, biological disease networks. The innovative and open culture in academic and research institutions is ideal for doing this, because application will require collaborations between engineers, statisticians, biologists, medical doctors, and others. Such application will be the true driver for the development of a full systems theory for doing pharmaceutical drug discovery.

Bibliography

[1] Paul Adler. Planar signalling and morphogenesis in *Drosophila*. *Developmental Cell*, 2(5):525–535, 2002.

[2] A. Agung Julius, Marcin Imielinski, and George J. Pappas. Metabolic networks analysis using convex optimization. In *Proceedings of 47th IEEE CDC*, pages 762–767, Cancun, Dec. 2008.

[3] P.W. Aitchison. Generalized inverse matrices and their applications. *International Journal of Mathematical Education in Science and Technology*, 13(1):99–109, 1982.

[4] F. Alizadeh and D. Goldfarb. Second-order cone programming. Technical Report RRR 51-2001, Rutgers University, 2001.

[5] Keith Amonlirdviman, Narmada Khare, David Tree, Wei-Shen Chen, Jeffrey Axelrod, and Claire Tomlin. Mathematical modeling of planar cell polarity to understand domineering nonautonomy. *Science*, 307(5708):423–426, 2005.

[6] Martin Andersson. A comparison of nine PLS1 algorithms. *Journal of Chemometrics*, 2009.

[7] David Angeli and Eduardo D. Sontag. Multi-stability in monotone input/output systems. *Systems and Control Letters*, 51(3-4):185–202, March 2004.

[8] A. Argyriou, T. Evgeniou, and M. Pontil. Convex multi-task feature learning. *Machine Learning*, 73(3):243–272, 2008.

[9] D.N. Arnosti, S. Barolo, M. Levine, and S. Small. The eve stripe 2 enhancer employs multiple modes of transcriptional synergy. *Development*, 122(1):205–214, 1996.

[10] Margaret Ashcroft and Karen H. Vousden. Regulation of p53 stability. *Oncogene*, 18:7637–7643, 1999.

[11] Anil Aswani. Reachability algorithm for a class of biologically inspired piecewise-affine hybrid systems. Master's thesis, University of California at Berkeley, 2007.

[12] Anil Aswani, Peter Bickel, and Claire Tomlin. Statistics for sparse, high-dimensional, and nonparametric system identification. In *International Conference on Robotics and Automation 2009*, pages 2133–2138, 2009.

[13] Anil Aswani, Peter Bickel, and Claire Tomlin. Learning global-sparsity structure. In preparation, 2010.

[14] Anil Aswani, Peter Bickel, and Claire Tomlin. Regression on manifolds: Estimation of the exterior derivative. *Annals of Statistics*, 2010. Accepted.

[15] Anil Aswani, Nicholas Boyd, and Claire Tomlin. Graph-theoretic topological control of biological genetic networks. In *Proceedings of the American Control Conference 2009*, pages 1700–1705, 2009.

[16] Anil Aswani, Harendra Guturu, and Claire Tomlin. System identification of Hunchback protein patterning in early *Drosophila* embryogensis. In *Proceedings of 48th IEEE CDC*, pages 7723–7728, 2009.

[17] Anil Aswani, Soile Keränen, James Brown, Charless Fowlkes, David Knowles, Mark Biggin, Peter Bickel, and Claire Tomlin. Nonparametric identification of regulatory interactions from spatial and temporal gene expression data. Submitted, 2010.

[18] Anil Aswani and Claire Tomlin. Reachability algorithm for biological piecewise-affine hybrid systems. In *Hybrid Systems: Computation and Control 2007*, pages 633–636, 2007.

[19] Anil Aswani and Claire Tomlin. Topology based control of biological genetic networks. In *Proceedings of 47th IEEE Conference on Decision and Control*, pages 781–786, 2008.

[20] Anil Aswani and Claire Tomlin. Monotone piecewise affine systems. *IEEE Transactions on Automatic Control*, 54(8):1913–1918, 2009.

[21] Anil Aswani and Claire Tomlin. Computer-aided drug discovery for pathway and genetic diseases. Submitted, 2010.

[22] Chris Atkeson, Andrew Moore, and Stefan Schaal. Locally weighted learning for control. *AI Review*, 11:75–113, 1997.

[23] Jeffrey Axelrod. Unipolar membrane association of Dishevelled mediates Frizzled planar cell polarity signaling. *Genes & Development*, 15:1182–1187, 2001.

[24] I. Baianu. Computer models and automata theory in biology and medicine. *Mathematical Modelling*, 7:1513–1577, 1987.

[25] Mukesh Bansal, Vincenzo Belcastro, Alberto Ambesi-Impiombato, and Diego di Bernardo. How to infer gene networks from expression profiles. *Molecular Systems Biology*, 3(78), 2007.

[26] Rebecca Bastock, Helen Strutt, and David Strutt. Strabismus is asymmetrically localised and binds to Prickle and Dishevelled during *Drosophila* planar polarity patterning. *Development*, 130:3007–3014, 2003.

[27] Mokhtar Bazaraa, Hamif Sherali, and C.M. Shetty. *Nonlinear Programming: Theory and Algorithms*. Wiley, 1993.

[28] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7:2399–2434, 2006.

[29] Calin Belta, Joel Esposito, Jongwoo Kim, and Vijay Kumar. Computational techniques for analysis of genetic network dynamics. *International Journal of Robotics Research*, 24(2-3), 2005.

[30] Rajendra Bhatia. *Perturbation Bounds for Matrix Eigenvalues*. Society for Industrial and Applied Mathematics, 2007.

[31] D. Bickel. Probabilities of spurious connections in gene networks: application to expression time series. *Bioinformatics*, 21(7):1121–1128, 2005.

[32] P. Bickel and E. Levina. Covariance regularization by thresholding. *Annals of Statistics*, 36(6):2577–2604, 2008.

[33] P. Bickel and B. Li. Local polynomial regression on unknown manifolds. In *Complex Datasets And Inverse Problems: Tomography, Networks And Beyond*, volume 54 of *IMS Lecture Notes-Monograph Series*. Institute of Mathematical Studies, 2007.

[34] P. J. Bickel and K. A. Doksum. *Mathematical Statistics*. Prentice Hall, 2006.

[35] Peter Bickel and David Freedman. Some asymptotic theory for the bootstrap. *Annals of Statistics*, 9(6):1196–1217, 1981.

[36] H. Bolouri and E. H. Davidson. Modeling transcriptional regulatory networks. *Bioessays*, 24:1118–1129, 2002.

[37] Richard Bonneau, David Reiss, Paul Shannon, Marc Facciotti, Leroy Hood, Nitin Baliga, and Vesteinn Thorsson. The Inferelator: an algorithm for learning parsimonious regulatory networks from systems-biology data sets de novo. *Genome Biology*, 7(5):R36, 2006.

[38] Stephen Boyd, Laurent El Ghaoui, Eric Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. Society for Industrial and Applied Mathematics, 1994.

[39] Kenneth P. Burnham and David R. Anderson. Multimodel inference: Understanding AIC and BIC in model selection. *Sociological Methods Research*, 33(2):261–304, 2004.

[40] A. Butte and I. Kohane. Mutual information relevance networks: Functional genomic clustering using pairwise entropy measurements. In *Pacific Symposium on Biocomputing*, pages 418–429, 2000.

[41] Frank M. Callier and Charles A. Desoer. *Linear System Theory*. Springer-Verlag, 1991.

[42] E. Candes and T. Tao. The Dantzig selector: Statistical estimation when p is much larger than n. *Annals of Statistics*, 35(6):2313–2351, 2007.

[43] R. Carroll, J. Maca, and D. Ruppert. Nonparametric regression in the presence of measurement error. *Biometrika*, 86(3):541–554, 1999.

[44] Joseph Cheriyan, Howard Karloff, and Yuval Rabani. Approximating directed multicuts. In *42nd IEEE Symposium on Foundatiosn of Computer Science*, 2001.

[45] Alongkrit Chutinan and Bruce H. Krogh. Verification of polyhedral-invariant hybrid automata using polygonal flow pipe approximations. In *HSCC*, pages 76–90, 1999.

[46] J. Chuzhoy and S. Khanna. Hardness of cut problems in directed graphs. In *STOC '06: Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 527–536, New York, NY, USA, 2006. ACM.

[47] O. Cinquin and J. Demongeot. Positive and negative feedback: striking a balance between necessary antagonists. *Journal of Theoretical Biology*, 216:229–241, 2002.

[48] J. Costa and A. Hero. Geodesic entropic graphs for dimension and entropy estimation in manifold learning. *IEEE Transactions on Signal Processing*, 52(8):2210–2221, 2004.

[49] O. Crauk and N. Dostatni. Bicoid determines sharp and precise target gene expression in the *Drosophila* embryo. *Current Biology*, 15:1888–1898, 2005.

[50] B. DasGupta, G. Enciso, E. Sontag, and Y. Zhang. Algorithmic and complexity results for decompositions of biological networks into monotone subsystems. *Biosystems*, 90(1):161–178, 2007.

[51] A. d'Aspremont, L. El Ghaoui, M. Jordan, and G. Lanckriet. A direct formulation of sparse PCA using semidefinite programming. *SIAM Review*, 49(3), 2007.

[52] A. Datta, R. Pal, A. Choudhary, and E. Dougherty. Control approaches for probabilistic gene regulatory networks. *IEEE Signal Processing Magazine*, 24(1):54–63, 2007.

[53] H. de Jong. Modeling and simulation of genetic regulatory systems: a literature review. *Journal of Computational Biology*, 9:67–103, 2002.

[54] H. de Jong, J.-L. Gouzé, C. Hernandez, M. Page, T. Sari, and J. Geiselmann. Qualitative simulation of genetic regulatory networks using piecewise-linear models. *Bulletin of Mathematical Biology*, 66(2):301–340, March 2004.

[55] H. de Jong and D. Ropers. Qualitative approaches towards the analysis of genetic regulatory networks. In Z. Szallasi, V. Periwal, and J. Stelling, editors, *System Modeling in Cellular Biology : From Concepts to Nuts and Bolts*, pages 125–148. MIT Press, 2006.

[56] T.L. Deans, C.R. Cantor, and J.J. Collins. A tunable genetic switch based on RNAi and repressor proteins for regulating gene expression in mammalian cells. *Cell*, 130:363–372, 2007.

[57] D. Del Vecchio, M. Malisoff, and R. Verma. A separation principle for a class of hybrid automata on a partial order. In *Proceedings of the ACC*, pages 3638–3643, 2009.

[58] D. Del Vecchio, R. M. Murray, and E. Klavins. Discrete state estimators for systems on a lattice. *Automatica*, 42:271–285, 2006.

[59] P. D'haeseleer, X. Wen, S. Fuhrman, and R. Somogyi. Linear modeling of mRNA expression levels during CNS development and injury. In *Pacific Symposium on Biocomputing*, pages 41–52, 1999.

[60] J. Dormand and P. Prince. A family of embedded Runge-Kutta formulae. *Journal of Computational and Applied Mathematics*, 6(1):19–26, 1980.

[61] Wolfgang Driever and Christiane Nüsslein-Volhard. The Bicoid protein is a positive regulator of *hunchback* transcription in the early *Drosophila* embryo. *Nature*, 337:138–143, 1989.

[62] Bardley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. Least angle regression. *Annals of Statistics*, 32(2):407–499, 2004.

[63] M. Eisen, P. Spellman, P. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *PNAS*, 98:14863–14868, 1998.

[64] G.A. Enciso and E.D. Sontag. A remark on multistability for monotone systems. In *Proceedings of 43rd IEEE CDC*, pages 249–254, Paradise Island, Bahamas, Dec. 2004.

[65] Walid Fakhouri, Ahmet Ay, Rupinder Sayal, Jacqueline Dresch, Evan Dayringer, and David Arnosti. Deciphering a transcriptional regulatory code: modeling short-range repression in the *Drosophila* embryo. *Molecular Systems Biology*, 6(341), 2010.

[66] J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001.

[67] J. Fan and Y. Truong. Nonparametric regression with errors in variables. *Annals of Statistics*, 21(4):1900–1925, 1993.

[68] Jianqing Fan and Irène Gijbels. *Local polynomial modelling and its applications*. CRC Press, 1996.

[69] E. Farcot and J.-L. Gouzé. Periodic solutions of piecewise affine gene network models: the case of a negative feedback loop. Technical Report 6018, INRIA, http://hal.inria.fr/inria-00112195, 2006.

[70] Joel Ferziger and Milovan Perić. *Computational Methods for Fluid Dynamics*. Springer-Verlag, 1999.

[71] Alekseĭ Fedorovich Filippov. *Differential equations with discontinuous righthand sides*. Kluwer Academic Publishers, 1988.

[72] C. Fowlkes, C. Luengo Hendriks, S. Keränen, G. Weber, O. Rübel, M-Y Huang, S. Chatoor, L. Simirenko, A. DePace, C. Henriquez, A. Beaton, R. Weiszmann, S. Celniker, B. Hamann, D. Knowles, M. Biggin, M. Eisen, and J. Malik. Constructing a quantitative spatio-temporal atlas of gene expression in the *Drosophila* blastoderm. *Cell*, 133(2):364–374, 2008.

[73] Ildiko E. Frank and Jerome H. Friedman. A statistical view of some chemometrics regression tools. *Technometrics*, 35(2):109–135, 1993.

[74] Jerome Friedman, Trevor Hastie, Holger Höfling, and Robert Tibshirani. Pathwise coordinate optimization. *Annals of Applied Statistics*, 1(2):302–332, 2007.

[75] N. Friedman, M. Linial, I. Nachman, and D. Pe'er. Using Bayesian networks to analyze expression data. *Journal of Computational Biology*, 7(3):601–620, 2000.

[76] Nir Friedman. Inferring cellular networks using probabilistic graphical models. *Science*, 303(5659):799–805, 2004.

[77] Wenjiang Fu. Ridge estimator in singular design with application to age-period-cohort analysis of disease rates. *Communications in statistics - Theory and Methods*, 29(2):263–278, 2000.

[78] Wenjiang Fu. A smooth cohort model in age-period-cohort analysis with applications to homicide arrest rates and lung cancer mortality rates. *Sociological Methods and Research*, 36(3):327–361, 2008.

[79] M. Fujioka, Y. Emi-Sarker, G.L. Yusibova, T. Goto, and J.B. Jaynes. Analysis of an even-skipped rescue transgene reveals both composite and discrete neuronal and early blastoderm enhancers, and multi-stripe positioning by gap gene repressor gradients. *Development*, 126:2527–2538, 1999.

[80] Gitali Ganguli and Bodhan Wasylyk. p53-independent functions of MDM2. *Molecular Cancer Research*, 1:1027–1035, 2003.

[81] U. Gaul and H. Jäckle. Analysis of maternal effect mutatnt combinations elucidates regulation and function of the overlap of hunchback and krüppel gene expression in the *Drosophila* blastoderm embryo. *Development*, 107:651–662, 1989.

[82] C. Geyer. On the asymptotics of constrained m-estimation. *Annals of Statistics*, 22:1993–2010, 1994.

[83] Ronojoy Ghosh and Claire Tomlin. Symbolic reachable set computation of piecewise affine hybrid automata and its application to biological modelling: Delta-Notch protein signalling. *Systems Biology*, 1(1):170–183, June 2005.

[84] Leon Glass. Combinatorial and topological methods in nonlinear chemical kinetics. *J. Chem. Phys*, 63(4):1325–1335, 1975.

[85] Bernard Glick and Jack Pasternak. *Molecular Biotechnology: Principles and Applications of Recombinant DNA*. American Society of Microbiology Press, 2003.

[86] Yair Goldberg, Alon Zakai, Dan Kushnir, and Ya'acov Ritov. Manifold learning: The price of normalization. *Journal of Machine Learning Research*, 9:1909–1939, 2008.

[87] Gene Golub and Charles Van Loan. *Matrix computations*. JHU Press, 3 edition, 1996.

[88] Gabriel Gomes, Roberto Horowitz, Alex Kurzhanskiy, Jaimyoung Kwon, and Pravin Varaiya. Behavior of the cell transmission model and effectiveness of ramp metering. *Transportation Research Part C*, pages 485–513, 2007.

[89] Jean-Luc Gouzé. Positive and negative circuits in dynamical systems. *Journal of Biological Systems*, 6:11–15, 1998.

[90] Jean-Luc Gouzé and Tewfix Sari. A class of piecewise linear differential equations arising in biological models. *Dynamical Systems*, 17(4):299–316, 2002.

[91] T. Gregor, E. Wieschaus, A. McGregor, W. Bialek, and D. Tank. Stability and nuclear dynamics of the bicoid morphogen gradient. *Cell*, 130:141–152, 2007.

[92] Thomas Gregor, David W. Tank, Eric F. Wieschaus, and William Bialek. Probing the limits to positional information. *Cell*, 130(1):153–164, 2007.

[93] Anupam Gupta. Improved results for directed multicut. In *14th Annual ACM-SIAM symposium on Discrete algorithms*, 2003.

[94] K. Harding, T. Hoey, R. Warrior, and M. Levine. Autoregulatory and gap gene response elements of the even-skipped promoter of *Drosophila*. *EMBO J.*, 8(4):1205–1212, 1989.

[95] Sandra L. Harris and Arnold J. Levine. The p53 pathway: positive and negative feedback loops. *Oncogene*, 24:2899–2908, 2005.

[96] F. He, Y. Wen, J. Deng, X. Lin, and L. Lu. Probing intrinsic properties of a robust morphogen gradient in *Drosophila*. *Developmental Cell*, 15:558–567, 2008.

[97] Matthias Hein and Jean-Yves Audibert. Intrinsic dimensionality estimation of submanifolds in $\mathbb{R}^d$. In *ICML*, pages 289–296, 2005.

[98] Inge Helland. On the structure of partial least squares regression. *Communications in statistics - Simulation and Computation*, 17(2):581–607, 1988.

[99] C. Helmberg and F. Rendl. A spectral bundle method for semidefinite programming. *SIAM Journal of Optimization*, 10:673–696, 2000.

[100] Harold Henderson and Paul Velleman. Building multiple regression models interactively. *Biometrics*, 37(2):391–411, 1981.

[101] M. W. Hirsch and Hal Smith. Monotone dynamical systems. In A. Canada, P. Drabek, and A. Fonda, editors, *Handbook of Differential Equations: Ordinary Differential Equations*, volume 2, pages 239–357. Elsevier, 2005.

[102] Morris W. Hirsch. Stability and convergence in strongly monotone dynamical systems. *Journal für die reine und angewandte Mathematik*, 383:1–53, 1988.

[103] John G. Hocking and Gail S. Young. *Topology*. Courier Dover Publications, 1961.

[104] R. Hocking. The analysis and selection of variables in linear regression. *Biometrics*, 32:431–453, 1976.

[105] A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 8:27–51, 1970.

[106] Gabriel M. Hoffmann, Haomiao Huang, Steven L. Waslander, and Claire J. Tomlin. Quadrotor helicopter flight dynamics and control: Theory and experiment. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Hilton Head, SC, 2007. AIAA Paper 2007-6461.

[107] Gabriel M. Hoffmann, Steven L. Waslander, and Claire J. Tomlin. Quadrotor helicopter trajectory tracking control. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Honolulu, HI, 2008. AIAA Paper Nuber 2008-7410.

[108] F. Hüffner, N. Betzler, and R. Niedermeier. Optimal edge deletions for signed graph balancing. In *Experimental Algorithms*, pages 297–310. Springer, 2007.

[109] K. J. Hunt and T. A. Johansen. Design and analysis of gain-scheduled control using local controller networks. *International Journal of Control*, 66(5):619–652, 1997.

[110] D. Hunter and K. Lange. A tutorial on MM algorithms. *The American Statistician*, 58(1):30–37, 2004.

[111] Herbert Jäckle, Diethard Tautz, Reinhard Schuh, Eveline Seifert, and Ruth Lehmann. Cross-regulatory interactions among the gap genes of *Drosophila*. *Nature*, 324:668–670, 1986.

[112] Johannes Jaeger, Maxim Blagov, David Kosman, Konstantin Kozlov, Manu, Ekaterina Myasnikova, Svetlana Surkova, Carlos Vanario-Alonso, Maria Samsonova, David Sharp, and John Reinitz. Dynamical analysis of regulatory interactions in the gap gene system of *Drosophila melanogaster*. *Genetics*, 167:1721–1737, 2004.

[113] Hilde Janssens, Shuling Hou, Johannes Jaeger, Ah-Ram Kim, Ekaterina Myasnikova, David Sharp, and John Reinitz. Quantitative and predictive model of transcriptional control of the *Drosophila melanogaster even skipped* gene. *Nature Genetics*, 38:1159–1165, 2006.

[114] Iain M. Johnstone and Arthur Y. Lu. On consistency and sparsity for principal components analysis in high dimensions. *Journal of the American Statistical Association*, 104(486):682–693, 2009. To appear.

[115] N. El Karoui. Operator norm consistent estimation of large dimensional sparse covariance matrices. *Annals of Statistics*, 36(6):2717–2756, 2008. To appear.

[116] Kenneth J Kauffman, Purusharth Prakash, and Jeremy S Edwards. Advances in flux balance analysis. *Current Opinion in Biotechnology*, 14(5):491–496, 2003.

[117] H. Khalil. *Nonlinear Systems*. Prentice Hall, 2001.

[118] Thomas Klein and Marek Mlodzik. Planar cell polarization: An emerging model points in the right direction. *Annual Review of Cell and Developmental Biology*, 21:155–176, 2005.

[119] M. Klinger and D. Tautz. Formation of embryonic axes and blastoderm patterns in *Drosophila*. In V.E.A. Russo, D. J. Cove, L. G. Edgar, R. Jaenisch, and F. Salamini, editors, *Development: Genetics, epigenetics, and enviornmental regulation*. Springer-Verlag, 1999.

[120] K. Knight and W. Fu. Asymptotics for lasso-type estimators. *Annals of Statistics*, 28(5):1356–1378, 2000.

[121] Tetsuya Kobayashi, Luonan Chen, and Kazuyuki Aihara. Modeling genetic switches with positive feedback loops. *Journal of Theoretical Biology*, 221:379–399, 2003.

[122] Ja-Yong Koo. Optimal rates of convergence for nonparametric statistical inverse problems. *The Annals of Statistics*, 21(2):590–599, 1993.

[123] S. Kritchman and B. Nadler. Determining the number of components in a factor model from limited noisy data. *Chemometrics and Intelligent Laboratory Systems*, 94:19–32, 2008.

[124] P. R. Kumar and P. Varaiya. *Stochastic Systems: Estimation, Identification, and Adaptive Control*. Prentice Hall, 1986.

[125] H. Kunze and D. Siegel. A graph theoretical approach to monotonicity with respect to initial conditions. In X. Liu and D. Siegal, editors, *Comparison Methods and Stability Theory*. CRC, 1994.

[126] John Lafferty and Larry Wasserman. Rodeo: Sparse nonparametric regression in high dimensions. In *Advances in Neural Information Processing Systems (NIPS)*, 2005.

[127] Peter Lawrence. *The Making of the Fly*. Wiley-Blackwell, 1992.

[128] O. Ledoit and M. Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88:365–411, 2003.

[129] John Lee. *Introduction to Smooth Manifolds*. Springer, 2003.

[130] Thierry Léveillard and Bohdan Wasylyk. The MDM2 C-terminal region binds to $TAF_{ii}250$ and is required for MDM2 regulation of the Cyclin A promoter. *The Journal of Biological Chemistry*, 272(49):30651–30661, 1997.

[131] E. Levina and P. Bickel. Maximum likelihood estimation of intrinsic dimension. In *Advances in NIPS 17*, 2005.

[132] J. Lewis and M. Yannakakis. The node-deletion problem for hereditary properties is NP-complete. *Journal Of Computer and System Sciences*, 20(2):219–230, 1980.

[133] X.-Y. Li, S. MacArthur, R. Bourgon, D. Nix, D.A. Pollard, V.N. Iyer, A. Hechmer, L. Simirenko, M. Stapleton, H.-C. Luengo Hendriks, C.L. Chu, N. Ogawa, W. Inwood, V. Sementchenko, A. Beaton, R. Weiszmann, S.E. Celniker, D.W. Knowles, T. Gingeras, T.P. Speed, M.B. Eisen, , and M.D. Biggin. Transcription factors bind thousands of active and inactive regions in the *Drosophila* blastoderm. *PLoS Biology*, 6(2), 2008.

[134] J.D. Licht, M.J. Grossel, J. Figge, and U.M. Hansen. *Drosophila* Krüppel protein is a transcriptional repressor. *Nature*, 346:76–79, 1990.

[135] Lennart Ljung. *System Identification*. Prentice Hall, 1999.

[136] Robin Lougee-Heimer. The Common Optimization INterface for operations research. *IBM Journal of Research and Development*, 47(1):57–66, 2003.

[137] L. Lovász and A. Schrijver. Cones of matrices and set-functions and 0-1 optimization. *SIAM Journal of Optimization*, 1:166–190, 1991.

[138] Cris Luengo Hendriks, Soile Keränen, Charless Fowlkes, Lisa Simirenko, Gunther Weber, Angela DePace, Clara Henriquez, David Kaszuba, Bernd Hamann, Michael Eisen, Jitendra Malik, Damir Sudar, Mark Biggin, and David Knowles. Three-dimensional morphology and gene expression in the *Drosophila* blastoderm at cellular resolution I: data acquisition pipeline. *Genome Biology*, 7(12):R123, 2006.

[139] Gábor Lugosi. Concentration-of-measure inequalities. Technical report, Pompeu Fabra University, 2006.

[140] John Lygeros, Claire Tomlin, and Shankar Sastry. Controllers for reachability specifications for hybrid systems. *Automatica*, 35(3):349–370, March 1999.

[141] Antonio Marco, Charlotte Konikoff, Timothy Karr, and Sudhir Kumar. Relationship between gene co-expression and sharing of transcription factor binding sites in *Drosophila melanogaster*. *Bioinformatics*, 25(19):2473–2477, 2009.

[142] Florian Markowetz and Rainer Spang. Inferring cellular networks - a review. *BMC Bioinformatics*, 8(Suppl 6):S5, 2007.

[143] V. A. Marčenko and L. A. Pastur. Distributions of eigenvalues of some sets of random matrices. *Math. USSR-Sb.*, 1(4):507–536, 1967.

[144] W. F. Massy. Principal components regression in exploratory statistical research. *Journal of the American Statistical Association*, 60:234–246, 1965.

[145] G.C. McDonald and R.C. Schwing. Instabilities of regression estimates relating air pollution to mortality. *Technometrics*, 15:463–482, 1973.

[146] N. Meinshausen and B. Yu. Lasso-type recovery of sparse representations for high-dimensional data. *Annals of Statistics*, 2006. To appear.

[147] Nicolai Meinshausen and Peter Buehlmann. Stability selection. Working paper.

[148] C. W. Misner, K. S. Thorne, and J. A. Wheeler. *Gravitation*. W. H. Freeman, 1973.

[149] Ian Mitchell, A. Bayen, and Claire Tomlin. A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games. *IEEE Transactions on Automatic Control*, 50(7):947–957, 2005.

[150] Adilson Motter, Natali Gulbahce, Eivind Almaas, and Albert-László Barabási. Predicting synthetic rescues in metabolic networks. *Molecular Systems Biology*, 4, 2008.

[151] James Murray. *Mathematical Biology*. Springer, 2007.

[152] Boaz Nadler. Finite sample approximation results for principal component analysis: A matrix perturbation approach. *Annals of Statistics*, 2008. To appear.

[153] Andrew Ng. Preventing overfitting of cross-validation data. In *14th International Conference on Machine Learning*, pages 245–253, 1997.

[154] Partha Niyogi. Manifold regularization and semi-supervised learning: Some theoretical analyses. Technical Report TR-2008-01, Computer Science Dept., University of Chicago, 2008.

[155] G. Obozinski, B. Taskar, and M. Jordan. Multi-task feature selection. Technical report, Department of Statistics, University of California at Berkeley, 2006.

[156] J. Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2000.

[157] Theodore J Perkins, Johannes Jaeger, John Reinitz, and Leon Glass. Reverse engineering the gap gene network of *Drosophila* melanogaster. *PLoS Computational Biology*, 2(5):e51, 05 2006.

[158] W.F. Phillips, C.E. Hailey, and G.A. Gebert. Review of attitude representations used for aircraft kinematics. *Journal of Aircraft*, 38(4):718–737, 2001.

[159] E. Plahte, T. Mestl, and S. Omholt. Feedback loops, stability, and multistationarity in dynamical systems. *Journal of Biological Systems*, 3:409–413, 1995.

[160] R. Porreca, S. Drulhe, H. de Jong, and G. Ferrari-Trecate. Structural identification of piecewise-linear models of genetic regulatory networks. *Journal of Computational Biology*, 15(10):1365–1380, 2008.

[161] John G. Proakis and Dimitris K. Manolakis. *Digital Signal Processing: Principals, Algorithms, and Applications*. Prentice Hall, 1995.

[162] M. Ptashne, A. Jeffrey, A. Johnson, R. Maurer, B. Meyer, C. Pabo, T. Roberts, and T. Sauer. How the lambda repressor and cro work. *Cell*, 19:1–11, 1980.

[163] Karthik Raman, Preethi Rajagopalan, and Nagasuma Chandra. Flux balance analysis of mycolic acid pathway: Targets for anti-tubercular drugs. *PLoS Computational Biology*, 1(5):e46, Oct 2005.

[164] Christopher V. Rao and Adam P. Arkin. Control motifs for intracellular regulatory networks. *Annual Review of Biomedical Engineering*, 3(1):391–419, 2001.

[165] R. Rao, G. Fung, and R. Rosales. On the dangers of cross-valdation: an experimental evaluation. In *SIAM Data Mining*, 2008.

[166] Juha Reunanen. Overfitting in making comparisons between variable selection methods. *Journal of Machine Learning Research*, 3:1371–1382, 2003.

[167] Stefan Ries, Carola Biederer, Douglas Woods, Ohad Shifman, Senji Shirasawa, Takehiko Sasazuki, Martin McMahon, Moshe Oren, , and Frank McCormick. Opposing effects of Ras on p53: Transcriptional activation of MDM2 and induction of p19 ARF. *Cell*, 103:321–330, 2000.

[168] D-K. Ro, E. M. Paradise, M. Ouellet, K. J. Fisher, K. L. Newman, J. M. Ndungu, K. A. Ho, R. A. Eachus, T. S. Ham, J. Kirby, M. C. Y. Chang, S. T. Withers, Y. Shiba, R. Sarpong, and J. D. Keasling. Production of the antimalarial drug precursor artemisinic acid in engineered yeast. *Nature*, 440:940–943, 2006.

[169] D. Ruppert and M. Wand. Multivariate locally weighted least squares regression. *Annals of Statistics*, 22(3):1346–1370, 1994.

[170] S. Sastry. *Nonlinear Systems*. Springer, 1999.

[171] F. Sauer and H. Jäckle. Concentration-dependent transcriptional activation or repression by Krüppel from a single binding site. *Nature*, 353:563–566, 1991.

[172] Michael A. Savageau. Michaelis-Menten mechanism reconsidered: Implications of fractal kinetics. *Journal of Theoretical Biology*, 176(1):115–124, 1995.

[173] Michael A. Savageau. Development of fractal kinetic theory for enzyme-catalysed reactions and implications for the design of biochemical pathways. *Biosystems*, 47(1-2):9–36, 1998.

[174] A. Savitzky and Marcel J.E. Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, 36:1627–1639, 1964.

[175] H. Schneeweiß. Consistent estimation of a regression with errros in variables. *Metrika*, 23(1):101–115, 1976.

[176] Eran Segal, Tali Raveh-Sadka, Mark Schroeder, Ulrich Unnerstall, and Ulrike Gaul. Predicting expression patterns from regulatory sequence in *Drosophila* segmentation. *Nature*, 451:535–540, 2008.

[177] Jun Shao. Linear model selection by cross-validation. *Journal of the American Statistical Association*, 88(422):486–494, 1993.

[178] Jun Shao. Bootstrap sample size in nonregular cases. *Proceedings of the American Mathematical Society*, 122(4):1251–1262, 1994.

[179] Jun Shao. Bootstrap model selection. *Journal of the American Statistical Association*, 91(434):655–665, 1996.

[180] Yuko Shimada, Shigenobu Yonemura, Hiroyuki Ohkura, David Strutt, and Tadashi Uemura. Polarized transport of Frizzled allong the planar microtubule arrays in *Drosophila* wing epithelium. *Developmental Cell*, 10(2):209–222, 2006.

[181] I. Shmulevich, E. Dougherty, and W. Zhang. Control of stationary behavior in probabilistic boolean networks by means of structural intervention. *Biological Systems*, 10(4):431–446, 2002.

[182] S. Small, A. Blair, and M. Levine. Regulation of two pair-rule stripes by a single enhancer in the *Drosophila* embryo. *Developmental Biology*, 175(2):314–324, 1996.

[183] Hal L. Smith. Systems of ordinary differential equations which generate an order preserving flow. A survey of results. *SIAM Review*, 30(1):87–113, 1988.

[184] Hal L. Smith. *Monotone dynamical systems: an introduction to the theory of competitive and cooperative systems*. American Mathematical Society, 1995.

[185] E. H. Snoussi. Qualitative dynamics of piecewise-linear differential equations: A discrete mapping approach. *Dynam. Stabil. Syst.*, 4(3-4):189–207, 1989.

[186] Eduardo Sontag, Alan Veliz-Cuba, Reinhard Laubenbacher, and Abdul Salam Jarrah. The effect of negative feedback loops on the dynamics of boolean networks. *Biophysical Journal*, 95:518–526, 2008.

[187] Eduardo D. Sontag. Monotone and near-monotone biochemical networks. *Systems and Synthetic Biology*, 1(2):59–87, April 2007.

[188] C. Soulé. Graphic requirements for multistationarity. *Complexus*, 1:123–133, 2003.

[189] Michael Spivak. *Calculus on Manifolds*. Addison-Wesley Publishing Company, 1965.

[190] M. K. Stephen Yeung, Jesper Tegnér, and James Collins. Reverse engineering gene networks using singular value decomposition and robust regression. *PNAS*, 99(9):6163–6168, 2002.

[191] R. Steuer, J. Kurths, C. Daub, J. Weise, and J. Selbig. The mutual information: Detecting and evaluating dependencies between variables. *Bioinformatics*, 18(Suppl. 2):S231–S240, 2002.

[192] G. Stewart and J. Sun. *Matrix Perturbation Theory*. Academic Press, Inc., 1990.

[193] David Strutt. Asymmetric localization of Frizzled and the establishment of cell polarity in the *Drosophila* wing. *Molecular Cell*, 7(2):367–375, 2001.

[194] J. Stuart, E. Segal, D. Koller, and S. Kim. A gene-coexpression network for global discovery of conserved genetic modules. *Science*, 302(5643):249–255, 2003.

[195] Dengfeng Sun and Alexandre Bayen. Multicommodity Eulerian-Lagrangian large-capacity cell transmission model for en route traffic. *Journal of Guidance, Control, and Dynamics*, 31(3):616–628, 2008.

[196] Masatoshi Takagi, Michael J. Absalon, Kevin G. McLure, and Michael B. Kastan. Regulation of p53 translation and induction after DNA damage by ribosomal protein L26 and Nucleolin. *Cell*, 123:49–63, 2005.

[197] William Thieman and Michael Palladino. *Introduction to Biotechnology*. Benjamin Cummings, 2004.

[198] Zhaolu Tian and Chuanqing Gu. A numerical algorithm for Lyapunov equations. *Applied Mathematics and Computation*, 202:44–53, 2008.

[199] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B*, 58(1):267–288, 1996.

[200] Jo-Anne Ting, Aaron D'Souza, Sethu Vijayakumar, and Stefan Schaal. A Bayesian approach to empirical local linearization for robotics. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2008.

[201] Claire Tomlin, John Lygeros, and Shankar Sastry. A game theoretic approach to controller design for hybrid systems. *Proceedings of the IEEE*, 88(7):949–969, July 2000.

[202] A. Torralba, K. Murphy, and W. Freeman. Sharing features: Efficient boosting procedures for multiclass object detection. In *IEEE Conference on Computer Vision and Pattern recognition*, volume 2, pages 762–769, 2004.

[203] David Tree, Joshua Shulman, Raphaeël Rousset, Matthew Scott, David Gubb, and Jeffrey Axelrod. Prickle mediates feedback amplification to generate assymetric planar cell polarity signalling. *Cell*, 109(3):371–381, 2002.

[204] Alan Turing. The chemical basis of morphogenesis. *Philosophical Transactions of the Royal Society B*, 237:37–72, 1952.

[205] Tadao Usui, Yasuyuki Shima, Yuko Shimada, Shinji Hirano, Robert Burgess, Thomas Schwarz, Masatoshi Takeichi, and Tadashi Uemura. Flamingo, a seven-pass transmembrane cadherin, regulates planar cell polarity under the control of Frizzled. *Cell*, 98(5):585–595, 1999.

[206] A. Van Der Shaft. L2 gain analysis of nonlinear systems and nonlinear state feedback h-infinity control. *IEEE Transactions on Automatic Control*, 37(6):770–784, 1992.

[207] Sabine Van Huffel and Joos Vandewalle. *The total least squares problem: computational aspects and analysis*. SIAM, 1991.

[208] Lyubomir T. Vassilev, Binh T. Vu, Bradford Graves, Daisy Carvajal, Frank Podlaski, Zoran Filipovic, Norman Kong, Ursula Kammlott, Christine Lukacs, Christian Klein, Nader Fotouhi, and Emily A. Liu. In vivo activation of the p53 pathway by small-molecule antagonists of MDM2. *Science*, 303(5659):844–848, 2004.

[209] Sethu Vijayakumar, Aaron D'Souza, and Stefan Schaal. Incremental online learning in high dimensions. *Neural Computation*, 17:2602–2634, 2005.

[210] George von Dassow, Eli Meir, Edwin Munro, and Garrett Odell. The segment polarity network is a robust developmental module. *Nature*, 406:188–192, 2000.

[211] Hansheng Wang and Chenlei Leng. A note on adaptive group lasso. *Computational Statistics and Data Analysis*, 52:5277–5286, 2008.

[212] Adriano Werhli, Marco Grzegorczyk, and Dirk Husmeier. Comparative evaluation of reverse engineering gene regulatory networks with relevance networks, graphical Gaussian models and Bayesian networks. *Bioinformatics*, 22(20):2523–2531, 2006.

[213] H. Wold. Soft modeling by latent variables: the nonlinear iterative partial least squares approach. In J. Gani, editor, *Perspectives in Probability and Statistics, Papers in Honour of M. S. Bartlett*. Academic Press, 1975.

[214] Tong Tong Wu and Kenneth Lange. Coordinate descent algorithms for lasso penalized regression. *Annals of Applied Statistics*, 2(1):224–244, 2008.

[215] Zuoshuang Xiang, Rebecca Minter, Xiaoming Bi, Peter Woolf, and Yongqun Hechmer. miniTUBA: medical inference by network integration of temporal data using Bayesian analysis. *Bioinformatics*, 23(18):2423–2432, 2007.

[216] Feng Xu and Hanshu Ding. A new kinetic model for heterogeneous (or spatially confined) enzymatic catalysis: Contributions from the fractal and jamming (overcrowding) effects. *Applied Catalysis A: General*, 317(1):70–81, 2007.

[217] Kun Yang, Hongjun Bai, Qi Ouyang, Luhua Lai, and Chao Tang. Finding multiple target optimal intervention in disease-related molecular network. *Molecular Systems Biology*, 4(228), 2008.

[218] Yang Yang, Wenjiang Fu, and Kenneth Land. A methodological comparison of age-period-cohort models: the intrinsic estimator and conventional generalized linear models. *Sociological Methodology*, 34:75–110, 2004.

[219] J. Yu, V. Smith, P. Wang, A. Hartemink, and E. Jarvis. Advances to bayesian network inference for generating causal networks from observational biological data. *Bioinformatics*, 20(18):3594–3603, 2004.

[220] Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society Series B*, 68(1):49–67, 2006.

[221] J. Zhang, Z. Ghahramani, and Y. Yang. Learning multiple related tasks using latent independent component analysis. In *Advances in neural information processing systems*, volume 18, pages 1585–1592, 2006.

[222] H. Zou. The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006.

[223] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B*, 67(2):301–320, 2005.

[224] H. Zou, T. Hastie, and R. Tibshirani. Sparse principal component analysis. *Journal of Computational & Graphical Statistics*, 15(2):265–286, 2006.

Appendix A

Proofs for Theorems on Exterior Derivative Estimation

In this appendix chapter, proofs are given for the theorems on exterior derivative estimation. A few lemmas not stated in the text are also given; they are needed for the proofs.

Lemma A.0.1. If the assumptions in Section 5.1 hold, then

a) $\left\| \mathbb{E}(\hat{C}_n{}^{22}) - C^{22} \right\|_2^2 = O(h^4)$;

b) $\left\| \mathbb{E}\left[(\hat{C}_n{}^{22} - C^{22})(\hat{C}_n{}^{22} - C^{22})' \right] \right\|_2^2 = O(1/nh^d)$;

c) $\hat{C}_n \xrightarrow{p} C$.

Proof. This proof follows the techniques of [169, 33]. We first prove part (c). Note that

$$\hat{C}_n{}^{11} = \frac{1}{nh^{d-p}} \sum_{i=1}^{n} K_h(x_i - x_0),$$

and consider its expectation

$$\begin{aligned}
\mathbb{E}\left(\hat{C}_n{}^{11}\right) &= \mathbb{E}\left(\frac{1}{h^{d-p}} K_h(x_i - x_0) \mathbb{1}(X \in (\mathcal{B}^p_{x,h^{1-\epsilon}})) \right) + \\
&\qquad \mathbb{E}\left(\frac{h^p}{h^d} K_h(x_i - x_0) \mathbb{1}(X \in (\mathcal{B}^p_{x,h^{1-\epsilon}})^c) \right) \\
&= \int_{B^d_{0,h^{1-\epsilon}}} \frac{1}{h^d} K\left(\frac{\phi(z) - \phi(0)}{h} \right) F(z) dz + o(h^{2+p}) \\
&= \int_{\mathbb{R}^n} K(d_u \phi \cdot u) F(0) du + O(h^2).
\end{aligned}$$

where we have used the assumption that $K(\cdot)$ is an even function, $K'(\cdot)$ is an odd function, and $K''(\cdot)$ is an even function.

A similar calculation shows that for

$$\hat{C}_n{}^{21} = \frac{1}{nh^{d+1-p}} \sum_{i=1}^{n} K_h(x_i - x_0)(x_i - x_0),$$

we have that the expectation is

$$\mathbb{E}\left(\hat{C}_n{}^{21}\right) = O(h) = o(1).$$

And, a similar calculation shows that for

$$\hat{C}_n{}^{22} = \frac{1}{nh^{d+2-p}} \sum_{i=1}^{n} K_h(x_i - x_0)(x_i - x_0)(x_i - x_0)',$$

we have that the expectation is

$$\mathbb{E}(\hat{C}_n{}^{22}) = F(0) d_u\phi \cdot \left[\int_{\mathbb{R}^d} K(d_u\phi(0) \cdot u) uu' du\right] \cdot d_u\phi' + O(h^2).$$

The result in part (c) follows from the weak law of large numbers. The last calculation also proves part (a).

Next, we prove part (b). For notational simplicity, let

$$T_i = K_h(x_i - x_0)(x_i - x_0)(x_i - x_0)'.$$

The variance is

$$\begin{aligned}\mathrm{Var}(\hat{C}_n{}^{22}) = \frac{1}{n^2 h^{4+2d-2p}} \mathrm{Tr}\Big(& n(\mathbb{E}(T_i T_i') - \mathbb{E}(T_i)\mathbb{E}(T_i)') + \\ & n(n-1)(\mathbb{E}(T_i T_j') - \mathbb{E}(T_i)\mathbb{E}(T_j)')\Big).\end{aligned}$$

Since T_i and T_j are independent, it follows that $\mathbb{E}(T_i T_j') - \mathbb{E}(T_i)\mathbb{E}(T_j)' = 0$. Next, note that

$$\begin{aligned}\mathbb{E}(T_i T_i') &= h^{-2p}\Bigg(\int_{B^d_{0,h^{1-\epsilon}}} \left(K\left(\frac{\phi(z) - \phi(0)}{h}\right)\right)^2 \Big[\phi(z) - \phi(0)\Big] \\ &\qquad \times \Big[\phi(z) - \phi(0)\Big]' \Big[\phi(z) - \phi(0)\Big] \Big[\phi(z) - \phi(0)\Big]' F(z) dz \\ &\qquad + o(h^{d+2})\Bigg) \\ &= h^{d+4-2p}\Big(F(0) \int_{\mathbb{R}^d} (K(d_u\phi \cdot u))^2 d_u\phi \cdot uu' \cdot d_u\phi' \cdot \\ &\qquad d_u\phi \cdot uu' \cdot d_u\phi' \cdot du + O(h^2)\Big).\end{aligned}$$

Thus, the variance is given by

$$\begin{aligned}\mathrm{Var}(\hat{C}_n{}^{22}) = \frac{1}{nh^d} \mathrm{Tr}\Big(& F(0) \int_{\mathbb{R}^d} (K(d_u\phi \cdot u))^2 d_u\phi \cdot uu' \cdot d_u\phi' \cdot \\ & d_u\phi \cdot uu' \cdot d_u\phi' \cdot du + o_p(1)\Big).\end{aligned}$$

□

Lemma A.0.2. If the assumptions in Section 5.1 hold, then the matrices $\hat{C}_n$, C^{22}, Π, $\hat{\Pi}_n$, $\hat{P}_n$, and P have the following properties:

a) $\mathrm{rank}(C^{22}) = d$ and $\mathcal{R}(C^{22}) = T_p\mathcal{M}$;

b) $\mathcal{R}(\Pi) = \mathcal{N}(C^{22})$, $\mathcal{N}(\Pi) = \mathcal{R}(C^{22})$, and $\mathcal{N}(\Pi) \cap \mathcal{N}(C^{22}) = \{0\}$;

c) $\|\hat{P}_n - P\|_2{}^2 = \|\hat{\Pi}_n - \Pi\|_2{}^2 = O_p(1/nh^d)$;

d) $\mathbb{P}(\text{rank}(\hat{C}_n + \lambda_n \hat{P}_n / nh^{d+2}) = p + 1) \to 1$.

Proof. To show property (a), we first show that for $M \in \mathbb{R}^{d \times d}$, where

$$M = \int_{\mathbb{R}^d} K(d_u\phi(0) \cdot u) uu' du,$$

we have that $\text{rank}(M) = d$. To prove this, choose any $v \in \mathbb{R}^d \setminus \{0\}$ and then consider the quantity

$$v'Mv = \int_{\mathbb{R}^d} K(d_u\phi(0) \cdot u) v'uu'v du.$$

By construction, $v'uu'v > 0$ almost everywhere. Additionally, since ϕ is three times differentiable, we have that $K(d_u\phi(0) \cdot u) > 0$ on a set of non-zero measure and $K(d_u\phi(0) \cdot u) \geq 0$ elsewhere. Thus, $v'Mv > 0$ for all $v \in \mathbb{R}^d \setminus \{0\}$. It follows that M is symmetric and positive definite with $\text{rank}(M) = d$. Since $\mathcal{M}$ is a d-dimensional manifold, we have that $\text{rank}(d_u\phi) = d$ by Corollary 8.4 of [129]. The Sylvester Inequality [170] implies that

$$\text{rank}(C_{22}) = \text{rank}(d_u\phi M d_u\phi') = d,$$

and this implies that

$$\mathcal{R}(C_{22}) = \mathcal{R}(d_u\phi M d_u\phi') = \mathcal{R}(d_u\phi).$$

However, $\mathcal{R}(d_u\phi) = T_p\mathcal{M}$, where we take $p = x_0$. This proves the result.

We next consider property (b). We have that

$$\sigma_1, \ldots, \sigma_d \neq 0 \text{ and } \sigma_{d+1} = \ldots = \sigma_p = 0,$$

because $\text{rank}(C_{22}) = d$ by property (a). Thus, the null-space of C_{22} is given by the column-span of U_N; however, the construction of P implies that the column-span of U_N is the range-space of P. Ergo, $\mathcal{R}(P) = \mathcal{N}(C_{22})$. Note that the column-span of U_R belongs to the null-space of P, because each column in U_R is orthogonal—by property of the SVD—to each column in U_N. Thus, we have the dual result that $\mathcal{N}(P) = \mathcal{R}(C_{22})$. The orthogonality of U_R and U_N due to the SVD implies that $\mathcal{N}(P) \cap \mathcal{N}(C_{22}) = \{0\}$.

Now, we turn to property (c). For $h = \kappa n^{-1/(d+4)}$, Lemma A.0.1 says that $\|\hat{C}_n{}^{22} - C_{22}\|_F{}^2 = O_p(1/nh^d)$. The result follows from Corollary 3 of [115], by the fact that $\mathbb{I} - P_X$ is the projection matrix onto the null-space of X, and by the equivalence:

$$\|P_X - P_Z\|_2{}^2 \equiv \|\sin\Theta[\mathcal{R}(X), \mathcal{R}(Z)]\|,$$

where P_X is a projection matrix onto the range space of X [192].

Lastly, we deal with property (d). Lemma A.0.1 shows that

$$C_n \xrightarrow{p} C = \begin{bmatrix} C_{11} & 0 \\ 0 & C_{22} \end{bmatrix}.$$

Since $F(0) \neq 0$ by assumption, $C_{11} \neq 0$; thus, $\text{rank}(C) = 1+\text{rank}(C_{22})$. Since $\mathcal{N}(P) \cap \mathcal{N}(C_{22}) = \{0\}$, we have that $\text{rank}(C_{22} + \lambda_n P/nh^{d+2}) = p$. Consequently, $\text{rank}(C) = p + 1$. Next, consider the expression

$$\begin{aligned}
&\left\| C_n + \lambda_n \tilde{P}_n/nh^{d+2} - C - \lambda_n \tilde{P}/nh^{d+2} \right\|_2^2 \\
&\qquad \leq \left\| \hat{C}_n - C \right\|_2^2 + \frac{\lambda_n}{nh^{d+2}} \left\| \tilde{P}_n - \tilde{P} \right\|_2^2 \\
&\qquad \leq O_p(h^2) + O_p\left(\frac{\lambda_n}{n^2 h^{2d+2}} \right) \\
&\qquad \leq o_p(h)
\end{aligned}$$

Weyl's theorem [30] implies that

$$\left\| \sigma_i(C_n + \lambda_n \tilde{P}_n/nh^{d+2}) - \sigma_i(C + \lambda_n \tilde{P}/nh^{d+2}) \right\|_2^2 \leq o_p(h). \tag{A.1}$$

Note that $\sigma_i(C + \lambda_n \tilde{P}/nh^{d+2})$ is non-decreasing because λ_n/nh^{d+2} is non-decreasing. Define

$$\eta = \min\left(\sigma_i(C + \lambda_n \tilde{P}/n^{2/(d+4)}) \right),$$

and consider the probability

$$\begin{aligned}
&\mathbb{P}(\text{rank}(C_n + \lambda_n \tilde{P}_n/nh^{d+2}) = p + 1) \\
&\geq \mathbb{P}(|\sigma_i(C_n + \lambda_n \tilde{P}_n/nh^{d+2}) + \\
&\qquad\qquad - \sigma_i(C + \lambda_n \tilde{P}/nh^{d+2})| \leq \eta, \forall i) \\
&\geq \sum_{i=1}^{p+1} \mathbb{P}(|\sigma_i(C_n + \lambda_n \tilde{P}_n/nh^{d+2}) + \\
&\qquad\qquad - \sigma_i(C + \lambda_n \tilde{P}/nh^{d+2})| \leq \eta) - p
\end{aligned} \tag{A.2}$$

The result follows from equations (A.1) and (A.2). □

For notational convenience, we define

$$B_n = h^p (f(X) - \beta X_x)' W_{x_0} X_x H^{-1/2},$$

and

$$M = \frac{1}{2}\left[\partial_i \partial_j (f \circ \phi) - d_x f \cdot \partial_i \partial_j \phi \right].$$

Then, we have the following result concerning the asymptotic bias of the estimator:

Lemma A.0.3. If $h = \kappa n^{-1/(d+4)}$, then $B_n \xrightarrow{p} B$, where

$$B = \left[\kappa F(0) \int_{\mathbb{R}^d} K(d_u \phi \cdot u) u u' du M' \quad \mathbb{O}_{p \times 1} \right]. \tag{A.3}$$

Proof. First, recall the Taylor polynomial of the pullback of f to z:

$$f(\phi(z)) = f(\phi(0)) + d_x f \cdot d_u \phi \cdot z + \frac{1}{2} \partial_i \partial_j (f \circ \phi) \cdot z z' + o(\|z\|^2),$$

where we have performed a pullback of $d_x f$ from $T_x^*\mathcal{M}$ to $T_x^*\mathcal{U}$. In the following expression, we set $z = hu$:

$$\begin{aligned} & f(\phi(z)) - f(x_0) - d_x f \cdot \Big[\phi(z) - \phi(0)\Big] \\ &\quad = \frac{h^2}{2}\Big[\partial_i\partial_j(f\circ\phi) - d_x f \cdot \partial_i\partial_j\phi\Big]uu' + o(\|hu\|^2) \\ &\quad = h^2 Buu' + o(\|hu\|^2). \end{aligned}$$

Because $\beta = \begin{bmatrix} f(x_0) & d_x f \end{bmatrix}'$, we can rewrite the expectation of the expression as

$$\begin{aligned} \mathbb{E}(B_n) &= \mathbb{E}\left(h^{p-d}K_H(x-x_0)(f(x) - x_{x_0}\beta)' x_{x_0}\begin{bmatrix}1 & 0\\ 0 & 1/h^2\mathbb{I}\end{bmatrix}\right) H^{1/2} \\ &= \Big(\int_{\mathbb{R}^d}\Big\{K(d_u\phi\cdot u)h^2 u'uM'\begin{bmatrix}1 & \frac{1}{h}d_u\phi\cdot u + \frac{1}{2}\partial_i\partial_j\phi\cdot uu'\end{bmatrix}\times \\ &\qquad (F(0) + hd_uF(0)\cdot u)\Big\}du + o(h^2)\Big)H^{1/2} \\ &= \sqrt{nh^d}h^2\begin{bmatrix}F(0)\int_{\mathbb{R}^d}K(d_u\phi\cdot u)uu'duM' + o(1) & O(\sqrt{h^2})\end{bmatrix}, \end{aligned}$$

where the last line follows because of the odd symmetries in the integrand. Since $h = \kappa n^{-1/(d+4)}$, this expectation becomes

$$\mathbb{E}(B_n) = B + o(1)\mathbb{1}_{1\times(p+1)}.$$

The result follows from application of the weak law of large numbers. □

Let

$$V = F(0)\int_{\mathbb{R}^d}(K(d_u\phi\cdot u))^2\begin{bmatrix}1 & 0\\ 0 & d_u\phi\cdot uu'\cdot d_u\phi\end{bmatrix}du, \tag{A.4}$$

then the following lemma describes the asymptotic distribution of the error residuals.

Lemma A.0.4. If $h = \kappa n^{-1/(d+4)}$, then

$$h^p\epsilon' W_{x_0}X_x H^{-1/2} \xrightarrow{d} \mathcal{N}\left(0, \sigma^2 V\right).$$

Proof. Since $\mathbb{E}(\epsilon) = 0$ and ϵ is independent of x, we have that

$$\mathbb{E}\left(\sqrt{n}h^p\epsilon K_H(x-x_0)x_{x_0}H^{-1/2}\right) = 0.$$

The variance of this quantity is

$$\begin{aligned} & \mathrm{Var}\left(h^p\sqrt{n}\epsilon K_H(x-x_0)x_{x_0}H^{-1/2}\right) \\ &= \mathbb{E}\left(\left(h^p\sqrt{n}\epsilon K_H(x-x_0)x_{x_0}H^{-1/2}\right)'\left(h^p\sqrt{n}\epsilon K_H(x-x_0)x_{x_0}H^{-1/2}\right)\right) \\ &= nh^{2p}\sigma^2\mathbb{E}\left((K_H(x-x_0))^2\begin{bmatrix}1 & (x-x_0)'\end{bmatrix}'H^{-1}\begin{bmatrix}1 & (x-x_0)'\end{bmatrix}\right) \\ &= \sigma^2\Bigg\{\int_{\mathbb{R}^d}(K(d_u\phi\cdot u))^2\begin{bmatrix}1 & (d_u\phi\cdot u + \frac{h}{2}\partial_i\partial_j\phi\cdot uu')' \\ . & d_u\phi\cdot uu'\cdot d_u\phi\end{bmatrix}(F(0) \\ &\qquad + hd_uF(0)\cdot u)du + o(h^2)\Bigg\} \\ &= \sigma^2(V + o(h)\mathbb{I}). \end{aligned}$$

Thus, the central limit theorem implies that

$$\frac{h^p\sqrt{n}\epsilon' W_{x_0}X_x H^{-1/2}}{\sqrt{n}} \xrightarrow{d} \mathcal{N}(0,\sigma^2 V) \qquad \square$$

Proof of Theorem 5.3.2. This proof follows the framework of [120, 222] but with significant modifications to deal with our estimator. For notational convenience, we define the indices of β such that: $\beta_0 = f(x_0)$ and $\begin{bmatrix}\beta_1 & \dots & \beta_p\end{bmatrix} = d_x f$. Let $\tilde{\beta} = \beta + H^{-1/2}u$ and

$$\Psi_n(u) = h^p \left\| {W_{x_0}}^{1/2}(Y - X_{x_0}(\beta + H^{-1/2}u)) \right\|_2^2 + \lambda_n \left\| P_n \cdot (\beta + H^{-1/2}u) \right\|_2^2.$$

Let $\hat{u}^{(n)} = \arg\min \Psi_n(u)$; then $\hat{\beta}^{(n)} = \beta + H^{-1/2}\hat{u}^{(n)}$. Note that $\Psi_n(u) - \Psi_n(0) = {V_4}^{(n)}(u)$, where

$$\begin{aligned}{V_4}^{(n)}(u) = u' H^{-1/2}(h^p X'_{x_0} W_{x_0} X_{x_0} + \lambda_n P_n) H^{-1/2} u \\ + 2(h^p (Y - X_{x_0}\beta)' W_{x_0} X_{x_0} + \lambda_n \beta' P_n) H^{-1/2} u.\end{aligned}$$

If $\lambda_n / n h^{d+2} \to \infty$ and $h\lambda_n / n h^{d+2} \to 0$, then for every u

$$\lambda_n \beta' P_n H^{-1/2} u = \lambda_n \beta' P u / \sqrt{n h^{d+2}} O_p(1) + \lambda_n h / n h^{d+2} O_p(1),$$

where we have used Lemma A.0.2. It follows from the definition of β (5.10) and Lemma A.0.2 that $\beta' P \equiv 0$; thus, $\lambda_n \beta' P_n H^{-1/2} u = \lambda_n h / n h^{d+2} O_p(1) = o_p(1)$. For all $u \in \mathcal{N}(P)$, we have

$$\lambda_n / n h^{d+2} u' P_n u = \lambda_n / n h^{d+2} O_p(1/nh^d) = o_P(h\lambda_n / n h^{d+2}),$$

and for all $u \notin \mathcal{N}(P)$, we have

$$\lambda_n / n h^{d+2} u' P_n u = \lambda_n / n h^{d+2} u' P_n u O_p(1) \to \infty.$$

Let $W \sim \mathcal{N}(0, \sigma^2 V)$. Then, by Slutsky's theorem we must have that ${V_4}^{(n)}(u) \xrightarrow{d} V_4(u)$ for every u, where

$$V_4(u) = \begin{cases} u'Cu - 2u'(W + B), & \text{if } u \in \mathcal{N}(P) \\ \infty, & \text{otherwise} \end{cases}$$

Lemma 5 shows that ${V_4}^{(n)}(u)$ is convex with high-probability, and Lemma 5 also shows that $V_4(u)$ is convex. Consequently, the unique minimum of $V_4(u)$ is given by $u = C^\dagger(W + B)$, where $C^\dagger$ denotes the Moore-Penrose pseudoinverse of C. Following the epi-convergence results of [82, 120], we have that $\hat{u}^{(n)} \xrightarrow{d} C^\dagger(W + B)$. This proves asymptotic normality of the estimator, as well as convergence in probability.

The proof for the NALEDE estimator comes for free. The proof formulation that we have used for the consistency of nonparametric regression in (5.11) allows us to trivially extend the proof of [222] to prove asymptotic normality and consistency. $\square$

Lemma A.0.5. Consider $A_n, B_n \in \mathbb{R}^{p_n \times p_n}$ that are symmetric, invertible matrices. If $\|A_n - B_n\|_2 = O_p(\gamma_n)$, $\|A_n^{-1}\|_2 = O_p(1)$, and $\|B_n^{-1}\|_2 = O_p(1)$, then $\|A_n^{-1} - B_n^{-1}\|_2 = O_p(\gamma_n)$.

Proof. Consider the expression

$$\begin{aligned}\|A_n^{-1} - B_n^{-1}\|_2 &= \|A_n^{-1}(B_n - A_n)B_n^{-1}\|_2 \\ &\leq \|A_n^{-1}\|_2 \cdot \|A_n - B_n\|_2 \cdot \|B_n^{-1}\|_2,\end{aligned}$$

where the last line follows because the induced, matrix norm $\|\cdot\|_2$ is sub-multiplicative for square matrices. □

Proof of Theorem 5.4.3. Under our set of assumptions, the results from [32] apply:

$$\left\|T_t(X'X/n) - (\Sigma_\xi + \sigma_\nu^2\mathbb{I})\right\|_2 = O_p\left(c_n\sqrt{\frac{\log p}{n}}\right) \tag{A.5}$$

$$\left\|T_t(X'Y/n) - \Sigma_\xi\overline{\beta}\right\|_2 = O_p\left(c_n\sqrt{\frac{\log p}{n}}\right). \tag{A.6}$$

An argument similar to that given in Lemma A.0.2 implies that

$$\left\|\hat{P}_n - P_n\right\| = O_p\left(c_n\sqrt{\log p/n}\right).$$

Consequently, it holds that

$$\left\|T_t(X'X/n) - \sigma_\nu^2\mathbb{I} + \lambda_n\hat{P}_n - (\Sigma_\xi + \lambda_n P_n)\right\|_2 = O_p\left(c_n(\lambda_n+1)\sqrt{\frac{\log p}{n}}\right). \tag{A.7}$$

Next, observe that

$$\Sigma_\xi + \lambda_n P_n = \begin{bmatrix}U_R & U_N\end{bmatrix}\operatorname{diag}(\sigma_1,\ldots,\sigma_d,\lambda_n,\ldots,\lambda_n)\begin{bmatrix}U_R & U_N\end{bmatrix}'.$$

Recall that we only consider the case in which $d < p$. We have that

a) $\left\|(\Sigma_\xi + \lambda_n P_n)^{-1}\right\|_2 = O(1)$, because of (5.14);

b) $\left\|\Sigma_\xi^\dagger - (\Sigma_\xi + \lambda_n P_n)^{-1}\right\|_2 = O_p(1/\lambda_n)$.

Weyl's theorem [30] and (A.7) imply that

$$\left\|(T_t(X'X/n) - \sigma_\nu^2\mathbb{I} + \lambda_n\hat{P}_n)^{-1}\right\|_2 = O_p(1).$$

Additionally, Lemma A.0.5 implies that

$$\left\|(T_t(X'X/n) - \sigma_\nu^2\mathbb{I} + \lambda_n\hat{P}_n)^{-1} - (\Sigma_\xi + \lambda_n P_n)^{-1}\right\| = O_p\left(c_n\lambda_n\sqrt{\frac{\log p}{n}}\right).$$

Note that the solution to the estimator defined in (5.18) is:

$$\hat{\beta} = (T_t(X'X/n) - \sigma_\nu^2\mathbb{I} + \lambda_n\hat{P}_n)^{-1}T_t(X'Y/n).$$

Next, we define

$$\beta^{(n)} \triangleq (T_t(X'X/n) - \sigma_\nu^2\mathbb{I} + \lambda_n\hat{P}_n)^{-1}\Sigma_\xi\overline{\beta},$$

and note that the projection matrix onto the range space of Σ_ξ is given by $P_{\Sigma_\xi} = \Sigma_\xi^\dagger \Sigma_\xi$. Thus, $\beta = P_{\Sigma_\xi}\overline{\beta} = \Sigma_\xi^\dagger \Sigma_\xi \overline{\beta}$. Consequently, we have that

$$\begin{aligned} &\left\|\hat{\beta} - \beta\right\|_2 \\ &\quad \leq \left\|\hat{\beta} - \beta^{(n)}\right\|_2 + \left\|\beta^{(n)} - \beta\right\|_2 \\ &\quad \leq \left\|(T_t(X'X/n) - \sigma_\nu^2 \mathbb{I} + \lambda_n \hat{P}_n)^{-1}\right\|_2 \cdot \left\|T_t(X'Y/n) \right. \\ &\qquad \left. - \Sigma_\xi \overline{\beta}\right\|_2 + \left\|(T_t(X'X/n) - \sigma_\nu^2 \mathbb{I} + \lambda_n \hat{P}_n)^{-1} \right. \\ &\qquad \left. - \Sigma_\xi^\dagger\right\|_2 \cdot \left\|\Sigma_\xi \overline{\beta}\right\|_2, \end{aligned} \tag{A.8}$$

where the inequality comes about because $\|\cdot\|_2$ is an induced, matrix norm and the expressions are of the form $\mathbb{R}^{p\times p}(\mathbb{R}^{p\times p}\mathbb{R}^p)$. Recall that for symmetric matrices, $\|A\|_1 = \|A\|_\infty$; ergo, $\|A\|_2 \leq \sqrt{\|A\|_1 \|A\|_\infty} = \|A\|_1$. Because of (5.13), we can use this relationship on the norms to calculate that $\|\Sigma_\xi\|_2 = O(c_n)$ and $\|\overline{\beta}\| = O(c_n)$. Consequently,

$$(A.8) \leq O_p\left(c_n \lambda_n \sqrt{\frac{\log p}{n}}\right) + O_p(c_n^2/\lambda_n).$$

The result follows from the relationship

$$\lambda_n = O\left(\sqrt{c_n}\left(\frac{n}{\log p}\right)^{1/4}\right). \qquad \square$$

We can show that the bias of the terms of the nonparametric exterior derivative estimation goes to zero at a certain rate.

Lemma A.0.6. Under the assumptions of Section 5.4, we have that

$$\begin{aligned} \left|\mathbb{E}\left([\hat{C}_n]_{ij}\right) - [C_n]_{ij}\right| &= O(h^2 c_n^2 (2\Omega)^{2d}) \\ \left|\mathbb{E}\left([\hat{R}_n]_{ij}\right) - [R_n]_{ij}\right| &= O(h^2 c_n^2 (2\Omega)^{2d}), \end{aligned}$$

where i, j denote the components of the matrices. Similarly, we have that

$$\begin{aligned} \mathrm{Var}\left([n\hat{C}_n]_{ij}\right) &= O\left(1/h^d\right) \\ \mathrm{Var}\left([n\hat{R}_n]_{ij}\right) &= O\left(1/h^d\right). \end{aligned}$$

Proof. By the triangle inequality and a change of variables

$$\begin{aligned} &\mathrm{Bias}(\hat{C}_n{}^{11}) = \\ &\quad \left|\int_{\mathcal{B}^d_{0,\Omega/h}} \frac{1}{h^d} K\left(\frac{\phi(z) - \phi(0)}{h}\right) F(z) dz - \int_{\mathcal{B}^d_{0,\Omega}} K(d_u\phi \cdot u) F(0) du\right| \\ &\quad \leq \left|\int_{\mathcal{B}^d_{0,\Omega}} \left[K\left(\frac{\phi(hu) - \phi(0)}{h}\right) - K(d_u\phi \cdot u)\right] F(hu) du\right| \\ &\qquad + \left|\int_{\mathcal{B}^d_{0,\Omega}} K(d_u\phi \cdot u)\left[F(hu) - F(0)\right] du\right| = T_1 + T_2 \end{aligned}$$

The Taylor remainder theorem implies that,

$$\begin{aligned} K\left(\frac{\phi(hu)-\phi(0)}{h}\right) &= K(d_u\phi\cdot u) + \partial_k K(d_u\phi\cdot u) \\ &\quad \times (h\partial_{ij}\phi^k|_0 u^i u^j/2 + h^2\partial_{ijm}\phi^k|_w u^i u^j u^m/6) \\ &\quad + \partial_{kl}K(v)/2 \times (h\partial_{ij}\phi^k|_0 u^i u^j/2 \\ &\quad + h^2\partial_{ijm}\phi^k|_w u^i u^j u^m/6) \times (h\partial_{ij}\phi^l|_0 u^i u^j/2 \\ &\quad + h^2\partial_{ijm}\phi^l|_w u^i u^j u^m/6), \end{aligned}$$

where $w \in \overline{\mathcal{B}^d_{0,\Omega}}$ and $v \in \overline{\mathcal{B}^d_{d_u\phi\cdot u, h\partial_{ij}\phi^k|_0 u^i u^j/2 + h^2\partial_{ijm}\phi^k|_w u^i u^j u^m/6}}$, and

$$F(hu) = F(0) + h\partial_i F|_0 u^i + h^2\partial_{ij}F|_v u^i u^j/2,$$

where $v \in (0, hu)$.

The odd-symmetry components of the integrands of T_1 and T_2 will be equal to zero, and so we only need to consider even-symmetry terms of the integrands. Recall that $K(\cdot), \partial_k K(\cdot), \partial_{kl}K(\cdot)$ are respectively even, odd, and even. By the sparsity assumptions, we have that

$$\begin{aligned} T_1 &= O(h^2 d^6 c_n^2 (2\Omega)^d) \\ T_2 &= O(h^2 d^2 (2\Omega)^d). \end{aligned}$$

Consequently, $T_1 + T_2 = O(h^2 d^6 c_n^2 (2\Omega)^d) = O(h^2 c_n^2 (2\Omega)^{2d})$.

We can compute the variance of $n\hat{C}_n{}^{11}$ to be

$$\begin{aligned} \mathrm{Var}(n\hat{C}_n{}^{11}) &= \int_{\mathcal{B}^d_{0,\Omega/h}} h^{-2d}[K\left((\phi(z)-\phi(0))/h\right)]^2 (F(z))^2 dz \\ &\quad - (\mathbb{E}(n\hat{C}_n{}^{11}))^2 \\ &= h^{-d}\int_{\mathcal{B}^d_{0,\Omega}} [K\left((\phi(hu)-\phi(0))/h\right)]^2 (F(hu))^2 dy \\ &\quad - (\mathbb{E}(n\hat{C}_n{}^{11}))^2 \\ &= O(1/h^d). \end{aligned}$$

The remainder of the results follow by similar, lengthy calculations. Note that for the variance of terms involving Y_i, a σ^2 coefficient appears, but this is just a finite-scaling factor which is irrelevant in O-notation. □

Proof of Theorem 5.4.1. The key to this proof is to provide an exponential concentration inequality for the terms in $\hat{C}_n$ and $\hat{R}_n$. Having done this, we can then piggyback off of the proof in [32] to immediately get the result. The proofs for $\hat{C}_n$ and $\hat{R}_n$ are identical; so we only do the proof for $\hat{C}_n$.

Using the Bernstein inequality [139] and the union bound,

$$\begin{aligned} &\mathbb{P}\left(\max_{i,j}\left\|\left[\hat{C}_n\right]_{ij} - \mathbb{E}\left[\hat{C}_n\right]_{ij}\right\| > t\right) \\ &\qquad \leq 2p^2 \exp\left(-\frac{nt^2}{2\mathrm{Var}\left(n\left[\hat{C}_n\right]_{ij}\right) + \max\left(\left|n\left[\hat{C}_n\right]_{ij}\right|\right) 2t/3}\right). \end{aligned}$$

Since the i-th component of X obeys: $|[X]_i| \leq M$, it follows that

$$\max\left(\left|n\left[\hat{C}_n\right]_{ij})\right|\right) = 2M/h^{\eta},$$

where $\eta \in \{0,1,2\}$ depending on i and j. Using this bound and Lemma A.0.6 gives

$$\max_{i,j}\left|\left[\hat{C}_n\right]_{ij} - \mathbb{E}\left[\hat{C}_n\right]_{ij}\right| = O_p(\sqrt{\log p/nh^d}).$$

Recall that

$$\max_{i,j}\left|[\hat{C}_n]_{ij} - [C_n]_{ij}\right| \leq \max_{i,j}\left|[\hat{C}_n]_{ij} - \mathbb{E}[\hat{C}_n]_{ij}\right| + \max_{i,j}\left|\mathbb{E}\left([\hat{C}_n]_{ij}\right) - [C_n]_{ij}\right|.$$

However, this second term is $o(\sqrt{\log p/nh^d})$. Consequently,

$$\max_{i,j}\left|\left[\hat{C}_n\right]_{ij} - [C_n]_{ij}\right| = O_p\left(\sqrt{\log p/nh^d}\right). \tag{A.9}$$

Using (A.9), we can follow the proof of Theorem 1 in [32] to prove the result. □

Appendix B

Statistical System Identification of a Quadrotor Helicopter

A free body diagram of a quadrotor helicopter is seen in Figure B.1. Its dynamics are complicated and much specialized, aerodynamics knowledge is used to generate accurate models [106, 107]. Additionally, the true system can be subject to wind-based disturbances. Local linearizations can be a useful modeling technique because it does not require the use of aerodynamics knowledge and it can take into account wind-based disturbances as a time-dependent component of the system dynamics.

Though this system is does not explicitly have a manifold structure, the statistical identification tools given in Chapters 4 and 5 provide better identified models [12]. This is likely due to the inability to fully sample the state-space of high-dimensional systems leading to a situation where the system trajectories look as if they lie on a low-dimensional manifold. Arguments about high sampling rates in engineering systems negating the usefulness of manifold structure are specious because the amount of data needed grows exponentially with dimension; high sample rates will never be able to generate enough data to overcome this curse-of-dimensionality.

Forty seconds of flight data sampled at $10Hz$ was taken from the STARMAC quadrotor helicopter [106, 107]. The quadrotor was flown using a feedback linearization controller which was given reference inputs to execute a dynamic trajectory in which it changed both height and horizontal position. The quadrotor was also subject to dynamic wind-disturbances.

Though the dynamics of the quadrotor do not lie on a low dimensional manifold, this data set can be used to numerically study the performance of different estimators in the case of relatively few data points. Such situations display similar problems with collinearity and near-collinearity, and these are the situations under which the new estimators are designed to perform better. Examples of robotic systems for which the local linearization identification benefits from explicit consideration of data collinearity can be found in [209]; however, that approach uses PLS, which the new estimators outperform in the example below.

To compare the accuracy of the new estimators with that of other estimators, one-step prediction errors were computed. A real-time learning process was simulated by calculating the one-step prediction error at time $t = t_0$ using only data from $t < t_0$. One-step prediction error, as opposed to n-step prediction error, is calculated because in a real-time system with feedback the practitioner will not know the exact controller inputs ahead of time.

The one-step prediction errors, at given values of t, are shown in Table B.1. The bold value in each row is the lowest prediction error for the corresponding time. Examining the values in the

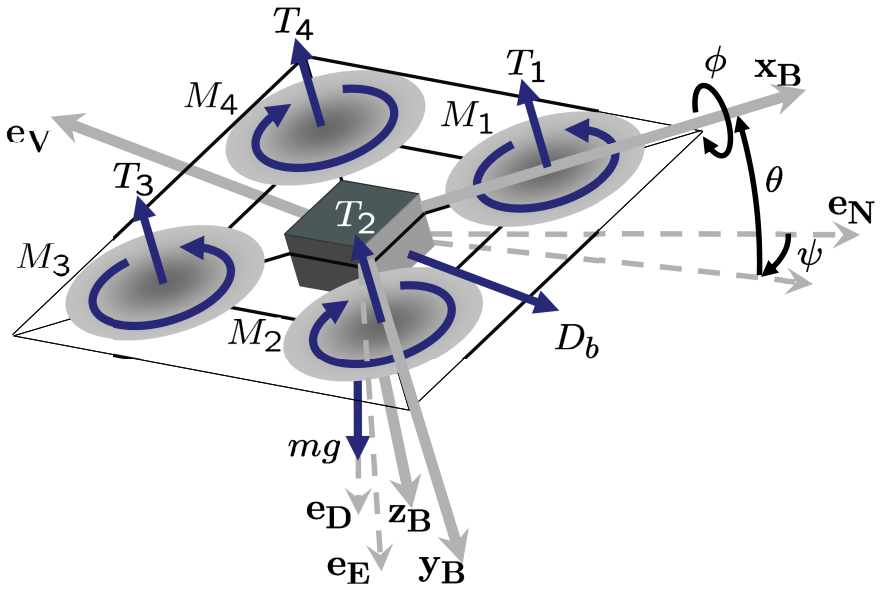


Figure B.1: The free body diagram of a quadrotor helicopter has several components: The roll, pitch, and yaw angles are ϕ, θ, and ψ. There are four motors with thrust T_i and moments M_i, for $i = 1, \ldots, 4$. The force due to gravity is m_g and the drag forces D_b are opposite to the free stream velocity e_V. This image is courtesy of [106, 107].

table, it is clear that no estimator is "best". PCR performs comparably to PLS, and PCR performs comparably to NEDE. This is not surprising, because NEDE tends towards PCR as $\lambda \to \infty$. The NALEDE, NEDEP, and NALEDEP perform well as a group, especially NALEDE and NALEDEP which are the estimators with the adaptive lasso terms. RR, EN, and MP perform decently, but do not significantly outperform the other estimators. EN also has an adaptive lasso term, and it does quite well.

time (s)	PLS	EN	MP	PCR	RR	NEDE	NALEDE	NEDEP	NALEDEP
2.0	1.0473	0.9721	2.0444	1.0517	0.9708	1.0514	1.0642	**0.9271**	1.0550
5.0	3.3119	3.2018	3.2154	3.1635	3.1883	3.1645	3.1593	2.8583	**2.8427**
10.0	2.2564	2.5515	2.1614	2.2832	2.9637	2.2832	**1.9388**	2.3030	2.0213
20.0	6.0661	6.2586	5.9916	5.9800	6.6051	5.9799	**5.9584**	6.0128	6.0140
30.0	3.3076	**2.5578**	3.0163	3.0606	2.6917	3.0606	3.2345	2.8776	2.9005
37.8	0.0184	0.1264	0.0145	0.0381	0.1338	0.0381	0.0394	0.0105	**0.0100**

Table B.1: Euclidean Norm-Squared of One-Step Prediction Error

Appendix C

Proofs for Theorems on Global-Sparsity Structure

Proof of Theorem 7.3.1. By construction, the solution to the linear averaged model $C^{ave}\beta = R^{ave}$ is given by β^{ave}. However, the solution to the limiting linear equation $\overline{C}\beta = \overline{R}$ is not immediately clear. Recall that $C(u) \succ 0$ for all $u \in \mathcal{A}$. Thus, for an arbitrary vector $\zeta \in \mathbb{R}^p \setminus \{0\}$ it holds that

$$\zeta^T \mathbb{E}(C(u))\zeta = \int_U \left[\zeta^T C(u)\zeta\right] \mu(du) > 0; \tag{C.1}$$

because we are integrating a strictly positive value over a set of non-zero measure. So, $\overline{C} = \mathbb{E}(C(u)) \succ 0$. Thus, the solution to $\overline{C}\beta = \overline{R}$ is given by $\overline{\beta}$.

All that is left to prove consistency of the linear averaged model is whether $\|X^{ave} - \overline{X}\| = O_p(a_n)$ for $X = \beta, C, R$. Note that (7.27), (7.25), and the triangle inequality imply that

$$\|\beta^{ave} - \overline{\beta}^{ave}\| \leq \|\hat{\beta}(u) - \beta(u)\| = O_p(a_n) \tag{C.2}$$

$$\|C^{ave} - \overline{C}^{ave}\| \leq \|\hat{C}(u) - C(u)\| = O_p(a_n). \tag{C.3}$$

For increasing m, Markov's inequality implies that

$$\|\overline{\beta}^{ave} - \overline{\beta}\| = O_p(m_n^{-1/2}) \tag{C.4}$$

$$\|\overline{C}^{ave} - \overline{C}\| = O_p(m_n^{-1/2}). \tag{C.5}$$

Therefore, $\|\overline{R}^{ave} - \overline{R}\| = O(m_n^{-1/2})$ by Slutsky's theorem and the triangle inequality. Applying the triangle inequality again gives:

$$\|\Xi^{ave} - \overline{\Xi}\| = O_p(a_n + m_n^{-1/2}), \tag{C.6}$$

for $\Xi = \beta, C, R$. This proves the consistency of the linear averaged model.

The remainder of the proof follows by consistency of the adaptive lasso [222]. □

Proof of Theorem 7.3.2. By construction, the solution to the linear squared averaged model

$$C^{ave}\Pi(C^{ave})^T = C^{ave}\Pi^{ave}(C^{ave})^T \tag{C.7}$$

is given by Π^{ave}. However, the solution to the limiting linear equation $\overline{C}\Pi(\overline{C})^T = \overline{C\Pi}(\overline{C})^T$ is not immediately clear. Note that the limiting linear equation can be rewritten as

$$(\overline{C} \otimes \overline{C})\text{vec}(\Pi) = \text{vec}(\overline{C\Pi}(\overline{C})^T). \tag{C.8}$$

In Theorem 7.3.1, we proved that $\overline{C} \succ 0$. Thus, the matrix $\overline{C} \otimes \overline{C}$ is invertible [198]. Thus, the solution to $\overline{C}\Pi(\overline{C})^T = \overline{C\Pi}(\overline{C})^T$ is given by $\overline{\Pi}$.

All that is left to prove consistency of the linear squared averaged model is whether $\|X^{ave} - \overline{X}\| = O_p(\gamma_n)$ for $X = \Pi, C, R$. Note that (7.27) and Slutsky's theorem imply that for fixed $u \in \mathcal{A}$:

$$\|\hat{\beta}(u)(\hat{\beta}(u))^T - \beta(u)(\beta(u))^T\| = O_p(a_n), \tag{C.9}$$

which combined with the triangle inequality yields

$$\|\Pi^{ave} - \overline{\Pi}^{ave}\| = O_p(a_n). \tag{C.10}$$

It follows from Markov's inequality that

$$\|\Pi^{ave} - \overline{\Pi}\| = O_p(a_n + m_n^{-1/2}). \tag{C.11}$$

Similar arguments show that

$$\|\Xi^{ave} - \overline{\Xi}\| = O_p(a_n + m_n^{-1/2}), \tag{C.12}$$

for $\Xi = \Pi, C, C\Pi(C)^T$. This proves the consistency of the linear squared averaged model.

The remainder of the proof follows by consistency of the adaptive group-lasso [211]. □

Appendix D

Proofs for Theorems on Monotone Piecewise-Affine Systems

Proof of Lemma 9.3.1. Define the smooth map $g(\cdot,\cdot) : \mathcal{M} \times \mathbb{R}_- \to \mathbb{R}$, $g(u,t) = e_j^T R(u,t;A,b) - \gamma_j$, and note that $R(g^{-1}(0);A,b)$ defines a level set which lies on the same hyperplane as $\mathcal{E}$. If there does not exist a $(u,t) \in \mathcal{M} \times \mathbb{R}_-$ such that $g(u,t) = 0$ and $R(u,t;A,b) \cap \mathcal{E} = R(u,t;A,b)$, then $R(\mathcal{X};A,b) \cap \mathcal{E} = \emptyset$.

Otherwise, consider the Jacobian

$$D_{u,t}g(u,t) = e_j^T \begin{bmatrix} e^{At} & A \cdot R(u,t;A,b) + b \end{bmatrix} \times \begin{bmatrix} D_u h(u) & 0 \\ 0 & 1 \end{bmatrix}. \tag{D.1}$$

By hypothesis, $e_j^T(A \cdot R(u,t;A,b) + b) \neq 0$, and so it follows that $\operatorname{rank}(D_{u,t}g(u,t)) = 1$. From Theorem 8.8 of [129] — a variant of the Implicit Function Theorem — it follows that $g^{-1}(0) \subseteq \mathcal{M} \times \mathbb{R}_-$ is an embedded submanifold of dimension d. From Corollary 8.4 of [129], there is a smooth embedding $\phi : \mathbb{R}^d \to \mathcal{M} \times \mathbb{R}_-$ such that $g^{-1}(0) = \phi(\mathbb{R}^d)$.

Next, observe that the Jacobian of the smooth map $\zeta(\nu) = (R \circ \phi)(\nu)$ is given by

$$D_\nu \zeta(\nu) = \begin{bmatrix} e^{A\phi(\nu)_t} & A \cdot R(\phi(\nu);A,b) + b \end{bmatrix} \times \begin{bmatrix} D_u h(\phi(\nu)_u) & 0 \\ 0 & 1 \end{bmatrix} D_\nu \phi(\nu), \tag{D.2}$$

where $\phi(\nu)_u$ denotes the first d components of $\phi(\nu)$ and $\phi(\nu)_t$ denotes the $(d+1)$-th component of $\phi(\nu)$. Recall two facts from linear systems theory: A and e^{At} commute and $\int_0^t Ae^{-A\tau}d\tau = I - e^{-At}$. Using these facts, we can rewrite (D.2) as

$$D_\nu \zeta(\nu) = e^{A\phi(\nu)_t} \begin{bmatrix} D_u h(\phi(\nu)_u) & A \cdot h(\phi(\nu)_u) + b \end{bmatrix} D_\nu \phi(\nu). \tag{D.3}$$

Recall that the matrix on the left has rank n by the Spectral Mapping Theorem [41] applied to the matrix exponential. Furthermore, it holds that $\operatorname{rank}(D_\nu \phi(\nu)) = d$ since ϕ is a smooth embedding and $\operatorname{rank}(D_u h(\phi(\nu)_u)) = d$ since h is also a smooth embedding.

Note that the hypothesis condition $h(u) \in \overline{\mathcal{E}}$, with e_j normal to $\mathcal{E}$, implies that $\partial h_j / \partial u_m = 0$, for $m = \{1,\ldots,d\}$; however, by hypothesis we have that $e_j^T(A \cdot h(u) + b) \neq 0$. This implies that $A \cdot h(\phi(\nu)_u) + b$ is linearly independent of the column space of $D_u h(\phi(\nu)_u)$, and so the rank of the middle matrix is $(d+1)$. Thus, the Sylvester Inequality [41] implies that $\operatorname{rank}(D_\nu \zeta(\nu)) = d$, which means that $R \circ \phi$ is a smooth embedding on $g^{-1}(0)$. Theorem 8.3 of [129] implies that $R(g^{-1}(0);A,b)$ is an embedded submanifold of dimension d. Because $\mathcal{E}$ is connected by construction

and R is a smooth embedding, the definition of embedded submanifold implies that $R(\mathcal{X}; A, b) \cap \mathcal{E} = R(g^{-1}(0); A, b) \cap \mathcal{E}$ is an embedded submanifold of dimension d. □

Proof of Lemma 9.3.2. There are sixteen possible cases, because there are four different cases for which $\hat{f}(x)$ is defined. Let I, II, III, and IV correspond to the respective cases of (9.11), and for convenience we denote a case A-B, if x falls in case A and y falls in case B. For case I-I, the result trivially follows by Hypothesis 9.3.1.

For case II-III, suppose that the claim were false. Then there exists $x \in {\mathcal{W}_{II}}^{qr}$ and $y \in {\mathcal{W}_{III}}^{su}$, such that $x \leq_{K_m} y$ and $x_k = y_k$, but $\hat{f}(x)_k > \hat{f}(y)_k$. Let e_j be normal to $\mathcal{E}^q$ and e_m be normal to $\mathcal{E}^s$. If $e_j \neq e_m$, then we can define $\tilde{x} = x + \delta e_j - \epsilon e_m$ and $\tilde{y} = y + \delta e_j - \epsilon e_m$. For small $\delta, \epsilon > 0$, we have $\tilde{x} \in \mathcal{C}^r$, $\tilde{y} \in \mathcal{C}^s$, $\tilde{x} \leq_{K_m} \tilde{y}$, and $\tilde{x}_k = \tilde{y}_k$. Thus, $(A^r \tilde{x} + b^r)_k \leq (A^s \tilde{y} + b^s)_k$ by hypothesis. Since all of these quantities are continuous, taking the limit as $\delta, \epsilon \to 0$ gives that $(A^r x + b^r)_k \leq (A^s y + b^s)_k$, which is a contradiction. If $e_j = e_m$ and $x_j \neq y_j$, then we can define $\tilde{x} = x + \delta e_j$ and $\tilde{y} = y - \delta e_j$ and follow the same procedure to reach the same contradiction. If $e_j = e_m$ and $x_j = y_j$, then we can define $\tilde{x} = x + \delta e_j$ and $\tilde{y} = y + \delta e_j$ and follow the same procedure to reach a contradiction on Hypothesis 9.3.1.

The proofs for the remaining fourteen cases are simple modifications of the proof for case II-III. □

Proof of Lemma 9.3.3. Left and right uniqueness of solution holds for $x \in \mathcal{C}^q$ because of the Lipschitz continuity of $f(x)$ in these sets. Right uniqueness of solution holds for $x \in {\mathcal{E}^q}_i \setminus \mathcal{Z}_0$, by Theorem 2, §10 of [71]. Corollary 1, §10 and Lemma 3, §10 of [71] also imply that $\hat{f}(\psi_s(x)) = \frac{d}{dt}\psi_t(x)|_{t=s^+}$. Under the additional hypothesis, left uniqueness follows from Theorem 2, §10 of [71] applied to the system with vector field $\tilde{f}(x)$. □

Proof of Lemma 9.3.4. We construct the set $\mathcal{Z}$ as follows: Let the set $\mathcal{Z}_0$ be as given in (9.10). We define the backward reach set of $\mathcal{Z}_0$ as

$$\mathcal{Z}_1 = \bigcup_{q>0} \left(\bigcup_{i>0} R({\mathcal{E}^q}_i; A^q, b^q) \cap \overline{\mathcal{C}^q} \right), \tag{D.4}$$

where the hyperedges ${\mathcal{E}^q}_i$ are of dimension $d \leq n-2$. Notice that for $x \in \mathcal{D} \setminus \mathcal{Z}_0$, a necessary condition for $\psi_t(x) \in \mathcal{W}_V$ is that $\psi_{t^-}(x) \in \mathcal{W}_{VI}$; if this statement were not true, we would reach an obvious contradiction with (9.9). This means that the backward reach set of $\mathcal{W}_V$ is a subset of $\mathcal{W}_{VI}$. Next, we recursively define the reach set of $\mathcal{Z}_m$ as

$$\mathcal{Z}_{m+1} = \bigcup_{q>0} \left(\bigcup_{i>0} R(({\mathcal{Z}_m} \cap {\mathcal{E}^q}_i) \setminus \mathcal{Z}_V; A^q, b^q) \cap \overline{\mathcal{C}^q} \right), \tag{D.5}$$

where we now use the hyperedges ${\mathcal{E}^q}_i$ of dimension $n-1$. Repeated application of Lemma 9.1.1 and Lemma 9.3.1 gives $\mu(\mathcal{Z}_m) = 0$, for all m.

Note that $({\mathcal{Z}_m} \cap {\mathcal{E}^q}_i) \setminus \mathcal{W}_V \subset \mathcal{D} \setminus \mathcal{Z}_0$, where the hyperedges ${\mathcal{E}^q}_i$ are of dimension $n-1$. Using Lemma 9.3.3, left and right uniqueness holds for points in this set, and Lemma 1, §10 of [71] allows us to continue backwards trajectories on ${\mathcal{E}^q}_i \setminus \mathcal{W}_V$, exactly in the manner defined in (D.5). Also, points in $({\mathcal{Z}_m} \cap {\mathcal{E}^q}_i) \cap \mathcal{W}_V$, already lie on $\mathcal{Z}_0$. Thus by construction, $\mathcal{Z} \subseteq \bigcup_{m=0}^{\infty} \mathcal{Z}_m$; it follows that $\mu(\mathcal{Z}) = 0$. □

Proof of Theorem 9.4.1. This proof follows the structure of Theorem 3.2 in [101]. Let $x, y \in \mathcal{D} \setminus \mathcal{Z}_0$, where $x \leq_{K_m} y$. By definition, $\mu(\mathcal{Z}_0) = 0$. Let $v \gg_{K_m} 0$ be fixed and define $y_\epsilon = y + \epsilon v$ and $f_\epsilon(x) = f(x) + \epsilon v$. By hypothesis, Lemma 9.3.3 holds, and so there exists $\delta > 0$, such that trajectories are unique for $t \in [0, \delta]$. Thus, it follows from Corollary 1, §8 of [71] that $\Phi_t(y_\epsilon)$ — which solves the PWA system (9.3), with initial condition y_ϵ and vector-field $f_\epsilon(x)$ — is continuous with respect to ϵ, for small $\epsilon > 0$. Recall that by construction $x \ll_{K_m} y_\epsilon$, and suppose that there exists $s \in (0, \delta)$ such that $\psi_s(x) \ll \Phi_s(y_\epsilon)$ for all $t \in [0, s)$, $\psi_s(x)_i = \Phi_s(y_\epsilon)_i$, and $\psi_s(x) \leq_{K_m} \Phi_s(y_\epsilon)$.

In our present situation, we have

$$\left[\frac{d}{dt}\psi_t(x)_i - \frac{d}{dt}\Phi_t(y_\epsilon)_i\right]_{t=s^+} \geq 0, \tag{D.6}$$

which leads to

$$\hat{f}(\Phi_s(y_\epsilon))_i < \hat{f}_\epsilon(\Phi_s(y_\epsilon))_i \leq \hat{f}(\psi_s(x))_i. \tag{D.7}$$

However, combining $\psi_s(x) \leq_{K_m} \Phi_s(y_\epsilon)$ and Lemma 9.3.2 leads to the conclusion $\hat{f}(\psi_s(x)) \leq_{K_m} \hat{f}(\Phi_s(y_\epsilon))$, which is a contradiction of (D.7). Hence, for all small $\epsilon > 0$ it must be that $\psi_t(x) \ll_{K_m} \Phi_t(y_\epsilon)$, for all $t \in [0, \delta)$. Because $\Phi_t(y_\epsilon)$ is continuous with respect to ϵ, taking the limit gives $\lim_{\epsilon \to 0} \Phi_t(y_\epsilon) = \psi_t(y)$. Therefore, $\psi_t(x) \leq_{K_m} \psi_t(y)$, for all $t \in [0, \delta)$.

Under the additional hypothesis, we consider $x, y \in \mathcal{D} \setminus \mathcal{Z}$, where $x \leq_{K_m} y$. Note that Lemma 9.3.4 implies that $\mu(\mathcal{Z}) = 0$. Using Lemma 9.3.3 and (9.10), we have left and right uniqueness for all points in the trajectory. It follows from Corollary 1, §8 of [71] that $\Phi_t(y_\epsilon)$ is continuous with respect to ϵ, for small $\epsilon > 0$. We suppose that $\exists s \in \text{int}(\mathcal{T}_x \cap \mathcal{T}_{y_\epsilon})$, such that $\psi_s(x) \ll_{K_m} \Phi_s(y_\epsilon)$ for all $t \in [0, s)$, $\psi_s(x)_i = \Phi_s(y_\epsilon)_i$, and $\psi_s(x) \leq_{K_m} \Phi_s(y_\epsilon)$. The proof proceeds exactly the same as the latter hypothesis. □

Proof of Theorem 9.4.2. Under these assumptions, Theorem 1 of [125] implies that there exists P_m such that part (i) of Corollary 9.4.1 holds. Under the additional hypothesis, we have that $f(x(\xi; j, \nu))_j$ is non-decreasing for increasing ν. Because P_m is diagonal and $(P_m)_{jj} = \pm 1$, it is easy to show that $(P_m f(x(P_m \xi; j, \nu)))_j$ is non-decreasing for increasing ν. Thus, condition (a) of Hypothesis 9.3.2 is satisfied for $\tilde{f}(x) = P_m f(P_m x)$. Consequently, the second part of Corollary 9.4.1 holds. □

Appendix E

Proofs for Theorems on Negative Feedback Removal

The first lemma shows that paths that cross an odd number of inhibitory edges are equivalent to paths in $\delta(G)$ that cross from V to P.

Lemma E.0.7. A path in $\delta(G)$ that begins in V and ends in P has a preimage in G that crosses an odd number of negative edges, and the converse is also true.

Proof. 1. By definition of τ, any edge that crosses between V and P must be the image of an inhibitory edge in G - and any inhibitory edge in G must map to two edges that cross between V and P.

2. As V and P form a set cover for $\delta(G)$ and are disjoint sets, any path that crosses between V and P must cross between V and P an odd number of times.

3. Following from 1 and 2, a path in $\delta(G)$ crosses between V and P if and only if the preimage of the path in the G cross an odd number of inhibitory edges.

□

Our second Lemma considers the parity of paths with specific start and end vertices.

Lemma E.0.8. The existence of a path in $\delta(G)$ that begins at $v \in V$ and ends at $p \in P$ implies the existence of a path in G from v to $\eta^{-1}(p)$ that crosses an odd number of negative edges, and the converse is also true.

Proof. 1. By Lemma E.0.7, if we have a path in $\delta(G)$ from $v \in V$ to $p \in P$, the preimage of that path — which goes from v to $\eta^{-1}(p)$ — must cross an odd number of inhibitory edges.

2. Furthermore, if we have a path in G from v to $\eta^{-1}(p)$ that crosses an odd number of inhibitory edges, we can construct a path starting at v in $\delta(G)$ — for each edge e in the path, we choose the edge in $\tau(e)$ such that the path is consistent. Because the original path crossed an odd number of inhibitory edges, by Lemma E.0.7 our new path must end at $p \in P$.

□

Proof of Theorem 10.1.1. By definition, a negative feedback cycle in G is a path that crosses an odd number of inhibitory edges. Thus the proof follows directly from Lemma E.0.8. □