FORECASTING REGIME SHIFTS IN NONLINEAR DYNAMICAL PROCESSES

A Dissertation

Submitted to the Graduate School

of the University of Notre Dame

in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

by

Tua Agustinus Tamba

_____

M.D. Lemmon, Director

Graduate Program in Electrical Engineering

Notre Dame, Indiana

August 2015

FORECASTING REGIME SHIFTS IN NONLINEAR DYNAMICAL PROCESSES

Abstract

by

Tua Agustinus Tamba

Regime shifts refer to sudden changes in the structure and function of a system due to forces from external disturbances. Such shifts occur because the system has *alternative stable states* and external disturbances force the system's operating point to shift from one stable state to another. Examples of regime shifts include the collapse of coastal fisheries as a result of human-induced nutrient enrichment and the voltage collapse in power network due to variations in storm frequency or user demand. Due to such undesired consequences, there is a great challenge in finding methods to forecast their occurrence. The work presented in this thesis addresses this challenge using techniques from polynomial optimization. We first identify two mechanisms by which regime shifts may occur and then formulate some real-valued quantities that can be used as indicators of how close a system is to each type of regime shifts. The first regime shift mechanism will be referred to as *bifurcation-induced regime shifts* and it occurs because variation in the system's parameters exceeds a critical threshold and forces the system's equilibria to undergo a bifurcation. We use a quantity known as the *minimum distance to bifurcation* as a measure of how close a system is to this type of regime shift and then formulate polynomial optimization problems that can be used to compute the global minimum of this quantity. We show that by using techniques from algebraic geometry and polynomial optimization, the computation of this quantity in a class of nonnegative systems

with *kinetic realizations* can be simplified. The second regime shift mechanism will be referred to as *noise-induced regime shifts* and it occurs because the underlying system has multiple stable equilibria and external stochastic disturbances drive the system's state from the region of attraction (ROA) of one stable equilibrium to the ROA of an alternative stable equilibria. We use probabilistic quantities called *mean first passage times* and *safety probability* to characterize the *expected time* and the *likelihood* for this type of regime shifts to occur. We also formulate polynomial optimization problems that can be used to compute upper bounds for these quantities.

CONTENTS

FIGURES

TABLES

CHAPTER 1

INTRODUCTION

1.1   Motivation and Need

Regime shifts [132, 48, 72] refer to sudden and rapid changes in the structure or function of a system due to the presence of forces/pressures from external disturbances. Originating in the ecological systems community, the term 'regime shifts' has long been used to conceptualize the observed dramatic ecological change that can occur suddenly, and at times without warning, potentially causing drastic or irreversible shifts in ecosystem state [72, 82, 112, 104, 102]. In this concept, the change is often conceived as a shift in the system state from one locally stable regime to another, usually after external triggers or disturbances force the system towards its local stability boundary or threshold. Examples of regime shifts in ecosystems include eutrophication (excessive nutrient enrichment) of freshwater lakes that shifts a clear water lake into a turbid one [20, 131], the collapse of fish population and other species due to excessive harvesting or human exploitation [29, 114, 28], degradation of coral reefs due to seasonal hurricanes [113, 77, 5] and desertification (loss of perennial vegetation in arid and semi-arid regions) of the Sahara desert due to climate change [33, 45, 132]. Each of these shifts has the potential to disrupt the services that these ecosystems provide to society. The study and better understanding of mechanisms leading to regime shifts are therefore important to help find urgent management actions that can be taken to avoid the undesired consequences that the shifts may cause [48, 127, 46, 47, 4].

The objective of this thesis is to contribute to the study and understanding of the underlying mechanisms leading to regime shifts. Specifically, this thesis develops mathematical and computational methods to forecast the onset of regime shifts in biological and ecological systems by characterizing the likelihood that a system shifts from its current operating regime in the presence of external forces that perturb either its states or parameters. The basic approach involves the characterization of some measures of how close a system is to regime shifts. In this regard, we identify two mechanisms by which regime shifts may occur and then formulate some real-valued quantities that can be used as indicators of how close the system is to each type of regime shift. The first regime shift mechanism will be referred to as *bifurcation-induced regime shifts* which occurs because variation in the system's parameters exceed a critical value and forces the system's equilibria to undergo bifurcations. We use a quantity called the "minimum distance to bifurcation" [36] as a measure of how close a system is to this type of regime shifts. The second regime shift mechanism will be referred to as *noise-induced regime shifts* which occurs because the underlying system has multiple stable equilibria and external stochastic disturbances drive the system's state from the region of attraction (ROA) of one stable equilibrium to the ROAs of alternative stable equilibria. We use probabilistic quantities called *mean first passage times* and *safety probability* to characterize the *expected time* and the *likelihood* for this type of regime shifts to occur. The key technical approach in the developed methods is the use of sum of squares (SOS) relaxation techniques [118] to recast the computation of each of these quantities as an SOS optimization problem [120, 117].

We point out that other phenomena with similar characteristics as ecological regime shifts have also been observed in different systems or applications such as engineering, physics, economics, chemistry and biology. Examples of these phenomena includes phase transition due to variation in local thermodynamic potentials in

ferromagnetism [86], cascade of voltage collapse due to storms or sharp variations in power utilization in electrical power grids [35, 15], regime switching and crash of stock market prices due to arrivals of new price information [148] and multistability/switching behaviors in chemical reaction networks and biological systems that result from variation in reaction/interaction rates between species/population [26, 44, 23]. As in the case of ecological regime shifts, each of these phenomena also shift the system's states towards undesired and irreversible alternative equilibria. Understanding the underlying mechanisms leading to regime shifts in these systems is also important as it may help assist the development of sustainable control and management strategies. The method for characterizing regime shifts described in this thesis can therefore be used as a starting point to study and better understand regime shifts in these different areas.

1.2 Background and Prior Work

An important framework that has been adopted in various studies of regime shifts is the concept of *multistability* or *alternative stable states* from dynamical systems theory [104, 131, 99]. Within these concepts, a regime shift has been viewed as a nonlinear dynamical behavior in which a system undergoes switches/jumps between alternative stable states after a sufficiently large external disturbance pushes the system state off of its current or nominal stable regime. The underlying mechanisms or drivers governing such switches are often unknown for certain [130, 8, 126]. Nevertheless, the majority of works that studied regime shifts have usually assumed that the mechanism by which the shifts occured follows one of the following two general mechanisms.

- First, regime shifts occur through a bifurcation of the system's equilibria as external perturbations cause variation in system's parameters that exceeds some critical threshold. In this case, the regime shift is characterized by changes in the number (single or multiple) and types (stable or unstable) of system

3

equilibria. Towards the end, we will refer to such mechanisms as *bifurcation-induced regime shifts.*

- Second, regime shifts occur because the system's nominal regime (or operating point) has multiple equilibria and external stochastic disturbances drive the system's state to shift/jump between the ROAs of the competing stable equilibria. Towards the end, we will call this mechanism a *noise-induced regime shift.*

As described below, the differences between these two mechanisms are in terms of how the perturbations affect the nominal system and what impact do they cause to the nominal system.

- In bifurcation-induced regime shifts, perturbations that cause the shifts are viewed as the result of system's internal dynamics/feedbacks. In particular, the impact of these perturbations occur at slower time scales than the time scale of the system's states [130]. Examples of these perturbations are variations in individual growth/death/consumption rates of each species in a population which occur at slower time scales than the time scales of the total population's growth/death/consumption rates [129, 20]. From a modeling standpoint, these perturbations are more suitable to be viewed as slow variations in the system's parameters rather than variations in the states. As a result, any change that occurs due to these perturbations will be reflected as change in the nominal system's qualitative properties (phase portrait, number/stability of equilibria).

- In noise-induced regime shifts, perturbations that cause the shifts are viewed as the result of external forces and directly affect the system's states [74]. The impacts of these perturbation are therefore assumed to occur at relatively similar time scales as the time scales of the system's states. Examples of these perturbations are annual variation of fish population density in a lake due to seasonal storms, floods or human harvestings [17, 38]. From a modeling stand point, these perturbations are more suitable to be viewed as small variations in the system's states. Moreover, the presence of these perturbations do not cause any change on the nominal system's qualitative properties [130].

To better illustrate the aforementioned two regime shifts mechanisms, let us consider the following ordinary differential equation (ODE) model of the lake eutrophication process [20].

$$\dot{x}(t) = a - bx(t) + \frac{x(t)^2}{1 + x(t)^2}, \quad x(0) = x_0. \tag{1.1}$$

In equation (1.1), the state variable $x(t)$ denotes the Phosphorus (P) concentration in the lake water column at time t whereas parameters $a \geq 0$ and $b \geq 0$ denote the rates of inflow and outflow of P into and out of the lake, respectively. The nonlinear function $\frac{x(t)^2}{1+x(t)^2}$ in the model denotes the rate of P increase in the water column due to the recycling of P from the lake sediment. Model (1.1) will be used to further illustrate the characteristics and difference between the two regime shift mechanisms. Note that when steady state analysis is of concern, we will use a slight abuse of notation by dropping the variable time t from the model and simply use x to denote the system's states.

1.2.1   Bifurcation-induced Regime Shifts

To evaluate possible bifurcations in model (1.1), one needs to analyze the impact of parameter variation on the number and qualitative properties (i.e stability) of its equilibria. The equilibria of system (1.1) can be computed by setting the right hand side of the ODE to zero and then solving for the corresponding value of x. That is, the equilibria are given by those values of x such that

$$\underbrace{a + \frac{x^2}{1+x^2}}_{f(x)} = \underbrace{bx}_{g(x)}.$$

The left ($f(x)$) and right ($g(x)$) hand side terms in the above equality denote the rate of increase and decrease of x, respectively, and so the system's equilibria will correspond to a state with a constant value where its rates of increase and decrease are equivalent. Geometrically, these equilibria are given by those values of x at the intersections between the curves of $f(x)$ and $g(x)$ (cf. Figure 1.1).

Let us, for example, consider changes in the intersection between $f(x)$ and $g(x)$ when the parameter b is held constant while a is varied. Figure 1.1 plots the curves for a constant $b = 0.525$ and different values of a. This plot shows the change in
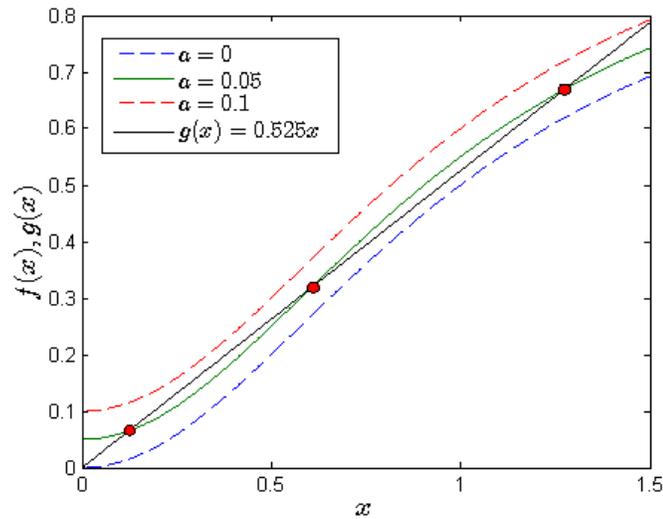
Figure 1.1. Bifurcation-induced regime shifts in lake model (1.1).

the number of intersections between the two curves (i.e. equilibria) as the value of parameter a is varied. In particular, for a small inflow rate ($0 \leq a \leq 0.025$), the system has a single *oligotrophic* equilibrium that is characterized by pristine water with high oxygen concentration that supports species biodiversity. For larger inflow rate ($0.026 < a \leq 0.075$), this equilibrium bifurcates into two stable equilibria: an oligotrophic one and a *eutrophic* one (lake water characterized by algal bloom with low oxygen concentration that does not support species biodiversity). Finally, for an even larger inflow rate ($a > 0.076$), these two stable equilibria coalesce to a single eutrophic equilibrium through a saddle-node bifurcation. This series of bifurcations therefore demonstrates a transition from a lake with clear water to a lake with turbid water.

The preceeding example illustrates that variations in the system's parameters that exceed a critical value may cause the system's equilibria to undergo bifurcations. The phenomena in which the number (from single to multiple and vice versa) and the type (stable to unstable and vice versa) of the system's equilibria change as the

parameters are varied are what we refer to as *bifurcation-induced regime shifts*. In particular, these bifurcations are followed by flips or shifts in the system's operating regime/point from one stable state to another [132].

Various studies to better understand the underlying mechanisms and possible consequences of bifurcation-induced regime shifts have been reported in the ecological systems literature (cf. [72, 104, 131, 129, 18, 27] and the references therein). Most of these studies have mainly focused on using numerical bifurcation analysis to characterized the parameter values where a bifurcation occured. The results from this analysis usually gave the set of possible equilibria and the corresponding values of the parameters where bifurcations occured. However, the currently available numerical bifurcation tools can only handle up to at most two or three parameters simultaneously. This implies that this analysis can only be done for particular choices/subsets of the entire groups of parameters [92]. The choice of such parameter subsets is usually guided by ecologically meaningful hypotheses that were developed prior to the analysis. Similar bifurcation analysis method has been used to study bistability, switching between stable equilibria and other complex nonlinear dynamics in the models of biological systems or chemical reaction networks (cf. [40, 101, 129]).

A slightly different problem in engineering that has strong relevance with the study of bifurcation-induced regime shifts is the *distance to bifurcation* (D_2B) problem. However, instead of trying to characterize all possible bifurcations that may occur in the system, the objective in solving the D_2B problem is to find the *closest* critical parameter values at which a bifurcation occurs. Specifically, for dynamical systems

$$\dot{x}(t) = f(x(t), k), \quad x(0) = x_0, \tag{1.2}$$

whose vector fields are parameterized by a vector of real-valued parameters k, one

defines a quantity

$$\gamma = \inf_{k} |k^* - k^0|, \tag{1.3}$$

to measure the distance between a nominal parameter k_0 and the closest critical paramater k^* at which a bifurcation occurs. The quantity γ is often called the *minimum D_2B* [36, 35] and is an indicator of how close a system is to a bifurcation or instability. Prior works have proposed several methods to compute γ in the context of robust stability analysis [88, 106, 110, 154] and voltage collapse in power systems [35, 36]. In general, these methods combine both numerical bifurcation analysis and optimization-based search techniques to compute the minimum γ over the bifurcation manifold [63, 92]. These methods, however, are computationally demanding since the search for the minimum γ requires the computation of the system's equilibria x^* at every iteration of the search.

This thesis presents an SOS optimization method to bound the minimum D_2B in a class of *nonnegative systems with kinetic realizations*. A dynamical system $\dot{x}_i(t) = f_i(x, k)$ with polynomial vector fields $f_i(x, k)$ is said to have a kinetic realization if there exist polynomials $g_i(x, k)$ and $h_i(x, k)$ with nonnegative coefficients such that

$$f_i(x, k) = g_i(x, k) - x_i h_i(x, k). \tag{1.4}$$

The condition in equation (1.4) was presented in [67] and it essentially guarantees that a polynomial dynamical system is a *mass action systems* for which a chemical reaction network (CRN) graph can be realized. The existence of such realizations implies that the system's vector fields $f(x, k)$ satisfy a decomposition of the form $f(x, k) = Nv(x, k)$ where N is a real-valued matrix and $v(x, k)$ is a nonnegative-valued vector of monomials in x. As will be discussed in Chapter 3, the special structure of kinetic systems allows one to compute an expression for their equilibria in terms of the system's parameters. We show that such an equilibrium expression

can simplify the D_2B problem because the constraints that define the bifurcation conditions can now be expressed only in terms of the system's parameters, rather than the system's parameters and equilibria. By formulating the bifurcation conditions using semialgebraic set descriptions in the parameter space, we show that the SOS optimization techniques introduced in [117, 118] can be used to compute a lower bound for the globally minimum D_2B.

1.2.2 Noise-induced Regime Shifts

Consider again the lake model (1.1). Let us set the parameters to be $a = 0.06$ and $b = 0.525$. For this choice of parameters, the system has two stable equilibria (oligotrophic and eutrophic) that are separated by one unstable equilibrium. Figure 1.2a plots the phase portrait and the equilibria of the system for the chosen parameters. The ROA of each equilibrium of the deterministic system is the area to the left or to the right of the dashed line marked as the separatrix. In this case, any trajectory (or sample path) that starts from the region to the left (right) hand side of the separatrix line will remain in that region and eventually settle to the oligotrophic (eutrophic) equilibrium. Now assume that the system is driven by an additive Wiener process $\{w(t)\}$ of constant variance σ. The system's state is now a random process $\{x(t)\}$ modeled by a stochastic differential equation (SDE)

$$
\begin{aligned}
dx(t) &= f(x)dt + g(x)dw(t), \\
&= \left(a - bx + \frac{x^2}{1 + x^2}\right) dt + \sigma dw(t),
\end{aligned}
\tag{1.5}
$$

where $f(x)$ and $g(x)$ are the *drift* and *diffusion* terms, respectively [85, 91].

Figure 1.2b plots one realization of $\{x(t)\}$ for a fixed $\sigma = 0.05$. The horizontal dashed line in this plot marks the separatrix x_U^* from Figure 1.2a. Figure 1.2b shows that the process' sample path that starts inside the region below the dashed line
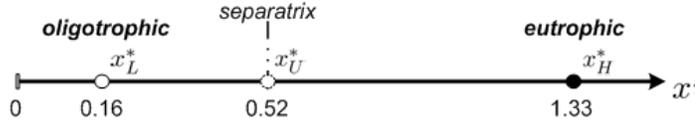
(a) Phase portrait of (1.5) for $a = 0.06$, $b = 0.525$.
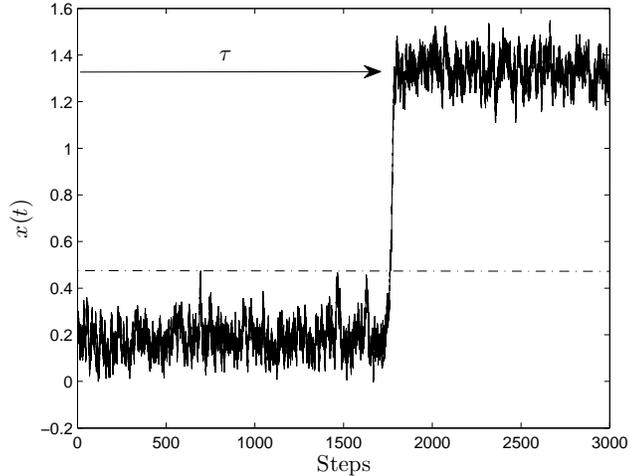
(b) One sample path of (1.5) for $\sigma = 0.05$.

Figure 1.2. Noise-induced regime shifts in lake model (1.5).

(oligotrophic lake) eventually shifts to the region above the dashed line (oligotrophic lake). This example illustrates one mechanism of noise-induced regime shifts where the presence of noise in systems with multiple equilibria may force the system's state to shift from the ROA of one stable state to the ROA of an alternative stable states.

The above example suggests that the mechanism of noise-induced regime shifts can at least be characterized in terms of the *expected time* at which a shift may occur and the *probability* for a shift to occur over a given finite time interval. In particular, the characterization of the expected time and probability to shift can be formulated using *mean first passage time* (MFPT) and *finite time stochastic safety* analyses, respectively. Each of these analyses is formally stated as follows.

- *MFPT analysis:* Let $\{x(t)\}$ be a stochastic process whose state $x(t)$ at time $t \geq 0$ takes values on a bounded open subset $\mathcal{X} \subseteq \mathbb{R}^n$ of the Euclidean space with smooth boundary $\partial \mathcal{X}$. Let $\mathcal{X}_0 \subset \mathcal{X}$ be an initial set such that $x(0) = x_0 \in$

$\mathcal{X}_0$. The time at which the sample paths of $\{x(t)\}$ hit the set $\partial\mathcal{X}$ is a random variable τ called the first passsage time and is defined as

$$\tau \equiv \inf_t \{t \geq 0 \mid x(t) \in \partial\mathcal{X}\}. \tag{1.6}$$

The MFPT analysis is concerned with the computation of the expected value $\mathbb{E}\{\tau\}$ of τ.

- *Finite time stochastic safety analysis:* Let $\{x(t)\}$ be a stochastic process whose state $x(t)$ at time $t \geq 0$ takes values on a bounded open subset $\mathcal{X} \subseteq \mathbb{R}^n$ of the Euclidean space. Let $\mathcal{X}_0 \subset \mathcal{X}$ be an initial set such that $x(0) = x_0 \in \mathcal{X}_0$ and let $\mathcal{X}_s \subseteq \mathcal{X}$ denote an arbitrary safe/desired set within the state space such that $\mathcal{X}_0 \subset \mathcal{X}_s$. The finite time stochastic safety analysis characterizes the probability that, starting from inside the initial set $\mathcal{X}_0$, the sample paths $x(t)$ of the process leave the safe set $\mathcal{X}_s$, at least once, in a finite time $t \in [0, T]$. Formally, this analysis seeks to compute $\beta \in [0, 1]$ such that

$$\mathbb{P}\{x(t) \notin \mathcal{X}_s, \text{ for some } 0 \leq t \leq T \mid x(0) \in \mathcal{X}_0\} \leq \beta. \tag{1.7}$$

Several studies have tried to characterize noise-induced regime shifts using quantities in equations (1.6)-(1.7). The work in [16, 24] uses Monte Carlo (MC) simulations [89] to compute the MFPT for noise-induced regime shifts in models of lake eutrophication whereas the works in [37, 60, 116] use the analytical solution to the Fokker Planck (FP) equation [125, 54, 74] to characterize the MFPT in bistable models of gene expression and intraguild predation systems. Analytical solution to the FP equation and MC simulation have also been used for stochastic safety analysis as defined in (1.7). One should note that solution methods based on the MC simulation and the FP equation have their own difficulties. The FP equation is difficult to solve for systems with dimensionality greater than one as it involves solving a set of partial differential equations with appropriate boundary conditions [125, 54]. On the other hand, the MC simulation is computationally expensive as it requires exhaustive simulations of the process's sample paths to estimate the process' statistics.

This thesis presents an extension of the methods introduced in [91, 122] to compute upper bounds for the quantities in (1.6)-(1.7) in systems that are modeled as jump diffusion processes. As in the case of diffusion processes [91, 122], the tech-

niques developed in this thesis are also based on searching for a barrier certificate, $V(x(t))$, that generates a supermartingale from which the bounds for (1.6)-(1.7) can be deduced. The main contribution of the proposed method is a polynomial characterization of the jump diffusion process' infinitesimal generators which allows the use of SOS optimization technique to compute the bounds for (1.6)-(1.7).

1.3 Approach and Contribution

This thesis presents mathematical and computational methods for analyzing and forecasting regime shift phenomena in nonlinear system models. The main contributions of this thesis are stated below.

1.3.1 Classification of Regime Shifts Mechanisms

Although there exists an overwhelmingly large literature on the study and characterization of regime shifts mechanisms, there currently exist no clear frameworks by which external perturbation that cause regime shifts to occur should be viewed or understood. As discussed previously, various works in the literature often submit to a view/approach that exclusively treat/study regime shifts as *either* a bifurcation problem *or* a noise-induced transition problem. Such a confined view of regime shifts can be problematic as it has the potential to limit the very definition of regime shifts which includes any sudden and rapid change in a system's structure/function in the presence of forces from external perturbations, regardless of how the perturbations affect the system.

As discussed in Section 1.2, this thesis provides a classification of mechanisms (i.e. bifurcation-induced and noise-induced) by which regime shifts may occur. The underlying reasoning for such a classification is a careful understanding of how the perturbations affect the nominal system and what impact these perturbations have on the nominal system. The main advantage of such a classification is that it provides

clear guidance for looking at the established solution methods that can be used to better understand the regime shift phenomena.

1.3.2 Polynomial Optimization Methods to Forecast Regime Shifts

The second contribution of this thesis is the development of mathematical and computational methods to forecast the onset of a possible regime shift. As described in Section 1.2, the main approach in the developed methods is the formulation of some real-valued quantities that can be used as indicators of how close a system is to either bifurcation-induced or noise-induced regime shifts. In bifurcation-induced regime shifts, this quantity is called the *minimum distance to bifurcation*, $\gamma \in \mathbb{R}_{\geq 0}$, and it measures the distance between nominal parameter values and their values at which a bifurcation will possibly occur [36]. In noise-induced regime shifts, these quantities are the *mean first passage time* (MFPT) and the *finite time safety probability* defined in equations (1.6) and (1.7), respectively [54, 91]. The key technical tool in the proposed methods is the use of sum of squares (SOS) relaxation techniques [118] to recast the polynomial optimization problems that correspond to the computation of each of these quantities [120, 117]. The main advantage of this formulation is that the resulting SOS optimization problem can be solved using semidefinite programming solvers [121, 100, 69]. This contribution is detailed below for each of the regime shifts mechanisms.

1.3.2.1 Bifurcation-induced Regime Shifts

In the case of bifurcation-induced regime shifts, the contribution of the thesis is on the use of SOS optimization methods [117, 118] to compute a lower bound on the distance to bifurcation γ in (1.3) for the class of nonnegative systems with *kinetic realizations*. A dynamical system $\dot{x}_i(t) = f_i(x, k)$ with polynomial vector fields $f_i(x, k)$ is said to have a kinetic realization if there exists polynomials $g_i(x, k)$

and $h_i(x, k)$ with nonnegative coefficients such that $f_i(x, k)$ can be rewritten as [67]

$$f_i(x, k) = g_i(x, k) - x_i h_i(x, k).$$

An important property of systems with this special structure is that it allows one to compute an expression for the system's equilibria in terms of the system's parameters and some convex parameters (cf. Chapter 3). The method to compute this expression is based on concepts and techniques from algebraic geometry and was first proposed in [57] for the study of chemical reaction network. As discussed in Chapter 4, the use of such an equilibrium expression can helps simplify the computation of a lower bound for the distance to bifurcation γ.

This thesis' first contribution in this topic is the identification of larger class of systems (other than chemical reaction network model) for which the equilibrium parameterization method introduced in [57] can be applied. We call these systems *nonnegative systems with kinetic realizations* and show through example how the parameterization method from [57] can be applied to them. The thesis' second contribution in this topic is the use of such equilibrium parameterizations to formulate an SOS optimization problem for computing a lower bound on the minimum distance to bifurcation, γ. We show that the use of such parameterizations helps simplify the computation of γ's lower bound because the constraints in the optimization problem can now be expressed only in term of the parameters rather than the parameters and the equilibria.

1.3.2.2 Noise-induced Regime Shifts

In noise-induced regime shifts, the contribution of the thesis is on the use of Lyapunov like methods to characterize the quantities in (1.6)-(1.7). This method is commonly used in stochastic stability analysis [91] and essentially searches for a

"stochastic" Lyapunov function from which upper bounds for quantities in (1.6) and (1.7) can be deduced. Using the framework introduced in [122], we show that the computation of these quantities can be formulated as an SOS optimization problem.

The main contribution of this thesis is the extension of the method in [122] to compute upper bounds of (1.6)-(1.7) for systems that are modeled as jump diffusion processes. The use of jump diffusion models (rather than pure diffusion) was motivated by the fact that many noise-induced regime shifts are triggered by extreme or abnormal events that result in jumps or discontinous changes in the system's states [130, 19]. These events are no longer suitable to be described by a Wiener process but are better characterized as a stochastic renewal process in the form of jump diffusion processes. By using the polynomial representation of the jump diffusion process' infinitesimal generator, we show that the computation of upper bounds for quantities in (1.6)-(1.7) can be formulated as an SOS optimization problem.

1.3.3 Software Toolkit

The third contribution of this thesis is a software toolkit that can be used to assist the analysis of bifurcation-induced regime shifts in nonnegative systems with kinetic realizations [147]. Some of the features in the toolkit include the construction of kinetic realization for given ordinary differential equations or a set of elementary chemical reactions, the computation of equilibrium parameterization based on the systems' kinetic realization and the computation of lower bounds for minimum distance to bifurcation. The toolkit is developed under MATLAB and it integrates several computational tools [32, 87] that are required for the bifurcation-induced regime shifts analysis. The developed toolkit is described in Appendix C.

1.4   Outline of Thesis

The thesis is structured as follows. Chapter 2 presents mathematical background and preliminaries on algebraic geometry and relaxation methods for solving polynomial optimization problem. This material is essential for the methods we use to forecast bifurcation-induced regime shifts. In Chapter 3, we present the concept of kinetic realizations that arise from the models of chemical reaction network (CRN) and discuss a parameterization method that can be used to compute an expression for the system's equilibria. We show that such a parameterization can be applied to a larger class of systems other than CRN models and discuss its use in analyzing the system's properties. Chapter 4 presents a method to characterize bifurcation-induced regime shifts in the class of nonnegative systems with kinetic realizations. We formulate the prediction of such regime shifts as a distance to bifurcation problem and show that the solution to this problem for a class of nonegative systems with kinetic realizations can often be simplified using the equilibrium parameterization method described in Chapter 3. Chapter 5 presents a method to characterize noise-induced regime shifts in systems that are modeled as jump diffusion processes (JDP). The approach used to forecast this type of regime shift is based on formulating the prediction either as a mean first passage time problem or as a stochastic safety analysis. The main result in this chapter is the construction of a polynomial representation for the jump JDP's infinitesimal generator which allows the use of SOS optimization methods to compute upper bounds for both the MFPT and the safety probability. Chapter 6 describes an application of the proposed method to study regime shift in a model identified from an experiment on an ecological system test bed. The test bed is a laboratory scale chemostat that cultures a microbial predator and prey system between green algae, *Chlorella vulgaris*, and rotifer, *Brachionus calyciflorus*. This chapter shows that the proposed method can be used to compute a lower bound on the distance to a regime shift in the model of *Chlorella vulgaris* and *Brachionus calyciflorus* interaction.

CHAPTER 2

ALGEBRAIC GEOMETRY AND POLYNOMIAL OPTIMIZATION

2.1 Introduction

As mentioned in Chapter 1, the approach used to forecast regime shifts is based on computing some real-valued quantities that characterize the onset of either bifurcation-induced or noise-induced regime shifts. In particular, these quantities were computed from the solutions of polynomial optimization problems. From the formulation's standpoint, the polynomial optimization set up arises naturally from the fact that many real life systems such as those in biology, ecology and chemistry are often modeled as polynomial dynamical systems [64, 54, 44]. This implies that the conditions for the occurrence of regime shifts can be formulated as semialgebraic sets (cf. Chapters 3-5). From the solution approach's standpoint, the polynomial optimization problems have an attractive property in that, even if they cannot be solved exactly, their approximate solutions (i.e. lower or upper bounds) can be computed from convex relaxations of the original problem. This approach was based on recent developments in convex relaxation methods for solving polynomial optimization problems [117, 142]. This implies that the proposed method can take advantage of further developments in these areas.

In general, the polynomial optimization problems discussed in this thesis take the following form

$$
\begin{aligned}
\min \quad & p(x) \\
\text{s.t.} \quad & F_i(x) = 0, \quad i = 1, \ldots, N_F, \\
& G_j(x) \geq 0, \quad j = 1, \ldots, N_G,
\end{aligned}
\tag{2.1}
$$

where $p(x), F_i(x), G_j(x)$ are some polynomial functions and N_F and N_G denote the number of polynomial equalities and inequalities, respectively, characterizing the problem's feasible set. For the regime shift analyses discussed in this thesis, the objective function $p(x)$ in (2.1) is a real-valued function that characterizes the likelihood of a regime shift occurring whereas functions $F_i(x)$ and $G_j(x)$ are constraints that must be satisfied when a regime shift occurs.

This chapter discusses mathematical background from algebraic geometry that is used to recast the polynomial optimization problem (2.1) into a convex optimization problem [117, 142]. The first part of this chapter (Section 2.2) covers such concepts as the Gröbner basis (which can be used to simplify a set of polynomial equations $F_i(x) = 0$) and the positivstellensatz's theorem (which can be used to check the existence of solutions/zeros of polynomial equations over a semialgebraic set) [25]. The second part of this chapter then discusses the use of the positivstellensatz's theorem to recast the polynomial optimization problem (2.1) into an equivalent convex optimization problem. Examples are given throughout the text to help illustrate the presented concepts.

The background discussed in this chapter will be useful for the presentations in Chapters 3-5. The presented materials are mostly drawn from existing papers [25, 10, 118, 117] and therefore the interested readers are urged to refer to these references for a more detailed exposition.

Notational convention: Let $\mathbb{R}$, $\mathbb{C}$, $\mathbb{Q}$ and $\mathbb{Z}$ denote the set of real, complex, rational and integer numbers, respectively. The set of *nonnegative* real, rational, and integer numbers are denoted as $\mathbb{R}_{\geq 0}, \mathbb{Q}_{\geq 0}$ and $\mathbb{Z}_{\geq 0}$, respectively, whereas the set of *positive* real, rational, and integer numbers are denoted as $\mathbb{R}_+, \mathbb{Q}_+$ and $\mathbb{Z}_+$, respectively. $\mathbb{R}^n$ denotes the n-dimensional Euclidean space. For a vector $x \in \mathbb{R}^n$, we use x_i, $i = 1, \ldots, n$, to denote the ith component of x.

An n-dimensional *multi-index* $\alpha \in \mathbb{Z}_{\geq 0}^n$ is an n-tuple $\alpha \equiv (\alpha_1, \alpha_2, \cdots, \alpha_n)$ of non-negative integers with an absolute value of $|\alpha| = \sum_{i=1}^n \alpha_i$. For two multi-indices $\alpha, \beta \in \mathbb{Z}_{\geq 0}^n$, we say that $\alpha \geq \beta$ if and only if $\alpha_i \geq \beta_i$ for $i = 1, 2, \ldots, n$. The sum/difference of two multi-indices is the component-wise sum/difference of the indices.

Given a vector $x \in \mathbb{R}^n$ and a multi-index $\alpha \in \mathbb{Z}_{\geq 0}^n$, a *monomial* in variables x with a total degree of $|\alpha|$ is a product of the form $x^\alpha \equiv x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$. We use *monomial ordering* to unambiguously arrange a pair of monomials in an ascending (or descending) order. Let $\alpha, \beta \in \mathbb{Z}_{\geq 0}^n$ be multi-indices of vector $x \in \mathbb{R}^n$. Let $>$ denotes a total ordering on $\mathbb{Z}_{\geq 0}^n$ such that for $\alpha, \beta \in \mathbb{Z}_{\geq 0}^n$ at least one of the following conditions is true: $\alpha > \beta$, $\alpha < \beta$, $\alpha = \beta$. Then a monomial ordering in $\mathbb{Z}_{\geq 0}^n$ is a total ordering if (i) given multi-indices $\gamma, \alpha, \beta \in \mathbb{Z}_{\geq 0}^n$ with $\alpha > \beta$, then $\alpha + \gamma > \beta + \gamma$, (ii) $>$ is a *well-ordering* such that every nonempty subset of $\mathbb{Z}_{\geq 0}^n$ has a smallest element. One of the most frequently used monomial orderings is the *lexicographic* (lex) order and is denoted as $>_{lex}$. We say that α and β is in lex order, $\alpha >_{lex} \beta$, if the left-most nonzero entry of the vector difference $\alpha - \beta$ is positive. Thus, $x^\alpha >_{lex} x^\beta$ holds if $\alpha >_{lex} \beta$. Another commonly used monomial ordering is the *graded reverse lexicographic* (grlex) order and is denoted as $>_{grlex}$. We say that α and β is in grlex order, $\alpha >_{grlex} \beta$, if $|\alpha| > |\beta|$ or $|\alpha| = |\beta|$ and the right-most nonzero entry of the difference $\alpha - \beta$ is negative.

The set of $n \times n$ real symmetric matrices is denoted as $\mathcal{S}^n$. A matrix $Q \in \mathcal{S}^n$ is *positive semidefinite* (psd) if $x^T Q x \geq 0$ for all $x \in \mathbb{R}^n$ and is *positive definite* (pd) if $x^T Q x > 0$ for all nonzero $x \in \mathbb{R}^n$. The set of $n \times n$ psd matrices is denoted as $\mathcal{S}_{\geq 0}^n$ and the set of $n \times n$ pd matrices is denoted as $\mathcal{S}_+^n$. We use the symbol "$\succeq$" to denote the partial order induced by $\mathcal{S}_{\geq 0}^n$ such that if $Q, R \in \mathcal{S}_{\geq 0}^n$, then $Q \succeq R$ if and only if $Q - R$ is psd.

2.2 Algebraic Geometry

Algebraic geometry is concerned with the the study of solutions (or zeros) of systems of polynomial equations. It does this by taking into consideration the close relationship between the geometric properties of these solutions and the algebraic structures associated with them [25]. Some fundamental concepts in algebraic geometry are *ideal*, *variety*, and *basis*. This section presents a review of these concepts and discusses the use of *Gröbner bases* in solving a system of polynomial equations. Most of the following material is drawn from [25, 103].

2.2.1 Ideals, Varieties, and Gröbner Basis

A fundamental concept in algebra is a *field*. Intuitively, a field consists of a set $\mathbb{K}$ and binary operations addition ($+$) and multiplication ($\cdot$) such that if $a, b \in \mathbb{K}$ then their binary operations satisfy the associative, commutative, distributive, identitites, additive inverse, and multiplicative properties [103]. Examples of fields include the sets of real ($\mathbb{R}$), complex ($\mathbb{C}$) and rational ($\mathbb{Q}$) numbers. The set of integer numbers ($\mathbb{Z}$) is not a field since the only elements of $\mathbb{Z}$ that have multiplicative inverses are 1 and -1 (e.g. the multiplicative inverse of 2 is 1/2 but $1/2 \notin \mathbb{Z}$).

Let $\mathbb{K}$ be any field. A dth order *polynomial* in n unknown variables $x \in \mathbb{R}^n$ with coefficients $k \in \mathbb{K}$ is a finite linear combination of monomials of the form

$$f(x, k) = \sum_{|\alpha| \leq d} k_\alpha x^\alpha, \text{ with } k_\alpha \in \mathbb{K},$$

where α is an $n-$dimensional multi-index. The set of all such polynomials forms a *polynomial ring* and is denoted as $\mathbb{K}[x]$. We use the symbol $\mathbb{K}(k)[x]$ to denote a polynomial ring with unknown coefficients k in the field $\mathbb{K}$ (cf. Chapters 3-4). Unless stated otherwise, we will use the phrase "polynomials over $\mathbb{K}$" to denote the set of polynomials whose coefficients take values in the field $\mathbb{K}$, i.e. $f_i(x, k) \in \mathbb{K}[x]$ for

$i = 1, \ldots, m$. We will mostly consider polynomials over $\mathbb{R}$ for which the polynomial ring will simply be denoted as $\mathbb{R}[x]$.

Given a set of polynomial equations $f_i(x, k) = 0$, $i = 1, \ldots, n$, we define the *solution* of these equations as the set $\{x \in \mathbb{K}^n : f_i(x, k) = 0\}$ of common zeros of the polynomials in $\mathbb{K}^n$.

Let $\alpha \in \mathbb{Z}_{\geq 0}^N$ be a multi-index and assume a fixed monomial ordering. For a given nonzero polynomial $f(x, k) \in \mathbb{K}[x]$, we define

- The *multidegree* of $f(x)$ as: $\mathtt{multideg}(f) = \max(\alpha \in \mathbb{Z}_+), k_\alpha \neq 0$
- The *leading monomial* of f as: $\mathtt{LM}(f) = x^{\mathtt{multideg}(f)}$.
- The *leading coefficient* of f as: $\mathtt{LC}(f) = k_{\mathtt{multideg}(f)}$.
- The *leading term* of f as: $\mathtt{LT}(f) = \mathtt{LC}(f).\mathtt{LM}(f)$

Example 1 below illustrates how to determine each of the above definitions.

Example 1. *Consider polynomial $f(x, k) = 2x_1^2 x_2^8 - 3x_1^5 x_2 x_3^4 + x_1 x_2 x_3^3 - x_1 x_2^4$ in $\mathbb{R}[x]$ where $x = (x_1, x_2, x_3)$ and $k \in \mathbb{R}$. By choosing the lex ordering for its monomials, we can rewrite the monomials of $f(x)$ in a decreasing order as $f = -3x_1^5 x_2 x_3^4 + 2x_1^2 x_2^8 - x_1 x_2^4 + x_1 x_2 x_3^3$. We then have $\mathtt{multideg}(f) = (5, 1, 4)$, $\mathtt{LM}(f) = x_1^5 x_2 x_3^4$, $\mathtt{LC}(f) = -3$, $\mathtt{LT}(f) = -3x_1^5 x_2 x_3^4$.*

Definition 1 formally defines an *ideal* in $\mathbb{R}[x]$.

Definition 1 ([25]). *The set $I \subseteq \mathbb{R}[x]$ is an* ideal *if it satisfies: (i) $0 \in I$, (ii) $\forall a, b \in I \Rightarrow a + b \in I$, and (iii) $\forall a \in I, b \in \mathbb{R}(k)[x] \Rightarrow a \cdot b \in I$.*

Let $f_1, \ldots, f_m$ be polynomials in $\mathbb{R}[x]$ and set

$$I = \{\Sigma_{i=1}^m h_i f_i, \text{ with } h_i \in \mathbb{R}[x]\} = \langle f_1, \ldots, f_m \rangle \subseteq \mathbb{R}[x].$$

It can be shown that I is an ideal *generated* by $f_1, \ldots, f_m$ [25]. For an ideal $I = \langle f_1, \ldots, f_m \rangle$, the set of polynomials $f_1, \ldots, f_m$ is a *generator* of I. The set of all

possible generators of I is called the *basis* of I. One advantage in using the concept of an ideal is that it allows the use of different choices of generators for representing an ideal. This advantage was due to the fundamental result in Hilbert's Basis Theorem [25]. Hilbert's Basis Theorem states that every ideal $I \subseteq \mathbb{K}[x]$ is finitely generated (hence it has a finite number of basis) and therefore can always be expressed as $I = \langle f_1, \ldots, f_m \rangle$. By the definition of an ideal, then Hilbert's Basis Theorem implies that all polynomials of the form $\Sigma_{i=1}^m h_i f_i$ with $h_i \in \mathbb{K}[x]$ are also ideals generated by f_i's. Working with ideals therefore removes the dependency on a particular generator and allows for the use of any choice of generator that is more suitable with the context of the problem [25].

Now let $I = \langle f_1, \ldots, f_m \rangle \subseteq \mathbb{R}[x]$ be an ideal. For $1 \leq i \leq m$, the set

$$\mathbb{V}(I) = \mathbb{V}(f_1, \ldots, f_m) = \{x \in \mathbb{C}^n : f_i(x) = 0\},$$

is called an *algebraic variety* (or simply variety) of I generated by f_i's. Essentially, a variety $\mathbb{V}(I)$ is the set of common zeros (in $\mathbb{C}$) of polynomial functions that generate the ideal I. Given a variety $\mathcal{V} = \mathbb{V}(f_1, \ldots, f_m)$ at which polynomials $f_1, \ldots, f_m$ vanish, then by the definition of an ideal, any polynomial of the form $\Sigma_{i=1}^m g_i f_i$ with $g_i \in \mathbb{R}[x]$ will also vanishes in $\mathcal{V}$. Specifically, if ideals $I_1 = \langle f_1, \ldots, f_m \rangle$ and $I_2 = \langle g_1, \ldots, g_k \rangle$ are the same (i.e. they are generated by the same basis), then I_1 and I_2 also define the same variety, i.e. $\mathbb{V}(I_1) = \mathbb{V}(I_2)$. This illustrates that the varieties of an ideal are not affected by the choice of basis used for its representation [25].

The preceeding discussions suggest that the problem of solving a system of polynomial equations $f_i = 0, \ i = 1, \ldots, n$ is equivalent to that of computing the variety $\mathbb{V}(I)$ of the ideal $I = \langle f_i \rangle$ generated by f_i's. In other words, the solutions of polynomial equations $f_i = 0$ form the variety of the basis polynomials that generate the ideal $I = \langle f_i \rangle$. This implies that, even if a set of polynomial equations is too com-

plicated to solve, the solutions of polynomial equations may easily computed from the basis of the ideal generated by the polynomials. This fact is one of the basic approaches used in algebraic geometry for solving systems of polynomial equations. In particular, this approach can always be used since Hilbert's Basis Theorem [25] guarantees the existence of a finite basis for any ideal. A basis set that is useful for this approach is the Gröbner basis [14] which can be computed using computer algebra programs [32, 61]. Before we dwell further on the concept of a Gröbner basis, we will first discuss the nullstellensatz and the positivstellensatz concepts that can be used to verify the existence or emptiness of algebraic varieties of an ideal.

2.2.2 Existence of Algebraic Varieties

One method to verify the existence (or emptiness) of algebraic varieties of polynomials over $\mathbb{C}$ is using the result from the Hilbert's *nullstellensatz* stated below.

Theorem 2.2.1 (Nullstellensatz [25]). *Let $I \subset \mathbb{C}[x]$ be an ideal generated by a finite family of polynomials $(f_i)_{j=1,\ldots,m}$ in $\mathbb{C}[x]$. The following statements are equivalent:*

1. *The set*
$$\{x \in \mathbb{C}^n \mid f_i(x, k) = 0, \quad i = 1, \ldots, m\} \tag{2.2}$$
   is empty.
2. *The polynomial 1 belongs to the ideal I, i.e., $1 \in I$.*
3. *The ideal is equal to the whole polynomial ring: $I = \mathbb{C}(k)[x]$.*
4. *There exist polynomials $g_i(x, k) \in \mathbb{C}[x]$ such that:*
$$f_1(x, k)g_1(x, k) + \cdots + f_m(x, k)g_m(x, k) = 1. \tag{2.3}$$

The nullstellensatz means that if for an ideal $I = \langle f_i(x, k) \rangle$ there exist polynomials $g_i(x, k)$'s such that (2.3) is satisfied, then the complex solutions of polynomial equation $f_i(x, k) = 0$ do *not* exist (i.e. the set (2.2) is empty). Thus, the existence of polynomials $g_i(x, k)$ *certifies* the infeasibility of complex solutions of a system of polynomial equations. Example 2 illustrates an application of Theorem 2.2.1.

Example 2. *Let $I = \langle f_1, f_2 \rangle \subset \mathbb{C}[x]$ with $x = (x_1, x_2)$, $f_1(x) = x_1^2$, $f_2(x) = 1 - x_1 x_2$. Note that $\mathbb{V}(f_1, f_2) = \emptyset$ since there exist no $x \in \mathbb{C}^2$ that satisfies $f_1(x) = f_2(x) = 0$. By the nullstellensatz, there exist $g_1(x), g_2(x) \in \mathbb{C}^2[x]$ such that $f_1 g_1 + f_2 g_2 = 1$. One choice is given by $g_1(x) = 1 - x_1 x_2 + x_2^2$ and $g_2(x) = 1 + x_1 x_2 - x_1^2$.*

A generalization of the nullstellensatz concept that can be used to verify or prove the existence (or emptiness) of solutions over the reals, $\mathbb{R}$ is the *positivstellensatz* [137]. In essence, the positivstellensatz gives a sufficient condition for the infeasibility of *real* solutions to a system of polynomial equalities and inequalities. In order to state the positivstellensatz theorem, let us first recall the definitions of a *monoid* and a *cone* as given in [103].

Consider polynomials $f_i(x, k) \in \mathbb{R}[x]$ for $i = 1, \ldots, m$. The *multiplicative monoid* $\mathcal{M}(f_i)$ generated by f_i's is the set of finite products of the elements of f_i (including the identity and the empty product). A *cone* $\mathcal{C}$ of $\mathbb{R}[x]$ is a subset of $\mathbb{R}[x]$ such that (i) $a, b \in \mathcal{C} \Rightarrow a + b \in \mathcal{C}$, (ii) $a, b \in \mathcal{C} \Rightarrow a \cdot b \in \mathcal{C}$, and (iii) $a \in \mathbb{R}[x] \Rightarrow a^2 \in \mathcal{C}$. Given a set $S \subseteq \mathbb{R}[x]$, let $\mathcal{C}(S)$ be the smallest cone of $\mathbb{R}[x]$ that contains S. It can be shown that $\mathcal{C}(\emptyset)$ is the smallest cone in $\mathbb{R}[x]$ and can be expressed as a sum of squares [103]. The cone associated with a finite set $S = \{a_1, \ldots, a_m\} \subseteq \mathbb{R}[x]$ can be expressed as

$$\mathcal{C}(S) = \{f + \Sigma_{i=1}^r g_i b_i \mid f, g_1, \ldots, g_r \in \mathcal{C}(\emptyset), \text{ and } b_1, \ldots, b_r \in \mathcal{M}(a_i)\}.$$

The following theorem is due to Stengle [137].

Theorem 2.2.2 (Positivstellensatz, [137])**.** *Let $\{f_i\}_{i=(1,\ldots,s)}$, $\{g_j\}_{j=(1,\ldots,t)}$ and $\{h_\ell\}_{\ell=(1,\ldots,u)}$ be finite families of polynomials in $\mathbb{R}[x]$. Denote by $\mathcal{C}$ the cone generated by $\{f_i\}_{i=(1,\ldots,s)}$, $\mathcal{M}$ the multiplicative monoid generated by $\{g_j\}_{j=(1,\ldots,t)}$, and I the ideal generated by $\{h_\ell\}_{\ell=(1,\ldots,u)}$. Then, the following properties are equivalent*

- *The set*
$$\{x \in \mathbb{R}^n \mid f_i(x, k) \geq 0, \; g_j(x, k) \neq 0, \; h_\ell(x, k) = 0\} \tag{2.4}$$

is empty for $i = (1, \ldots, s)$, $j = (1, \ldots, t)$ and $\ell = (1, \ldots, u)$.

- *There exist $f \in \mathcal{C}$, $g \in \mathcal{M}$, and $h \in I$ such that $f + g^2 + h = 0$.*

Theorem 2.2.2 gives a sufficient condition for infeasibility/emptiness of real solutions to a system of polynomial equalities/inequalities. Such an infeasibility is certified by the existence of the *positivstellensatz refutation* in the forms of polynomials f, g and h. Example 3 illustrates an application of Theorem 2.2.2.

Example 3. *Consider the problem of checking the emptiness of the following set.*

$$\{x \in \mathbb{R} : x \leq -2, \, x \neq 0, \, x^2 - x = 2\} \tag{2.5}$$

The above set satisfies the expression in equation (2.4) with $i = j = \ell = 1$ and $f_1(x) = -x - 2$, $g_1(x) = x$ and $h_1(x) = x^2 - x - 2$. Now consider the following polynomials $f = f_1 \in \mathcal{C}(f_1)$, $g = g_1 \in \mathcal{C}(g_1)$ and $h = -h_1 \in \mathcal{I}(h_1)$. We then have $f + g^2 + h = (-x - 2) + (x^2) + (-x^2 + x + 2) = 0$. By Theorem 2.2.2, we conclude that the set in (2.5) is empty. Note from the equality in (2.5) that the solutions of equation $x^2 - x - 2 = 0$ are $x = 2$ or $x = -1$. The inequality $x \leq -2$ in (2.5), however, excludes these solutions and so the set in (2.5) will always be empty.

2.2.3 The Method of Gröbner Basis

This section discusses a technique for solving a system of polynomial equations using the method of Gröbner basis. The discussion uses the Buchberger's algorithm [14] to compute a Gröbner basis of an ideal.

2.2.3.1 Buchberger's Algorithm

Given an ideal $I \subseteq \mathbb{K}[x]$ with a finite set of generators, the Buchberger's algorithm takes the generators of I as an input and returns a Gröbner basis G of I as an output in a finite number of steps [14]. At the heart of Buchberger's algorithm is

the *division algorithm* [25, 14] which generalizes the concept of Euclidean division algorithm for canceling out the high order leading monomials of a polynomial to get another polynomial with lower order leading monomial. To discuss the basic idea in Buchberger's algorithm, the division algorithm and $S-$polynomial will be needed.

Let $f(x)$ be a polynomial and let $S = \{s_1(x), s_2(x), \ldots, s_q(x)\}$ be a list of polynomial divisors. For a fixed monomial ordering $>$, a division algorithm is one that finds polynomials $\lambda_i(x) \subseteq \mathbb{R}[x]$ and a remainder term $\overline{f(x)}^S$ which satisfy

$$f(x) = \sum_{i=1}^{q} \lambda_i(x)s_i(x) + \overline{f(x)}^S, \tag{2.6}$$

and such that

- $\texttt{LT}(\overline{f(x)}^S)$ is not divisible by any $\texttt{LT}(s_i(x))$,

- $\texttt{LT}(\overline{f(x)}^S) < \texttt{LT}(f(x))$,

- $\texttt{LT}(\lambda_i(x)s_i(x)) < \texttt{LT}(f(x))$.

Now let α and β be multi-indices. The *least common multiple* ($\texttt{LCM}$) of monomials x^α and x^β is

$$\texttt{LCM}(x^\alpha, x^\beta) = x_1^{\max(\alpha_1, \beta_1)} x_2^{\max(\alpha_2, \beta_2)} \ldots x_n^{\max(\alpha_n, \beta_n)}.$$

The $S-$polynomial of a pair of polynomials f_1 and f_2 is defined as

$$S(f_1, f_2) = \frac{x^\gamma}{\texttt{LT}(f_1)} f_1 - \frac{x^\gamma}{\texttt{LT}(f_2)} f_2, \tag{2.7}$$

where $x^\gamma = \texttt{LCM}(\texttt{LM}(f_1), \texttt{LM}(f_2))$. Based on the division algorithm and the $S-$polynomial defined above, a Gröbner basis of an ideal I is formally defined as follows [14].

Theorem 2.2.3 ([14])**.** *Let $I \subseteq \mathbb{K}[x]$ be an ideal with basis $G = \{g_1, \ldots, g_q\}$. Then G is a Gröbner basis for I if and only if the remainder on division of every $S-$polynomial $S(g_i, g_j)$, $i \neq j$, by G is zero.*

26

Theorem 2.2.3 was the basis of Buchberger's algorithm [14] for computing a Gröbner basis of an ideal. This algorithm essentially computes the $S-$polynomial of each ideal's generator pair and then tests whether the remainder upon division of the generator by each of the $S-$polynomial is zero or not. If there exist non-zero remainders, the original set of generators is extended by adding those non-zero remainders and then the iteration is repeated over the new, extended generator. A Gröbner basis is given by those extended generators whose $S-$polynomials divide the elements of the extended generator with zero remainder. As was proven in [14], Buchberger's algorithm always terminates in a finite number of steps and so a Gröbner basis of an ideal can always be computed. Example 4 illustrates the basic idea in using Buchberger's algorithm to compute the variety (or zeros) of an ideal.

Example 4. *Consider an ideal $I = \langle f_1, f_2 \rangle$ generated by two polynomials $f_1 = 2x_1^2 - 4x_1 + x_2^2 - 4x_2 + 3$ and $f_2 = x_1^2 - 2x_1 + 3x_2^2 - 12x_2 + 9$. We will apply Buchberger's algorithm to compute a Gröbner basis of I. To begin, we use the generator of I as the initial Gröbner basis $G_0 = \{f_1, f_2\}$ and consider the lex order $x >_{lex} y$ for the monomials of I. Note that the ideal I is characterized by $LM(f_1) = 2x^2$ and $LM(f_2) = x^2$ such that $\gamma = (2, 0)$. The $S-$polynomial of f_1, f_2 is then given by*

$$S(f_1, f_2) = \frac{x_1^2}{2x_1^2} f_1 - \frac{x_1^2}{x_1^2} f_2 = -\frac{5}{2} x_2^2 + 10x_2 - \frac{15}{2}.$$

Upon division of $S(f_1, f_2)$ by G_0, the remainder r_{12} is simply $S(f_1, f_2) \neq 0$. So we now define $f_3 = S(f_1, f_2)$ and extend the Gröbner basis G_0 to $G_1 = \{f_1, f_2, f_3\}$. We then repeat the computation of the $S-$polynomials for the new generator pairs (f_1, f_3) and (f_2, f_3). Note that we do not need to recompute the $S-$polynomial $S(f_1, f_2)$ as it

does not change from its value prior to extending the initial generator. Thus,

$$S(f_1, f_3) = 4x_1^2 x_2 - 3x_1^2 - 2x_1 x_2^2 + \frac{1}{2}x_2^4 - 2x_2^3 + \frac{3}{2}x_2^2,$$

$$S(f_2, f_3) = 4x_1^2 x_2 - 3x_1^2 - 2x_1 x_2^2 + 3x_2^4 - 12x_2^3 + 9x_2^2.$$

Dividing these S−polynomials by G_1 gives

$$S(f_1, f_3) = (2x_2 - 3/2)f_1 + (4x_1 - x_2^2 + 4x_2 - 3)f_3/5,$$

$$S(f_2, f_3) = (2x_2 - 3/2)f_1 + (4x_1 - 6x_2^2 + 4x_2 - 3)f_3/5,$$

from which we have $r_{13} = r_{23} = 0$. Thus, a Gröbner basis of I is $\mathbb{G} = \{f_1, f_2, f_3\}$.

Buchberger's algorithm has been implemented in many computer algebra programs such as SINGULAR [32] and Macaulay2 [61]. One issue in the implementation of Buchberger's algorithm is that it has doubly exponential worst case complexity in the number of unknown variables. As was shown in [39], for a given polynomial $f(x)$ in n variables with a total degree not exceeding d, then the degree of polynomials in Gröbner basis G is bounded by $2(d^2/2 + d)^{2^{n-1}}$. This bound is doubly exponential with respect to the number of unknowns n, but only polynomial with respect to the maximum degree d of the polynomial. The computation of Gröbner basis for high dimensional systems will therefore require a large amount of computer memory [103]. Nevertheless, the use Buchberger's algorithm for Gröbner basis computation in many applications has shown that such a worst case bound is not frequently encountered [25].

Note that the Gröbner basis of an ideal is not unique. For instance, another Gröbner basis of the ideal I in Example 4 is $\mathbb{G} = \{x_2^2 - 4x_2 + 3, x_1^2 - 2x_1 + 3x_2^2 - 12x_2 + 9\}$. This basis was computed in SINGULAR [32] using the code snippet in Listing 2.1. In this listing, the command `ring` declares a polynomial ring 'R' of unknown variables 'x1,x2' with real coefficients '0'. The monomial ordering in R

is set to be the lex ordering 'lp'. The command `poly` defines polynomial functions 'f1,f2' whereas the command `ideal` defines the ideal I. Finally, a Gröbner basis 'G' of ideal 'I' is computed using the command `groebner(I)`.

Listing 2.1: Computing Gröbner basis of Example 4 using SINGULAR[32].

```
> ring  R =  0 ,( x1 , x2 ) , lp ;
> poly  f1  =  2∗x1^2 −  4∗x1  +  x2^2 −  4∗x2  +  3;
> poly  f2  =  x1^2 −  2∗x1  +  3∗x2^2 −  12∗x2  +  9;
> ideal  I  =  ( f1 , f2 );
> ideal  G  =  groebner ( I );
> G;
G[1]=x2^2−4∗x2+3
G[2]=x1^2−2∗x1+3∗x2^2−12∗x2+9
```

2.2.3.2 Elimination and Extension Theorems

It should now be intuitive to see how the Gröbner basis can be used to solve a system of polynomial equations. In particular, Example 4 shows that one of the Gröbner basis's elements was $f_3 = S(f_1, f_2) = -5x_2^2/2 + 10x_2 - 15/2$. In this basis element, the variable x_1 was *eliminated*. Since f_3 is a univariate polynomial, it can be solved using numerical root finding methods [25]. If the solution x_2 is *extended* by substitutions into f_1 or f_2, the total solution to $f_i = 0$, $i = 1, 2$ can be obtained. It is then clear that the Gröbner basis method for solving a system of polynomial equations involves two main steps, namely the *elimination* and *extension* steps.

Given a Gröbner basis of an ideal I, the elimination step computes the rth *elimination ideal* of I defined below.

Definition 2 ([25]). *Given $I = \langle f_1, \ldots, f_q \rangle \subset \mathbb{R}[x_1, \ldots, x_n]$, the rth elimination ideal*

I_r is the ideal of $\mathbb{R}[x_{r+1}, \ldots, x_n]$ defined by

$$I_r = I \cap \mathbb{R}[x_{r+1}, \ldots, x_n].$$

It can be seen that the elimination ideal is obtained by eliminating subsets of variables of the original ideal. Given a Gröbner basis of an ideal I with lex monomial ordering, the Elimination Theorem can be used to compute a Gröbner basis for the rth elimination ideal of I.

Theorem 2.2.4 (Elimination Theorem [25]). *Let $I \subset \mathbb{R}[x_1, \ldots, x_n]$ be an ideal and let G be a Gröbner basis of I with respect to lexicographic order where $x_1 >_{lex} x_2 >_{lex} \cdots >_{lex} x_n$. Then, for every $0 \le r \le n$, the set*

$$\mathbb{G}_r = \mathbb{G} \cap \mathbb{R}[x_{r+1}, \ldots, x_n],$$

is a Gröbner basis of the rth elimination ideal I_r.

Note that the variety $\mathbb{V}(I_r)$ of the rth elimination ideal I_r can be obtained by computing the zeros of its Gröbner basis, $\mathbb{G}_r$. The variety $\mathbb{V}(I_r)$, however, is only a subvariety of the original ideal I (i.e. $\mathbb{V}(I_r) \subset \mathbb{V}(I)$) because it is defined only for variables that are not eliminated from I. Thus, it only serves as a partial solution and should be extended to the original ideal to get the total solution. Such an extension, however, is not always guaranteed to be feasible. Equation (2.8) in the Extension Theorem stated below gives the condition when such an extension is feasible.

Theorem 2.2.5 (Extension Theorem [25]). *Let $I = \langle f_1, \ldots, f_m \rangle \subset \mathbb{R}[x]$ and let I_1 be the first elimination ideal of I. For each $1 \le i \le m$, write f_i in the form*

$$f_i = g_i(x_2, \ldots, x_n)x_1^{N_i} + \ \text{terms in which } x_1 \text{ has degree} < N_i, \tag{2.8}$$

where $N_i > 0$ and $g_i \in \mathbb{R}[x]$ is nonzero. Assume we have a partial solution $(x_2^*, \ldots, x_n^*) \in$

$\mathbb{V}(I_1)$. If $(x_2^*, \ldots, x_n^*) \notin \mathbb{V}(g_1, \ldots, g_m)$, then there exists $x_1^* \in \mathbb{C}$ such that $(x_1^*, \ldots, x_n^*) \in \mathbb{V}(I)$.

The above discussions suggest that the goal of the elimination step is to iteratively reduce the original problem into problems with a smaller number of variables for which partial solutions can be computed. The extension step then back substitutes these partial solutions into the original ideal to get the total solution. The importance of the Gröbner basis is that it allows for a systematic execution of the elimination step.

Example 5 illustrates the Elimination and the Extension Theorems.

Example 5. *Consider the ideal and Gröbner basis from Example 4. We will show how the Gröbner basis, combined with the elimination and extension steps, is used to compute the varieties of an ideal $I = \langle f_1, f_2 \rangle \subseteq \mathbb{R}[x_1, x_2]$. First, we let $I_2 = \langle I \cap \mathbb{R}[x_2] \rangle$ which gives $I_2 = 0$. This means that any value of $x_2 = c_2$ is a partial solution and $c_2 \in \mathbb{R}$. We now evaluate whether this partial solution extends to the total solution $x_1 = c_1, x_2 = c_2$.*

The application of Extension Theorem to the case of $\mathbb{R}[x_2] \subset \mathbb{R}[x_1, x_2]$ suggests that the partial solution $x_2 = c_2$ extends to $x_1 = c_1, x_2 = c_2$, provided that the coefficients of the highest power of x_1 in the Gröbner basis f_1 or f_2 are not zero at $x_2 = c_2$. The coefficients of the highest power of x_1 in f_1 and f_2 are given by 1 and 2, respectively, which are not zero regardless of the value of x_2. This implies that the solution $x_2 = c_2$ always extends to the total solution $x_1 = c_1, x_2 = c_2$. From the Gröbner basis of I_2, we have that

$$\mathbb{G}(I_2) = \mathbb{G} \cap \mathbb{R}[x_2] = -\frac{5}{2}x_2^2 + 10x_2 - \frac{15}{2}.$$

Since the variety of an ideal is equivalent to the variety of its Gröbner basis, the partial solution x_2 is then given by the zeros of $\mathbb{G}(I_2)$. These are $x_2^* = 1$ or $x_2^* = 3$. We have
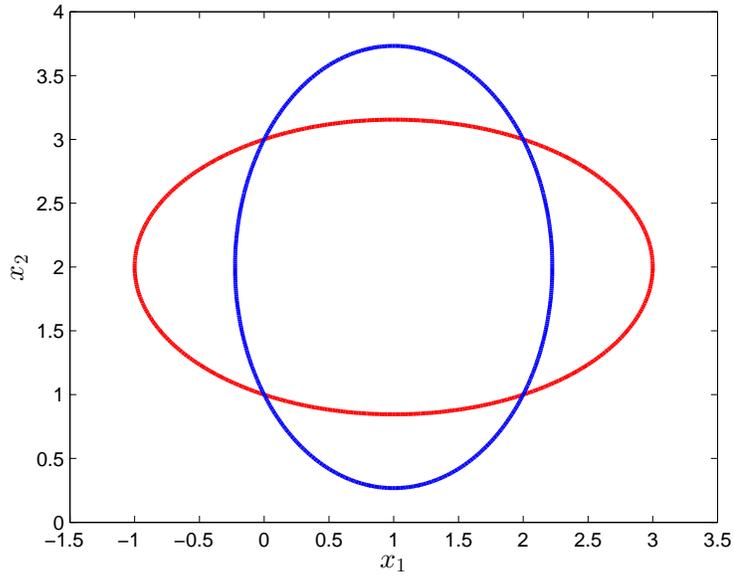
Figure 2.1. Algebraic variety of $I = \langle f_1, f_2 \rangle$ in Example 5.

shown previously that this partial solution extends to the total solution. Thus, we can directly substitute x_2^* into either f_1 or f_2 and then solve for their corresponding zeros. Upon substitution and solving for the zeros, the variety of $I = \langle f_1, f_2 \rangle$ is given by

$$\mathbb{V}(I) = \{(x_1^*, x_2^*) \; : \; (0,1), \; (0,3), \; (2,1), \; (2,3)\},$$

which are the four intersection points between ellipses f_1 and f_2 shown in Figure 2.1.

2.3 Polynomial Optimization

A fundamental problem that arises in many applications is that of proving the nonnegativity of some functions in several variables. Specifically, given a function $p(x)$ in variables $x = (x_1, \ldots, x_n)$, one is often required to check the validity of the proposition $p(x) \geq 0$ for all $x \in \mathcal{S}$ in which $\mathcal{S}$ is a *semialgebraic set*. A semial-

gebraic set is a set defined by functional equalities and/or inequalities of the form $F_i(x) = 0$, $G_j(x) \geq 0$ for $i = 1, \ldots, N_F$ and $j = 1, \ldots, N_G$ with $N_F, N_G \in \mathbb{Z}_+$. From a computational standpoint, this problem is an NP-hard (Non-deterministic Polynomial-time hard) problem. This means that the validity of the proposition generally cannot be decided in a polynomial time manner [119]. However, if the functions $p(x)$, $F_i(x)$ and $G_j(x)$ are *multivariate polynomials*, a checkable condition for proving the validity of such a proposition can be obtained using methods for representing nonnegative polynomials over semialgebraic set [119, 66]. This section will focus on such a case where $p(x)$, $F_i(x)$ and $G_j(x)$ are polynomial functions.

Let $\mathbb{R}[x]$ be the ring of polynomials in n unknowns $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$ with real coefficients $k \in \mathbb{R}$. Consider a *polynomial optimization problem* (POP)

$$\min \ p(x, k), \quad \text{such that} \ \ x \in \mathcal{S} := \{x \in \mathbb{R}^n : F_i(x, k) = 0, \ G_j(x, k) \geq 0\}, \ k \in \mathbb{R} \tag{2.9}$$

where $p(x, k) \in \mathbb{R}[x]$ and $F_i(x, k)$, $G_j(x, k) \subseteq \mathbb{R}[x]$ are vectors of polynomial functions that define a semialgebraic set $\mathcal{S}$ and $i = 1, \ldots, N_F$, $j = 1, \ldots, N_G$ with $N_F, N_G \in \mathbb{Z}_+$. As mentioned previously, the exact solutions of the POP (2.9) are generally difficult to obtain. However, recent advances in real algebraic gemetry and convex optimization techniques have developed methods to estimate a sequence of converging lower bounds for the minimum of $p(x, k)$ in 2.9 [119, 118, 95, 93, 9]. These methods generally use the theory of positive polynomials over the reals (*positivstellensatz*) [137, 66, 25, see also Section 2.2.2] to formulate hierarchies of semidefinite or linear programming relaxation problems that correspond to the POP (2.9) [119, 118]. In particular, such a *relaxation of polynomial optimization problem* (RPOP) takes the form

$$\gamma^* := \max \ \gamma, \quad \text{such that} \ \ f(x, k) \in \mathcal{K}, \tag{2.10}$$

where $f(x, k) = p(x, k) - \gamma \in \mathcal{K}$ with $\gamma \geq 0$ is a constant and $\mathcal{K}$ is a *cone* of

polynomials that is nonnegative in $\mathcal{S}$. It is clear that $f(x, k) \in \mathcal{K}$ implies $p(x, k) \geq \gamma$ and so the solution γ^* of the RPOP in (2.10) gives a lower bound for the POP (2.9).

There are two types of characterizations that are often used to represent the nonnegative cone $\mathcal{K}$. The first one seeks to represent $\mathcal{K}$ as a sum of squares polynomial (SOS relaxation) while the second one aims to represent $\mathcal{K}$ in term of Handelman basis polynomial (Handelman relaxation) [66]. The key important aspect in both relaxations is that their formulations are convex optimization problems: the search of SOS and Handelman relaxations of $\mathcal{K}$ is a semidefinite programming (SDP) problem and a linear programming (LP) problem, respectively [94]. This implies that the RPOP (2.10) is a computationally tractable method for solving the POP (2.9).

This section presents a brief introduction to the SOS relaxation of the RPOP (2.10) for solving the POP in (2.9). The SOS relaxation technique discussed in this section is the main computational approach used to forecast the onset of regime shifts (Chapters 4 - 5).

Remark 2.3.1. Readers who are interested in the Handelmann relaxation of the RPOP (2.10) may refer to Appendix A. This appendix also shows a comparison between the SOS and the Handelman relaxations obtained from numerical simulations.

2.3.1 SOS Decomposition

A polynomial $g(x, k) \in \mathbb{R}[x]$ is said to be *nonnegative* or *positive semidefinite* (psd) if $g(x, k) \geq 0$ for all $x \in \mathbb{R}^n$. A necessary condition which guarantees a polynomial to be psd is that its total degree is even [117]. A class of polynomial systems for which this condition is always satisfied is the SOS polynomials.

Definition 3 ([118, SOS polynomial]). *We say that a polynomial $g(x, k)$ is SOS if it can be rewritten as $g(x, k) = \sum_{i=1}^{\ell} q_i^2(x)$ for some set ℓ of polynomials $q_i(x), i = 1, \ldots, \ell$.*

Clearly, if polynomial $g(x, k)$ is SOS then it is also psd. In what follows, we use $\mathcal{P}_n^{2d}$ to denote the set of SOS polynomials in n unknown variables with degree less than or equal to $2d$.

Consider a polynomial $g(x, k) = \sum_{|\alpha| \leq 2d} k_\alpha x^\alpha \in \mathbb{R}(k)[x]$ and $g \in \mathcal{P}_n^{2d}$. Note that the number of coefficients of $g(x, k)$ is $\binom{n+2d}{2d}$. Let

$$[x]_d \doteq [1, x_1, \ldots, x_n, x_1^2, x_1 x_2, \ldots, x_n^d]^T,$$

be the vector of all $\binom{n+d}{d}$ monomials in x with degrees less than or equal to d. Let Q be an $\binom{n+d}{d} \times \binom{n+d}{d}$ symmetric matrix and consider the equation

$$g(x, k) = [x]_d^T Q [x]_d. \tag{2.11}$$

Theorem 2.3.2 formally states that a polynomial $g(x, k)$ is SOS if it can be decomposed as in (2.11).

Theorem 2.3.2 ([117, SOS decomposition]). *A polynomial $g(x, k) = \sum_{|\alpha| \leq 2d} k_\alpha x^\alpha$ of degree $2d$ in n variables is SOS if and only if there exists $Q \in \mathcal{S}_{\geq 0}^{\binom{n+d}{d}}$ such that (2.11) holds.*

The symmetric psd matrix Q in Theorem 2.3.2 is also called a Gramian matrix and can be factored as $Q = V^T V$. This implies that (2.11) can be rewritten in a form

$$g(x, k) = [x]_d^T Q [x]_d = [x]_d^T V^T V [x]_d = (V[x]_d)^T (V[x]_d),$$

that satisfies the condition in Definition 3. Thus, any polynomial function that can be decomposed as in (2.11) is always a psd function.

Now consider the decomposition in (2.11) and let us index the elements of matrix Q in such a decomposition by the $\binom{n+d}{d}$ monomials in $[x]_d$. By comparing the monomials of equivalent degrees on the left and right hand sides of the equation, then the

35

following relationship between the coefficients of $g(x, k)$ and the elements of Q can be obtained.

$$k_\alpha = \sum_{i+j=\alpha} Q_{ij}, \quad Q \succeq 0. \tag{2.12}$$

Equation (2.12) is a system of $\binom{n+2d}{2d}$ linear equations relating the entries of matrix Q and the coefficients k_α of $g(x, k)$. This relationship shows that finding a polynomial's SOS decomposition of the form (2.11) is equivalent with searching for a psd matrix Q for which equation (2.12) holds. This is a well established problem which can be posed as an SDP problem. Consequently, this suggests that the search of an SOS representation of a polynomial function can be formulated as an SDP problem. This fundamental equivalence was a result proved in [117, 118].

Theorem 2.3.3 ([117]). *The existence of SOS decomposition of a polynomial in n variables of degree $2d$ can be decided by solving an SDP feasibility problem.*

Note that the connection between the feasibility of the SOS decomposition (2.11) and its SDP formulation arises from the simultaneous requirements for matrix Q to satisfy both the positive definiteness condition and the linear equalities in (2.12). This shows that such requirements can be formulated as an SDP problem.

2.3.1.1 SOS Relaxation of POP

Theorem 2.3.3 is the basis for constructing an SOS relaxation of the form (2.10). In particular, for a semialgebraic set $\mathcal{S}$ in (2.9), consider the following cone of polynomial $\mathcal{K}_{sos}$ that is positive over $\mathcal{S}$ [117]

$$\mathcal{K}_{sos} = \left\{ f(x) \in \mathbb{R}[x] : f(x) - \sum_{j=1}^{N_G} q_j(x) G_j(x) - \sum_{i=1}^{N_F} r_i(x) F_i(x) \quad \text{is SOS} \right\}, \tag{2.13}$$

where $r_i(x) \in \mathbb{R}[x]$, $q_j(x) \in \mathcal{P}_n^{2d}$. The SOS relaxation of the form (2.10) is given by

$$
\begin{aligned}
\gamma^* := &\max_{\gamma, r_i, q_j} \gamma \\
\text{s.t.} \quad &p(x) - \gamma - \sum_{j=1}^{N_G} q_j(x) G_j(x) - \sum_{i=1}^{N_F} r_i(x) F_i(x) \quad \text{is SOS.}
\end{aligned}
\tag{2.14}
$$

Now for any $\gamma > 0$, the equivalence in (2.12) implies that the optimization problem (2.14) can be solved iteratively by testing the feasibility of solutions in the following SDP problem

$$
\begin{aligned}
\text{Find} \quad &Q_0, \, Q_j, \, R_i \\
\text{s.t.} \quad &p(x) - \gamma = [x]_d^T \left(Q_0 + \sum_{j=1}^{N_G} Q_j(x) G_j(x) + \sum_{i=1}^{N_F} R_i(x) F_i(x) \right) [x]_d,
\end{aligned}
\tag{2.15}
$$

where $Q_0, Q_j(x), R_i(x) \in \mathcal{S}_{\geq 0}^N$ with $N = \binom{n+d}{d}$ and $[x]_d$ denotes the vector of monomials in x of degree less than or equal to d.

There are various software tools that can be used to solve the POP (2.9) using the SDP relaxation (2.15). These tools include SOSTOOLS [121], GloptiPoly [69], and YALMIP [100]. In general, these tools are parser codes which transform the POP (2.9) into an SDP formulation (2.15) and then use available SDP solvers such as Sedumi [139] or Mosek [111] to obtain the optimal solution.

2.4 Example

This section illustrates an application of the Gröbner basis and SOS relaxation methods described in the previous sections. We use these methods to solve the voltage collapse problem in power system.

This example considers a single machine, two-bus generator-line-load model of a power network that was used in [15] to study the voltage collapse problem. The ODE

model governing the system is given by [15]

$$\dot{\omega} = \frac{1}{M} \left[P_m - P_{e1}(\delta, V) - D_G \omega \right],$$

$$\dot{\delta} = \omega - \frac{1}{D_L} \left[P_{e2}(\delta, V) - P_d \right], \qquad (2.16)$$

$$\dot{V} = \frac{1}{\tau} \left[Q_e(\delta, V) - Q_d \right],$$

where

$$\delta = \delta_1 - \delta_2,$$

$$P_{e1} = G - V(G \cos \delta - B \sin \delta),$$

$$P_{e2} = -V^2 G + V(G \cos \delta + B \sin \delta),$$

$$Q_e = -V^2 (B - B_c) - V (G \sin \delta - B \cos \delta),$$

$$G = \frac{R}{R^2 + X_L^2}, \qquad B = \frac{X_L}{R^2 + X_L^2}.$$

Equation (2.16) is a lumped model of power network consisting three main parts namely (i) the generator part (represented as a second order mechanical system), (ii) the transmission line and (iii) the load part (active and reactive load powers) (cf. Figure 2.2). One important subject in power system analysis is the voltaga collapse phenomenon that occurs due to variations in the active (P_d) and reactive (Q_d) load powers. The collapse usually corresponds to voltage instability that occurs when the system's parameters, P_d and Q_d, are varied. This instability is characterized by the singularity of the system's Jacobian matrix at some critical parameters P_d^* and Q_d^*. One approach that has been used to predict the onset of a voltage collapse is by computing a quantity called the *minimum distance to instability* [15]. This quantity measures the shortest distance from a nominal set of parameters k^0 to the set of critical parameters k^* at which such an instability occurs.

We will use the methods described in the previous sections to bound the minimum distance to instability. The set of parameters considered in this case is $k = (P_d, Q_d)$ (i.e. the active and reactive load powers). In particular, the minimum distance to
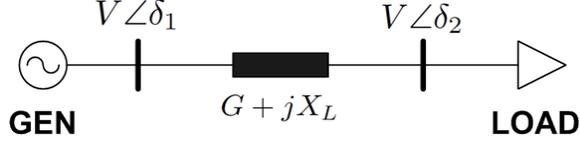
Figure 2.2. A single machine power network [15].

instability is defined as $\gamma = \inf_k |k^* - k^0|$, where k^* is the set of critical parameters

at which the system's Jacobian matrix becomes singular [15].

In this analysis, we consider a nominal equilibrium in which $\omega = 0$ and choose

$R = 0.1, X_L = 0.5$ for the transmission line's parameters [15]. Using this nominal

equilibrium, we are interested in characterizing the critical parameter $k^* = (P_d^*, Q_d^*)$

at which a voltage collapse occurs. The methods described in the previous sections

will be used for this purpose.

We first transform the vector fields in (2.16) into polynomial functions. To do

this, we introduce new variables $x = \sin \delta$ and $y = \cos \delta$. Using these new variables, it

can be shown that the equilibria of (2.16) are defined by polynomial equations below

$$0 = -V^2 G + V G y + V B x - P_d,$$
$$0 = -V^2 B - V G x + V B y - Q_d, \quad (2.17)$$
$$0 = x^2 + y^2 - 1.$$

In addition, it can be shown that the condition for the system's Jacobian matrix to

be singular (i.e. has zero determinant) is given by

$$0 = B^2 + G^2 - 2B^2 V y - 2G^2 V y. \quad (2.18)$$

Thus, the computation of the minimum distance to instability can be formulated as

the following polynomial optimization problem (POP)

$$
\begin{aligned}
\text{minimize:} \quad & (P_d^* - P_d^0)^2 + (Q_d^* - Q_d^0)^2 \\
\text{such that:} \quad & 0 = -V^2 G + V G y + V B x - P_d, \\
& 0 = -V^2 B - V G x + V B y - Q_d, \\
& 0 = x^2 + y^2 - 1, \\
& 0 = B^2 + G^2 - 2B^2 V y - 2G^2 V y, \\
& P_d \geq 0, \ Q_d \geq 0.
\end{aligned}
\tag{2.19}
$$

Note that the first four constraints in POP (2.19) define a set of polynomial equations. These constraints can be simplified if there exists a Gröbner basis of the ideal I generated by the polynomials in the constraints. In fact, when we computed a Gröbner basis for these polynomials, we found a Gröbner basis $\mathcal{G}$ below (see Listing B.1 in Appendix B for SINGULAR code)

$$
\begin{aligned}
\mathcal{G} = \{ B^2 \left(B\left(B - 4Q_d\right) + 2\left(G^2 - 2GP_d - 2P_d^2\right) \right) \\
+ G^2 \left(G(G - 4P_d) - 4Q_d(B + Q_d)\right) + 8BGP_dQ_d \}.
\end{aligned}
\tag{2.20}
$$

Note that the Gröbner basis $\mathcal{G}$ has only single element. In particular, for given values of parameters B and G, the element of $\mathcal{G}$ is a polynomial function in variables P_d and Q_d only. The Gröbner basis $\mathcal{G}$ thus defines a manifold in the parameter space where an instability occurs.

Using the element of $\mathcal{G}$ and the SOS relaxation method described in the previous sections, a lower bound on the minimum distance to instability can be computed using the following SOS optimization

$$
\begin{aligned}
\text{max:} \quad & \gamma \\
\text{s.t.:} \quad & (P_d^* - P_d^0)^2 + (Q_d^* - Q_d^0)^2 - \gamma + \sum_i r_i(\mu)\mathcal{G}_i, \ i = 1, \ldots, m, \quad \text{is SOS,}
\end{aligned}
\tag{2.21}
$$
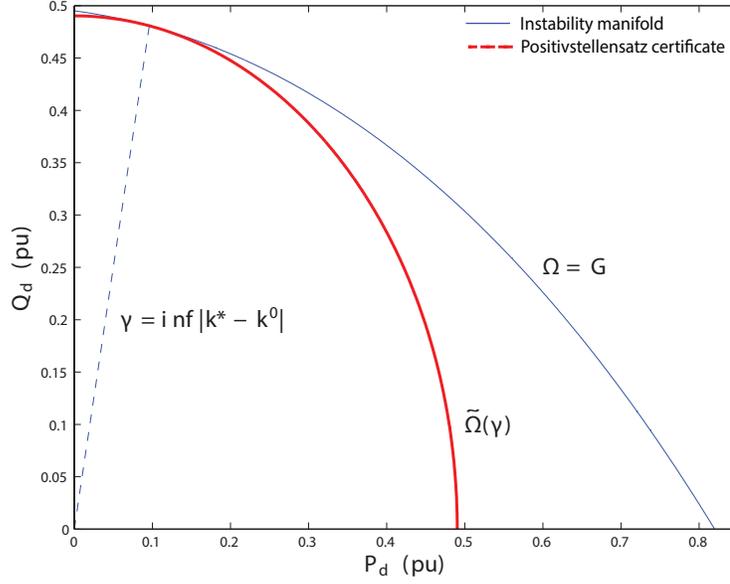
Figure 2.3. Instability manifold and positivstellensatz certificate.

where m denotes the number of elements of $\mathcal{G}$. In this example, $m = 1$.

We consider a nominal set of parameters $k^0 = (0, 0)$. We used SOSTOOLS [118] to solve (2.21) (see Listing B.2 in Appendix B for MATLAB code). We found a minimum $\gamma = 0.2404$ which corresponds to critical parameters $k^* = (0.0961, 0.4808)$. These are the same as that obtained in [15], but our formulation is much simpler as it only uses a single equality constraint. Figure 2.3 plots the curve of the Gröbner basis' element which defines the manifold where instability occurs. The curve $\tilde{\Omega}(\gamma)$ plotted in this figure is the positivstellensatz certificate which verifies that γ is the shortest distance from the nominal parameter set k^0 to the manifold defined by $\mathcal{G}$.

CHAPTER 3

CHARACTERIZATION OF EQUILIBRIA IN NONNEGATIVE SYSTEMS WITH
KINETIC REALIZATIONS

3.1   Introduction

This chapter presents a method to characterize the equilibria of nonnegative sys-
tems with kinetic realizations. A dynamical system $\dot{x}_i(t) = f_i(x, k)$, $i = 1, \ldots, n$,
with polynomial vector fields $f_i(x, k)$ is said to have a kinetic realization if there ex-
ist polynomials $g_i(x, k)$ and $h_i(x, k)$ with nonnegative coefficients such that $f_i(x, k)$
can be written as $f_i(x, k) = g_i(x, k) - x_i h_i(x, k)$. One important property of systems
with kinetic realizations is that their special structure allows one to compute expres-
sions for the system's equilibria as a rational function of the system's parameters and
some convex parameters [57, 58]. This chapter discusses method to compute such
expressions of the system's equilibria.

Recall from Chapter 2 that bifurcation-induced regime shifts in a parameterized
dynamical system

$$\dot{x}(t) = f(x, k), \qquad x(0) = x_0 \text{ and } f(x, k) \in \mathbb{R}(k)[x] \tag{3.1}$$

will be characterized by a quantity $\gamma := \inf |k^* - k^0|^2$ called the minimum D_2B.
The minimum D_2B essentially measures the shortest distance from a set of nominal
parameters k^0 to the closest set of critical parameters k^* at which the system's qual-
itative properties (i.e. phase portrait, stability of the equilibria) change. One way to
compute γ is by characterizing the parameters, k, at which the number or stability of

the nominal equilibrium $x^*(k^0)$ change. This can be done by analyzing the stability (i.e. eigenvalues) of the system's Jacobian matrix

$$J(x^*, k) = \left. \frac{\partial f(x, k)}{\partial x} \right|_{x^*} \tag{3.2}$$

at the equilibria $x^* := \{x : f(x, k) = 0\}$. Observe in equation (3.2) that the value of J can be obtained if the equilibria x^* is known.

Computing the algebraic expression of the equilibria x^* in high dimensional systems usually requires the use of symbolic computational methods. These methods are based on the fact that a set of polynomials generates an ideal in the polynomial ring and that the zeros of an ideal are equivalent to the zeros of any Gröbner basis of that ideal [25, 10, see also Chapter 2]. The value in using the Gröbner basis is that it allows one to formulate computational methods for systematically solving for the system's zeros in a manner that is very reminiscent of the back-substitution methods used for solving a system of linear algebraic equations. The main stumbling block is that the complexity of computing a Gröbner basis using Buchberger's algorithm is known to grow in a doubly exponential manner with respect to the number of the unknown variables [39].

This complexity can be reduced if the vector fields in (3.1) have a special structure. The special structure that we consider is the case when the vector fields have kinetic realizations. A dynamical system $\dot{x}_i(t) = f_i(x, k)$ with polynomial vector fields $f_i(x, k)$ is said to have a kinetic realization if there exist polynomials $g_i(x, k)$ and $h_i(x, k)$ with nonnegative coefficients such that $f_i(x, k)$ can be rewritten as [67]

$$f_i(x, k) = g_i(x, k) - x_i h_i(x, k). \tag{3.3}$$

The condition in equation (3.3) was presented in [67] and it essentially guarantees that a polynomial dynamical system is a *mass action systems* for which a chemical

43

reaction network (CRN) graph can be realized. In addition, the vector fields of these systems form a special class of ideals called toric ideals [57, 58]. One advantage in working with toric ideals is that there exist efficient algorithms (with lower complexity than the standard Buchberger's algorithm) for computing its Gröbner basis [140, 7]. Once such a Gröbner basis is obtained, the expression for the *toric variety* or the zeros of the corresponding toric ideal can then be computed [140, 7, 6, 27].

This chapter discusses a method to compute an expression of the equilibria x^* for system whose vector fields satisfy equation (3.3). In this thesis, these systems are called *nonnegative systems with kinetic realizations*. Dynamical systems with kinetic realizations originate from the differential equation models of CRNs [153, 22, 44]. One important property of systems with kinetic realizations is that their special structure allows one to compute expressions for the system's equilibria as a rational function of the system's parameters and some convex parameters. This computational method was first proposed in [57, 58] and is based on the concept of *toric variety* from algebraic geometry [6, 7]. The main advantage of this method is that it helps simplify the analysis of the system's qualitative behaviors (cf. Chapter 4).

The remainder of this chapter is structured as follows. Section 3.2 discusses the construction and properties of kinetic realizations. In particular, Section 3.2.1 discusses the construction of kinetic realizations for models of CRNs and nonnegative dynamical systems. Section 3.2.3 then discusses the properties of these kinetic realizations. Section 3.3 describes a method introduced in [57, 58] to compute the expressions of kinetic realizations' equilibria. An algorithmic implementation of this computational method is also presented. Conclusions are given in Section 3.4.

The discussion in this chapter will be central for the discussion of the method used to forecast bifurcation-induced regime shifts in Chapter 4. Specifically, the computation of kinetic realizations' equilibria is the starting point from which the simplification of the minimum D_2B problem is achieved. This chapter should there-

fore be viewed as an essential part of Chapter 4.

3.2 Kinetic Realizations

3.2.1 Kinetic Realizations of CRN

The dynamics of the reactant and the product species concentrations in a CRN can be modeled according to the law of *mass action kinetics.* This law states that the velocity or rate of each elementary reaction in the CRN is directly proportional to the product of the reactant concentrations [44]. Thus, this law can be used to derive differential equation models of a CRN.

Consider a set of r elementary reactions $\mathcal{R}_i$ $(i = 1, \ldots, r)$ between $s \geq 1$ chemical *species* $X_1, X_2, \ldots, X_s$ described in the following CRN

$$\mathcal{R}_i: \quad \sum_{j=1}^{s} a_{ij} X_j \xrightarrow{k_i} \sum_{j=1}^{s} b_{ij} X_j, \quad i = 1, 2, \ldots, r. \tag{3.4}$$
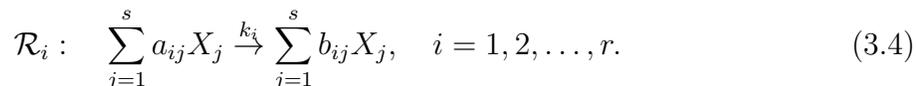
For each $i = 1, 2, \ldots, r$, the quantity k_i in (3.4) denotes the *rate constant* of the ith reaction $\mathcal{R}_i$ between the ith *reactant complex* $\mathcal{C}_i^R = \sum_{j=1}^{s} a_{ij} X_j$ (the sum of all species on the tail of the ith reaction arrow) and the ith *product complex* $\mathcal{C}_i^p = \sum_{j=1}^{s} b_{ij} X_j$ (the sum of all species on the head of the ith reaction arrow). Let $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_m\}$ be the union set of m complexes appear in both the heads and the tails of reaction arrows of CRN (3.4). Then for $i = 1, 2, \ldots, r$ and $p, q \in \ell = 1, 2, \ldots, m$ with $p \neq q$, the ith elementary reaction, $\mathcal{R}_i$, in CRN (3.4) can be represented using the elements of $\mathcal{C}$ as follows

$$\mathcal{R}_i: \quad \mathcal{C}_p \xrightarrow{k_i} \mathcal{C}_q. \tag{3.5}$$

In what follows, we use x_j to denote the concentration of species X_j at time t and $x = [x_1, \ldots, x_s]^T$ to denote the vector of species concentrations in CRN (3.4).

The association of all s chemical species to the set of m unique complexes in $\mathcal{C}$ defines a *bipartite graph* from X_i $(i = 1, \ldots, s)$ to $\mathcal{C}_\ell$ $(\ell = 1, \ldots, m)$. In particular, the weights of this graph are given by the coefficients a_{ij} or b_{ij} in each elementary reaction of the CRN (3.4). From this graph, we define an $(s \times m)$ matrix Y whose ℓth column entries are given by the coefficients a_{ij} or b_{ij} of the ℓth complex $\mathcal{C}_\ell$, $(\ell = 1, \ldots, m)$ in $\mathcal{C}$. By denoting the ℓth column of Y as $Y(\ell)$, we also define an m-rows vector $\psi(x)$ whose ℓth element is a monomial of the form

$$\psi_\ell(x) = (x^T)^{Y(\ell)^T}. \tag{3.6}$$

The representation of CRN (3.4) in terms of the elements of $\mathcal{C}$ in (3.5) defines a directed graph $G(V, E)$ with a set of vertices $V = \{\mathcal{C}_\ell, \ \ell = 1, \ldots, m\}$ and a set of directed edges $E = \{\mathcal{R}_i, \ i = 1, \ldots, r\}$. The weight of each edge in this graph is given by the rate constant k_i that corresponds to the ith reaction from $\mathcal{C}_p$ to $\mathcal{C}_q$ in (3.5) where $p, q \in \ell = 1, \ldots, m$ and $p \neq q$. Based on this graph, we can construct an $(m \times r)$ *incidence matrix*, I_a, such that for each pair of complexes $(\mathcal{C}_p, \mathcal{C}_q) \in \mathcal{R}_i$ in the ith reaction (3.5), the entries in the ith column of matrix I_a are defined as

$$I_a(\ell, i) \doteq \begin{cases} I_a(p, i) & = -1, \\ I_a(q, i) & = 1, \\ 0, & \text{otherwise.} \end{cases} \tag{3.7}$$

Corresponding to matrix I_a, we also construct an $(r \times m)$ *weighting matrix* I_K such that for each pair of complexes $(\mathcal{C}_p, \mathcal{C}_q) \in \mathcal{R}_i$ in the ith reaction (3.5), the entries in the ith row of matrix I_K is defined as

$$I_K(i, \ell) \doteq \begin{cases} I_K(i, p) = k_i, \\ 0, \ \text{otherwise.} \end{cases} \tag{3.8}$$

Using the matrices Y, I_a, I_K and the vector $\psi(x)$ defined above, the evolution of the species concentrations in the CRN (3.4) can be modeled by the following ODE.

$$\dot{x}(t) = Y I_a I_K \psi(x), \quad x(0) = x_0. \tag{3.9}$$

Equation (3.9) is called the *kinetic realizations* of the CRN (3.4).

There are two main properties of systems with kinetic realizations (3.9): (i) their vector fields are defined by polynomial functions and (ii) their state trajectories always lie in the positive orthant [44, 57]. As will be discussed in Section 3.3 and Chapter 4, these properties are useful for characterizing the system's equilibria or studying the system's stability. Example 6 illustrates the construction of kinetic realizations (3.9) from a CRN of the form (3.4).

Example 6. *Consider the following CRN of the Brusselator system consisting chemical species X, Y with concentration $[X], [Y]$, respectively [44].*

$$2X + Y \xrightarrow{k_1} 3X \qquad\qquad \emptyset \underset{k_3}{\overset{k_4}{\rightleftarrows}} X \xrightarrow{k_2} Y$$

This CRN can be rewritten as a set of elementary reactions of the form

$$\mathcal{R}_1 : 2X + Y \xrightarrow{k_1} 3X, \quad \mathcal{R}_2 : X \xrightarrow{k_2} Y, \quad \mathcal{R}_3 : X \xrightarrow{k_3} \emptyset, \quad \mathcal{R}_4 : \emptyset \xrightarrow{k_4} X$$
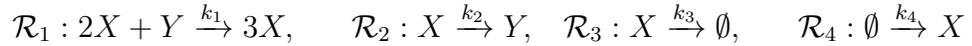
which satisfy the representation in (3.4). This CRN has $s = 2$ species (X and Y), $r = 4$ elementary reactions and $m = 5$ complexes ($\mathcal{C} = \{\mathcal{C}_1, \ldots, \mathcal{C}_5\}$ where $\mathcal{C}_1 = 2X + Y, \mathcal{C}_2 = 3X, \mathcal{C}_3 = X, \mathcal{C}_4 = Y, \mathcal{C}_5 = \emptyset$). Thus, the above CRN can be rewritten in terms of the elements of $\mathcal{C}$ as follows.

$$\mathcal{R}_1 : \mathcal{C}_1 \xrightarrow{k_1} \mathcal{C}_2, \quad \mathcal{R}_2 : \mathcal{C}_3 \xrightarrow{k_2} \mathcal{C}_4, \quad \mathcal{R}_3 : \mathcal{C}_3 \xrightarrow{k_3} \mathcal{C}_5, \quad \mathcal{R}_4 : \mathcal{C}_5 \xrightarrow{k_4} \mathcal{C}_3 \tag{3.10}$$
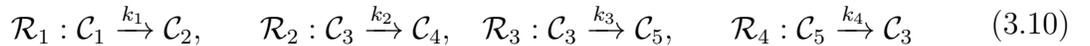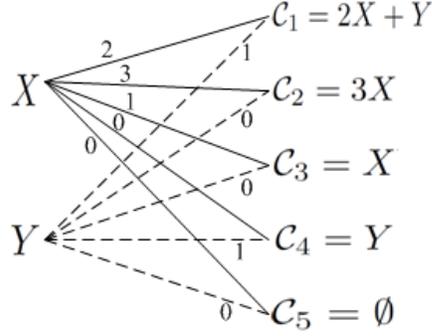
Figure 3.1. Bipartite graph of the Brusselator system.

We use x_1 and x_2 to denote the concentrations of species $[X]$ and $[Y]$, respectively, and define a vector of species concentrations $x = [x_1, x_2]^T$.

Figure 3.1 shows a bipartite graph that defines the association of species X and Y to the set of complexes in $\mathcal{C}$. Based on this graph, the matrix Y in equation (3.9) is

$$Y = \begin{bmatrix} 2 & 3 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

As a result, the vector of monomials, $\psi(x)$, in (3.6) is given by

$$\psi(x) = \begin{bmatrix} x_1^2 x_2 \\ x_1^3 \\ x_1 \\ x_2 \\ 1 \end{bmatrix}.$$

Figure 3.2 shows a directed graph associated with the CRN in (3.10). Using this
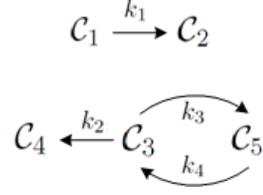
Figure 3.2. Directed graph of the Brusselator system.

graph and the rules in equations (3.7)-(3.8), the matrices I_a and I_K in (3.9) are

$$I_a = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & -1 & -1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}, \quad I_K = \begin{bmatrix} k_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & k_2 & 0 & 0 \\ 0 & 0 & k_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & k_4 \end{bmatrix}.$$

A kinetic realization of the form (3.9) for this CRN is then given by

$$\dot{x}(t) = Y I_a I_K \psi(x)$$

$$= \begin{bmatrix} 2 & 3 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & -1 & -1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} k_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & k_2 & 0 & 0 \\ 0 & 0 & k_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & k_4 \end{bmatrix} \begin{bmatrix} x_1^2 x_2 \\ x_1^3 \\ x_1 \\ x_2 \\ 1 \end{bmatrix}. \tag{3.11}$$

As a result, the following ODE model of the Brusselator system can be obtained.

$$\begin{aligned} \dot{x}_1(t) &= k_1 x_1^2 x_2 + k_4 - k_2 x_1 - k_3 x_1, \\ \dot{x}_2(t) &= -k_1 x_1^2 x_2 + k_2 x_1, \end{aligned} \tag{3.12}$$

with $x(0) = x_0 = [x_1(0),\ x_2(0)]^T$.

49

Now consider the kinetic realizations in (3.9) and define a real-valued matrix $N = YI_a$ and a vector of monomials $v(x, k) = I_K \psi(x)$. Then (3.9) can be written as

$$\dot{x}(t) = Nv(x, k) = N \operatorname{diag}(k)x^Z, \tag{3.13}$$

where $N \in \mathbb{Z}^{n \times m}$ is known as the *stoichiometric matrix* and $v(x, k) \in \mathbb{R}_+^m$ is a nonnegative-valued *flux vector* which satisfies a decomposition of the form

$$v(x, k) = \operatorname{diag}(k)x^Z, \tag{3.14}$$

in which k is a vector of reaction constants and $Z \in \mathbb{Z}_{\geq 0}^{n \times m}$ is a matrix of nonnegative integers whose ith column denotes the multi-index of the ith monomial $v_i(x, k)$. The kinetic realizations in (3.13) was introduced in [22] and used to study the dynamics of chemical species in CRNs based on the dynamics of their reaction fluxes/complexes. Since the state trajectories of (3.9) always lie in the nonnegative orthant, then the state trajectories of (3.13) are also guaranteed to be nonnegative. In particular, for the ODE $\dot{x}(t) = Nv(x(t), k)$ in (3.13) and any time $t_1 < t_2 \in [0, t]$, the vector difference $x(t_2) - x(t_1)$ is an element of $\operatorname{Im}(N)$. By integrating this difference along the solution of $x(t)$, we have

$$x(t_2) = x(t_1) + \int_{t_1}^{t_2} Nv(x(t), k)dt,$$

which implies

$$x(t_2) - x(t_1) = N \int_{t_1}^{t_2} \operatorname{diag}(k)\psi(x(t))dt.$$

The above equation shows that the difference $x(t_2) - x(t_1)$ is a linear combination of the column of matrix N with nonnegative coefficients $c_i = k_i \int_{t_1}^{t_2} x(t)^{\alpha_i} dt$ in which α is the multi-index of the monomials in $\psi(x)$. Thus for any initial condition $x_0 \in \mathbb{R}_{\geq 0}^n$, the state trajectories at any time $t \geq 0$ are guaranteed to stay in the positive orthant

50

$(x_0 + \mathrm{Im}(N)) \cap \mathbb{R}^n_{\geq 0}$ [57].

For a kinetic realization of the form (3.13), let $w = w_1, \ldots, w_m$ be a basis vector of the orthogonal complement of $\mathrm{Im}(N)$. This means w is a basis of $\ker(N^T)$ with dimension $d = \dim(\ker(N^T)) = m = \mathrm{rank}(N)$. Let $x(0) = x_0$ be an initial condition and define $c_i = w_i^T x_0$. We then have

$$x_0 + \mathrm{Im}(N) = \{x \in \mathbb{R}^n \mid w_i^T x - c_i = 0 \quad \text{for } i = 1, \ldots, d\}. \tag{3.15}$$

Equation (3.15) defines an invariant space of equationt (3.13) [153, 57]. In the context of CRN, equation (3.15) is often interpreted as a *conservation relation* describing the total conserved concentrations in the CRN. Thus if the rank, $\mathrm{rank}(N)$, of matrix N in (3.13) is less than the number of state variables, n, then (3.13) satisfies the following $n - \mathrm{rank}(N)$ algebraic equations.

$$w_i^T x = c_i, \qquad i = 1, \ldots, n - \mathrm{rank}(N). \tag{3.16}$$

This thesis will focuses on using the kinetic realizations in (3.13). Example 3.17 illustrates the construction of the kinetic realizations (3.13).

Example 6 (continuing from p. 47). *Using the matrices Y, I_a, I_K and vector $\psi(x)$ in 3.11, a kinetic realization (3.13) of the Brusselator system is given by*

$$\dot{x}(t) = Nv(x, k)$$

$$= \begin{bmatrix} 1 & -1 & -1 & 1 \\ -1 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} k_1 x_1^2 x_2 \\ k_2 x_1 \\ k_3 x_1 \\ k_4 \end{bmatrix}. \tag{3.17}$$

The matrices $diag(k)$ and Z such that $v(x,k) = diag(k)x^Z$ are

$$diag(k) = \begin{bmatrix} k_1 & 0 & 0 & 0 \\ 0 & k_2 & 0 & 0 \\ 0 & 0 & k_3 & 0 \\ 0 & 0 & 0 & k_4 \end{bmatrix}, \quad Z = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}^T. \tag{3.18}$$

The rank of matrix N $(rank(N) = 2)$ in (3.17) is equal to the number of state variables. Thus, the kinetic realization in (3.17) has no conservation relation.

3.2.2 Kinetic Realizations of Nonnegative Systems

Recall from Section 3.2.1 that the kinetic realizations of a CRN is given by

$$\dot{x}(t) = Y I_a I_K \psi(x) \tag{3.19}$$

where matrices Y, I_a, I_K and vector $\psi(x)$ were constructed from the graphs of the CRN. By defining a matrix $N = Y I_a$ and a vector of monomials $v(x,k) = I_K \psi(x)$, then (3.19) can be rewritten as

$$\dot{x}(t) = f(x,k) = Nv(x,k), \tag{3.20}$$

where $v(x,k)$ is a nonnegative-valued vector of monomials in variables x. Recall that the main properties of systems with kinetic realizations (3.19)-(3.20) are that (i) the systems' vector fields are polynomials and (ii) the system's trajectories are always nonnegative. Note that while any polynomial dynamical system can be written as in (3.20), not all of them have kinetic realizations of the form (3.19). The class of polynomial systems that have kinetic realizations is known as *mass action systems* [67]. A sufficient condition for polynomial systems to have a kinetic realization (3.19)

was presented in [67] as stated in Lemma 3.2.1.

Lemma 3.2.1 ([67]). *An m polynomial systems $f_1, \ldots, f_m$ in n variables $x = (x_1, \ldots, x_n)$ is a mass action kinetic systems if and only if for $i = 1, \ldots, n$, there exist real polynomials $g_i(x), h_i(x)$ with nonnegative coefficients such that $f_i = g_i(x) - x_i h_i(x)$.*

Lemma 3.2.1 essentially states that a polynomial dynamical system of the form $\dot{x}_i = f_i(x, k)$ will have a kinetic realization if its vector fields can be written as

$$f_i(x, k) = g_i(x, k) - x_i h_i(x, k) \tag{3.21}$$

in which $g(\cdot)$ and $h(\cdot)$ are polynomials with nonnegative coefficients. Note that any polynomial system which satisfies the condition in (3.21) is guaranteed to be nonnegative [65]. This is because a necessary and sufficient condition for the trajectories of a dynamical system $\dot{x}_i(t) = f(x, k)$ to be nonnegative is that its vector fields satisfy $f_i(x, k) \geq 0$ for all x in which $x_i = 0$ [64, 65]. Setting $x_i = 0$ in (3.21), we have that $f_i(x, k) = g_i(k) \geq 0$ due to the nonnegativity of the coefficients of $g_i(x, k)$ and therefore guarantees the satisfaction of the requirements for the system to be nonnegative.

From the above discussion, it can be concluded that the kinetic realizations (3.9) and (3.13) can be constructed for polynomial systems with vector fields that satisfy the condition (3.21). Below, we argue that such requirements are not too restrictive and are satisfied by the models of many real life systems.

- The first requirement that the system's vector fields to be polynomial functions is not too restrictive. This is because any smooth function with bounded variation defined on a compact set can be approximated as polynomial function using Taylor or higher order polynomials approximation methods [138, 55]. Moreover, there exist various methods to transform nonpolynomial functions into polynomial functions [83, 71, 43].

- The second requirement that the system's vector fields satisfy the condition in (3.21) is also not too restrictive. This is because many models of real life

systems (such as biological, ecological, or compartmental systems) satisfy this property [65, 64]. In particular, these models usually take the Lotka-Volterra type representation and so the transformation methods described in [83, 71, 43] can be used to obtain representation of the form (3.21).

The above argument suggest that there exist a large number of real life systems (other than CRN) whose models have kinetic realizations (3.19) and (3.20). This thesis will focus on studying a class of polynomial systems that have kinetic realizations (3.20). We refer to these systems as *nonnegative systems with kinetic realizations.* Specifically, a dynamical system modeled by differential equations

$$\dot{x}_i(t) = f_i(x, k), \quad x_i(0) = x_{i0}, \ i = 1, \ldots, n$$

is said to be nonnegative system with kinetic realization if its vector fields are polynomial functions that satisfy the condition in (3.21). Note that the construction of kinetic realizations (3.20) does not require the construction of the system's CRN. As long as the system's vector fields satisfy the condition in (3.21), then a kinetic realization (3.20) can directly be constructed by collecting all monomials in the system's vector fields in a vector $v(x, k)$ and then constructed the corresponding matrix N. By Lemma 3.2.1, the satisfaction of condition (3.21) guarantees that a kinetic realization of the form (3.19) can also be constructed. Example 7 illustrates the construction of kinetic realizations (3.20) for a model of tritrophic food web in ecology [68].

Example 7. *This example illustrates the construction of a kinetic realization (3.20) for a tritrophic food web model described in [68]. The differential equation model of the system is given by*

$$
\begin{aligned}
\dot{x}_1 &= x_1 (1 - x_1) - \frac{k_1 x_1 x_2}{k_2 + x_1} \\
\dot{x}_2 &= \frac{k_3 x_1 x_2}{k_2 + x_1} - k_4 x_2 x_3 - k_5 x_2 \\
\dot{x}_3 &= k_6 x_2 x_3 - k_7 x_3
\end{aligned}
$$

where x_1, x_2 and x_3 denote the biomasses of producers, consumers, and top predators, respectively. For each $i = 1, \ldots, 7$, the positive parameter k_i describes the rate of interaction between different species. Note that the system's vector fields are not polynomial functions due to the rational functions that appear in the first and the second state's equations. By introducing an augmented state $x_4 = (k_2 + x_1)^{-1}$ with time derivative

$$\dot{x}_4 = (\partial x_4 / \partial x_1) \dot{x}_1 = x_1 x_4^2 (x_1 - 1) + k_1 x_1 x_2 x_4^3,$$

the original system's model can be rewritten as

$$\begin{aligned}
\dot{x}_1 &= x_1 (1 - x_1) - k_1 x_1 x_2 x_4 \\
\dot{x}_2 &= k_3 x_1 x_2 x_4 - k_4 x_2 x_3 - k_5 x_2 \\
\dot{x}_3 &= k_6 x_2 x_3 - k_7 x_3 \\
\dot{x}_4 &= x_1 x_4^2 (x_1 - 1) + k_1 x_1 x_2 x_4^3
\end{aligned} \qquad (3.22)$$

Clearly the vector fields of the extended model are now polynomial functions. Now note that equation (3.22) can be rewritten in the form

$$\begin{aligned}
\dot{x}_1 &= x_1 - x_1 (x_1 + k_1 x_2 x_4) \\
\dot{x}_2 &= k_3 x_1 x_2 x_4 - x_2 (k_4 x_3 + k_5) \\
\dot{x}_3 &= k_6 x_2 x_3 - x_3 (k_7) \\
\dot{x}_4 &= \left(x_1^2 x_4^2 + k_1 x_1 x_2 x_4^3 \right) - x_4 (x_1 x_4)
\end{aligned}$$

which satisfies the condition in (3.21) with $(g_1, h_1) = (x_1, x_1 + k_1 x_2 x_4)$, $(g_2, h_2) = (k_3 x_1 x_2 x_4, k_4 x_3 + k_5)$, $(g_3, h_3) = (k_6 x_2 x_3, k_7)$ and $(g_4, h_4) = (x_1^2 x_4^2 + k_1 x_1 x_2 x_4^3, x_1 x_4)$. By Lemma 3.2.1, model (3.22) is guaranteed to have kinetic realizations.

We now construct a kinetic realization of the form (3.20). By collecting the unique

monomials of (3.22)'s vector fields in a vector $v(x,k)$ and arranging their correspond-

ing coefficients in a matrix N, then a matrix N and a vector $v(x,k)$ that form a kinetic

realization $\dot{x} = Nv(x,k)$ for system (3.22) are

$$
N = \begin{bmatrix}
1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & -1 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1
\end{bmatrix},
$$

$$
v(k,x) = [x_1, x_1^2, k_1 x_1 x_2 x_4, k_3 x_1 x_2 x_4, k_4 x_2 x_3, k_5 x_2, k_6 x_2 x_3, k_7 x_3, x_1^2 x_4^2, x_1 x_4^2, k_1 x_1 x_2 x_4^3]^T.
$$
(3.23)

The matrices $diag(k)$ and Z such that $v(x,k) = diag(k)x^Z$ are

$$
diag(k) = diag\ ([1, 1, k_1, k_3, k_4, k_5, k_6, k_7, 1, 1, k_1]),
$$

$$
Z = \begin{bmatrix}
1 & 2 & 1 & 1 & 0 & 0 & 0 & 0 & 2 & 1 & 1 \\
0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 2 & 2 & 3
\end{bmatrix}.
$$
(3.24)

Remark 3.2.2. As described in Section (3.2.1), the construction of the kinetic realiza-

tions (3.19) requires the graph of the CRN in (3.4). So in order to compute such a

realization from an ODE model $\dot{x}(t) = f(x,k)$, an equivalent CRN structure (3.4) of

the corresponding ODE should be constructed. The problem of constructing a CRN

graph of an ODE is often called the *inverse realization problem* [67, 143–145, 80, 79].

In particular, Lemma 3.2.1 gives a sufficient condition under which a CRN graph of

an ODE can be realized [67]. As shown in [67, 143], Lemma 3.2.1 also identifies a

method to construct a CRN model (3.4) from an ODE of the form

$$\dot{x}_i = f_i(x, k), \quad i = 1, \ldots, n,$$
$$= \sum_{j=1}^{z_i} k_{ij} x^{[\alpha^{ij}]}. \tag{3.25}$$

where α is the multi-index associated to monomials x^α and $f(x) = \sum_{k=1}^{z} c_k x^{[\alpha^k]}$ is a representation of polynomial $f(x)$ as linear combination of z monomials with coefficient k. The basic approach in this method is to construct an equivalent elementary reaction for each monomial in $f_i(x)$ using a set of rules that are defined based on the *sign of each monomial* and *in which polynomial f_i* that particular monomial appears. Algorithm 1 summarizes these rules. Given a polynomial dynamical system (3.25) with z_i monomials, Algorithm 1 produces at most $n(\mathcal{R}) = \sum_{i=1}^{n} z_i$ elementary reactions of the form (3.4). The output of Agorithm 1 may not be optimal in the sense that the number of constructed elementary reactions may exceeds the minimum number of elementary reactions required to construct the corresponding ODE model. In such a case, one may use the method in [144] to obtain a minimal realization based on Algorithm 1's output. Using the obtained set of elementary reactions, both realizations (3.19) and (3.20) can be constructed.

3.2.3 Properties of Kinetic Realizations

Consider a polynomial system $\dot{x}(t) = f(x, k)$ whose vector fields satisfy condition (3.21) and assume that the system's kinetic realization takes the form

$$\dot{x}(t) = Nv(x, k).$$

Algorithm 1 Realization of CRN from ODE model (adapted from [67, 143])

1: **procedure** ELEMENTARYREACTION(kx^α)
2:     **if** ($kx^\alpha \in f_i$) and (sign(kx^α) is *positive*) **then**
3:         build reaction $\mathcal{R}$: $\sum_{i=1}^n \alpha_i x_i \xrightarrow{k} (\sum_{i=1}^n \alpha_i x_i) + x_i$
4:     **else**
5:         **if** ($kx^\alpha \in f_i$) and (sign(kx^α) is *negative*) **then**
6:             build reaction $\mathcal{R}$: $\sum_{i=1}^n \alpha_i x_i \xrightarrow{k} (\sum_{i=1}^n \alpha_i x_i) - x_i$
7:         **end if**
8:     **end if**
9:     Return $\mathcal{R}$
10: **end procedure**

The *state equilibria, x^** of this realization is defined as

$$x^* = \{x \in \mathbb{Q}^n(k) : Nv(x, k) = 0\}. \tag{3.26}$$

Computing the analytical expression of equilibria (3.26) in high dimensional systems usually requires the use of symbolic computational methods. One such method is based on computing the zeros of the Gröbner basis of the polynomial equations $Nv(x, k) = 0$ [25, see also Chapter 2]. A Gröbner basis for a system of polynomial equations can be computed using Buchberger's algorithm [14, 25]. However, the standard Buchberger's algorithm has a drawback in that the degree of the computed bases grows in a doubly exponential manner with respect to the number of the unknown variables [39, see also Chapter 2].

One alternative approach for computing the equilibria in (3.26) is by studying the solution of kinetic realizations (3.20) in the flux space, $v \in \mathbb{R}^m$ [57]. Specifically, by considering continuous mapping functions $v : \mathbb{R}^n \to \mathbb{R}^m$ and $g : \mathbb{R}^m \to \mathbb{R}^n$, then the

following statements for realization (3.20) are equivalent [56, 57].

$$\exists x \in \mathbb{R}^n \text{ with } g(v(x,k)) = 0,$$

$$\exists v \in \mathbb{R}^m \text{ with } v \in \text{Im}(v(x,k)) \text{ and } g(v) = 0.$$

In kinetic realizations (3.20), the function $v(\cdot)$ is the nonnegative-valued flux vector $v(x,k) = \text{diag}(k)x^Z$ whereas the function $g(\cdot)$ is the matrix N. From Sections 3.2.1-3.2.2, we know that $v(x,k) \geq 0$ and so the above equivalence can be rewritten as

$$\exists x \in \mathbb{R}^m_{\geq 0} \text{ with } Nv(x,k) = 0.$$
$$\exists v \in \mathbb{R}^m_{\geq 0} \text{ with } v \in \text{Im}(v(\mathbb{R}^n_{\geq 0})) \text{ and } Nv = 0. \tag{3.27}$$

By the nonnegativity of v, the *flux equilibria* $v^* = \{v \in \mathbb{R}^m \mid g(v) = 0\}$ is given by

$$\exists v^* \in \mathbb{R}^m \text{ with } v^* \in \text{Im}(v(\mathbb{R}^n_{\geq 0})) \text{ and } v^* \in \ker(N) \cap \mathbb{R}^m_{\geq 0}. \tag{3.28}$$

The above observation suggests the following two important consequences:

First: it means that the flux equilibria form a convex polyhedral cone [22, 87, 57]

$$v^* \in \mathcal{K}_v = \ker(N) \cap \mathbb{R}^m_{\geq 0} = \left\{ v \in \mathbb{R}^m_{\geq 0} : v = \Sigma_{i=1}^q \lambda_i E_i \right\}. \tag{3.29}$$

The cone, $\mathcal{K}_v$, in equation (3.29) is finitely generated by a set of extreme rays, $E_i \in \mathbb{R}^m_{\geq 0}$ for $i = 1, 2, \ldots, q$. Such rays are routinely computed using tools such as `efmtool` [149, 87]. Every flux equilibrium in $\mathcal{K}_v$ can therefore be parameterized with respect to these rays. In equation (3.29), the parameters $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_q)$ are called the *convex parameters* [22] and so any equilibrium flux can be written as $v^*(\lambda)$ a linear function of the convex parameters.

Second: it means that any flux, v, in kinetic realizations (3.20) must satisfy

$$v_i = k_i x^{z_i}, \quad i = 1, 2, \ldots, m. \tag{3.30}$$

Equations (3.30) are *binomials* in $\mathbb{R}(k)[x, v]$ and this system's zeros characterize both the flux equilibria, v^*, and the state equilibria, x^*. The ideal generated by these binomials is a *toric ideal* [7, 57] for which efficient algorithms for computing a Gröbner basis are available [140, 62]. Thus, the equilibria of the system can be solved in terms of its flux equilibra and the system's parameters.

The preceding two consequences of kinetic realizations can be summarized as

(i) any flux equilibrium can be expressed as a function $v^*(\lambda)$ in terms of the convex parameters (λ), and

(ii) any state equilibrium can be expressed as a rational function $x^*(v^*, k)$ of the flux equilibria (v^*) and the system's parameters (k).

Using the expression $v^*(\lambda)$ of the flux equilibria in terms of the convex parameters, one may then parameterize the state equilibria $x^*(\lambda, k) \in \mathbb{Q}^n(k, \lambda)$ in terms of the system's parameters, k, and the convex parameters, λ. Section 3.3 discusses the details of this computation.

Another property of kinetic realizations (3.20) is that the system's Jacobian matrix can be parameterized as [57, 58]

$$J(\lambda, k) = N \mathrm{diag}(v^*(\lambda)) Z^T \mathrm{diag}(1/x^*(\lambda, k)). \tag{3.31}$$

The Jacobian in (3.31) is now parameterized in terms of parameters k and λ. Thus, this Jacobian can be used to study the properties of kinetic realizations' equilibria under parametric variations. Chapter 4 will discuss this type of analysis.

3.3 Equilibria of Kinetic Realizations

Section 3.2.3 stated that the special structure of kinetic realizations (3.20) allows one to compute an expression for the system's state equilibria, x^* in terms of the system's parameters, k, and some convex parameters, λ. This section discusses a computational method for characterizing the kinetic realizations' state equilibria, x^*. The method consists of two main steps. The first step computes a Gröbner basis for the toric ideal generated by the binomial system (3.30). This basis is defined only with respect to the flux variable, v, and so its computation involves the elimination of unknown variables x from the binomial system (3.30). The variety of this basis therefore defines the flux equilibria, v^*. The second step extends the flux equilibria, v^*, over the original binomial ideal and then uses equations (3.29)-(3.30) to compute the state equilibria, x^*. Such an extension is achieved through a transformation using Hermite normal form [142, Chapter 3.2]. Thus, the two steps essentially correspond to the elimination and extension steps for computing a Gröbner basis of the binomial ideal (3.30) [7, 57]. Each step is detailed below.

3.3.1 Computation of the Flux Equilibria

This section discusses the first step that computes a Gröbner basis of the binomial system (3.30).

Let $R(k)[x]$ and $R(k)[v]$ be polynomial rings in the unknowns $x = (x_1, \ldots, x_n)$ and $v = (v_1, \ldots, v_m)$, respectively. The mapping $v(x, k)$ from the state $x \in \mathbb{R}^n$ to the flux $v \in \mathbb{R}^m$ in (3.14) satisfies

$$v(x, k) : \mathbb{R}^n_{\geq 0} \mapsto \mathbb{R}^m_{\geq 0}, \qquad x \mapsto v(x, k).$$

The image of $v(x,k)$ is then given by

$$v_1(x,k) = k_1 x^{Z_1}, \ v_2(x,k) = k_2 x^{Z_2}, \ \ldots, \ v_m(x,k) = k_m x^{Z_m}.$$

The above relation shows that a flux equilibrium, $v^* \in \mathbb{R}_{\geq 0}^m$, will corresponds to a state equilibrium, $x^* \in \mathbb{R}_{\geq 0}^n$, if and only if v^* lies on the image of the function $v(x^*, k)$, i.e. if

$$v^* \in \text{Im}(v(x^*, k)).$$

This requirement is satisfied if

$$v_i - v_i(x,k) = 0, \quad \text{for } i = 1, \ldots, m. \tag{3.32}$$

The expression on the left hand side of (3.32) defines a binomial ideal over $\mathbb{R}(k)[x,v]$. Since the flux vector, v, satisfies the decomposition $v = \text{diag}(k)x^Z$ where $Z \in \mathbb{Z}_{\geq 0}^{n \times m}$, then matrix Z in this mapping induces a substitution homomorphism on $v(x,k)$ and that $v \in \mathbb{R}(k)[v]$ is a toric ideal associated with the function $v(x,k)$ [7]. This implies that any flux equilibrium, v^*, should lie on the *toric variety* $V(\mathcal{I})$ of the toric ideal $\mathcal{I} = \langle v_i - v_i(x,k) \rangle \cap \mathbb{R}(k)[v]$. Since the varieties of an ideal are equivalent to the varieties of the ideal's Gröbner basis, then the Elimination Theorem can be used to compute a Gröbner basis of $\mathcal{I}$. This Gröbner basis will be a defining ideal of the toric variety $\mathbb{V}(\mathcal{I})$ and can therefore be used to compute the flux equilibria, v^*. Below, we describe the procedure to compute v^*.

First, define an ideal $I \in \mathbb{R}(k)[x,v]$ that corresponds to the binomial system (3.32)

$$I = \langle v_1 - k_1 x^{Z_1}, \ldots, v_m - k_m x^{Z_m} \rangle \subseteq \mathbb{R}(k)[x,v]. \tag{3.33}$$

By Hilbert's Basis Theorem [25], we know that I is generated by a finite number of elements of its Gröbner basis [25]. Since $I \in \mathbb{R}(k)(x,v)$ is a binomial ideal, then its

Gröbner bases will also generate a binomial ideal in $\mathbb{R}(k)(x, v)$ [103]. The associated toric ideal $\mathcal{I} \subseteq \mathbb{R}(k)[v]$ (note the difference on the unknown variables of the ring) is defined as

$$
\begin{aligned}
\mathcal{I} &= \langle v_1 - k_1 x^{Z_1}, \ldots, v_m - k_m x^{Z_m} \rangle \cap \mathbb{R}(k)[v], \\
&= I \cap \mathbb{R}(k)[v].
\end{aligned} \tag{3.34}
$$

Now let $\mathbb{G}$ be the Gröbner basis of I in (3.33) with respect to a lex order that eliminates variables x. Note that the toric ideal $\mathcal{I}$ in (3.34), obtained from the intersection $I \cap \mathbb{R}(k)[v]$, is the nth elimination ideal of I (as it is obtained by eliminating n variables x from I) defined in $\mathbb{R}(k)[v]$. Using the Gröbner basis $\mathbb{G}$ of ideal I, the Elimination Theorem (cf. Chapter 2) implies that the basis

$$
\mathbb{G}_n(v) = \mathbb{G} \cap \mathbb{R}(k)[v]
$$

is a Gröbner basis for $I \cap \mathbb{R}(k)[v] = \mathcal{I}$. This means that the elements in the Gröbner basis of the toric ideal $\mathcal{I}$ are given by those elements of $\mathbb{G}$ that only contain indeterminates v. And since the variety of an ideal is equivalent to the variety of its Gröbner basis [25], the variety $\mathbb{V}(\mathcal{I})$ of the toric ideal (3.34) is defined as

$$
\mathbb{V}(\mathcal{I}) = \{v \in \mathbb{Q}^m(k) : \mathbb{G}_n(v) = 0\} \subseteq \mathbb{R}(k)[v]. \tag{3.35}
$$

As a result, the flux equilibria, v^*, can be computed from the toric variety $\mathbb{V}(\mathcal{I})$, i.e.

$$
v^*(k) \in \mathbb{V}(\mathcal{I}). \tag{3.36}
$$

3.3.2 Computation of State Equilibria

This section describes the second step that computes the state equilibria, x^*, using the flux equilibria in (3.36).

By the Ideal-Variety Correspondence theorem [25], any ideal that vanishes on the variety $\mathbb{V}(I) \in \mathbb{R}(k)[x, v]$ of the binomial ideal (3.32) will also vanishes on the toric variety $\mathbb{V}(\mathcal{I}) \subseteq \mathbb{R}(k)[v]$ in (3.36). But since $v^*(k) \in \mathbb{V}(\mathcal{I}) \subseteq \mathbb{R}(k)[v]$, then $v^*(k)$ only defines partial solutions to $\mathbb{V}(I)$. Thus, the obtained v^* should be extended over the ring $\mathbb{R}(k)[x, k]$ to get the solutions $x^* \in \mathbb{R}(k)[x, v]$. If the solutions x^* exist, then both x^* and v^* define the variety $\mathbb{V}(I) \subseteq \mathbb{R}(k)[x, v]$ of the binomial system (3.32).

The Extension Theorem [25] can be used to show that the partial solutions defined by the flux equilibria, $v^*(k)$, can be extended to get the state equilibria, x^*. To begin, note that equation (3.30) can be rewritten as

$$v = \mathrm{diag}(k)x^Z. \tag{3.37}$$

For a given nonzero solution $v^*(k)$, the above representation satisfies the condition in equation (2.8) of Theorem 2.2.5 for $g_i(\cdot) = \mathrm{diag}(k)$ (with the second term on the right hand side of (2.8) equals to zero) and thereby guarantees the existence of the extended solutions x^*.

Next, we show how to compute the state equilibria x^* from the flux equilibria v^* using the Hermite normal form [142, 58, 103]. First, introduce a coordinate transformation $x^* = \omega^U$ where $\omega \in \mathbb{R}^n$ is an arbitrary vector and U is a unimodular matrix. From linear algebra, we know that the Hermite normal form H of an integer-valued matrix Z satisfies $H = UZ$ [142]. Evaluating (3.37) at state and flux equilibria gives $v^*(k) = \mathrm{diag}(k)(x^*)^Z$. This implies

$$\mathrm{diag}(k)(x^*)^Z = \mathrm{diag}(k)\omega^{UZ} = \mathrm{diag}(k)\omega^H = v^*(k). \tag{3.38}$$

For a given $v^*(k)$, we first computed ω in the equation $\text{diag}(k)\omega^H = v^*(k)$. Using the obtained ω and the unimodular matrix U, the state equilibria can then be computed from equation $x^* = \omega^U$. Both $x^*(k)$ and $v^*(k)$ then define the varieties or the total solution of polynomial equations defined by the binomial ideal I in (3.32).

Now recall from (3.29) that the flux equilibria, v^*, is also defined on the convex cone $\mathcal{K}_v$ and satisfies a parameterization of the form

$$v^*(\lambda) \in \mathcal{K}_v. \tag{3.39}$$

This implies that v^* is given by the intersection of (3.36) and (3.39), i.e.

$$v^*(\lambda, k) = \{v^* \in \mathbb{V}(\mathcal{I})\} \cap \{v^* \in \mathcal{K}_v\} \subseteq \mathbb{R}(\lambda, k)[v]. \tag{3.40}$$

By the substitution of (3.40) into (3.38), then the state equilibria, $x^*(v^*, k)$, obtained from the transformation in (3.38), can be rewritten as a function $x^*(\lambda, k)$ of the system's parameters, k, and the convex parameters, λ.

3.3.3 Algorithm and Examples

Algorithm 2 illustrates the computation of the state and flux equilibria for the kinetic realizations (3.20). A MATLAB software toolkit [147] that implemented Algorithm 2 was also developed in this thesis (cf. Appendix C). In the following, we present some examples that illustrate the computation of kinetic realizations' equilibria. Algorithm 2 is used to guide the presentation of these examples.

Example 6 (continuing from p. 47)**.** *This example illustrates the computation of state and flux equilibria for a kinetic realization of the Brusselator system in (3.17)-(3.18). Step 1: We first compute the expression of the flux equilibria in (3.29). For kinetic*

Algorithm 2 Computing the equilibria of kinetic realizations (3.13).

Input: vector $v(x, k) = (k_1 x^{Z_1}, \ldots, k_m x^{Z_m})^T$ and matrix N in (3.13)

Output: flux equilibria, $v^*(\lambda, k)$, and state equilibria $x^*(\lambda, k)$

Step 1: Compute the flux equilibria v^* (3.29) in terms of the convex parameters λ

1: Use `efmtool` [149] or CellNetAnalyzer [87] to compute vectors E_i such that v^* can be expressed as

$$v^* \in \mathcal{K}_v = \ker(N) \cap \mathbb{R}_{\geq 0}^m = \left\{ v \in \mathbb{R}_{\geq 0}^m : v = \Sigma_{i=1}^q \lambda_i E_i \right\}.$$

Step 2: Compute the toric variety $\mathbb{V}(\mathcal{I})$ in (3.35) of the toric ideal $\mathcal{I}$ in (3.34)

2: Construct the binomial ideal $I \subseteq \mathbb{R}(k)[x, v]$ defined in (3.35), i.e.
$$I = \langle v_1 - k_1 x^{Z_1}, \ldots, v_m - k_m x^{Z_m} \rangle \in \mathbb{R}(k)[x, v]$$

3: Use SINGULAR computer algebra [32] to compute a Gröbner basis $\mathbb{G}$ of $I \in \mathbb{R}(k)[x, v]$ with respect to a lex ordering that eliminates the indeterminates x, i.e. $x_1 \succ \cdots \succ x_n \succ v_1 \cdots \succ v_m$

4: Use $\mathbb{G}$ and the Elimination Theorem to compute the Gröbner basis $\mathbb{G}_n \in \mathbb{R}(k)[v]$ for the nth elimination ideal of I, i.e. $\mathbb{G}_n = \mathbb{G} \cap \mathbb{R}(k)[v]$

- The elements of the Gröbner basis $\mathbb{G}_n$ are given by those elements of $\mathbb{G}$ which only contain variables v.
- The toric ideal (3.37) is given by $\mathcal{I}\langle \mathbb{G}_n \rangle \subseteq \mathbb{R}(k)[v]$ and its toric variety is
$$\mathbb{V}(\mathcal{I}) = \{v \in \mathbb{R}(k)^m : \mathbb{G}_n = 0\} \subset \mathbb{R}(k)[v]$$

5: The flux equilibria defined by the above toric variety is given by $v^*(k) \in \mathbb{V}(\mathcal{I})$.

Step 3: Compute the intersection between v^* that are obtained in Step 1 and Step 2

6: Substitute $v(\lambda)$ from Step 1 to variables v in $\mathbb{V}(\mathcal{I})$. The flux equilibria become
$$v^*(\lambda, k) = \{v^* \in \mathbb{V}(\mathcal{I})\} \cap \{v^* \in \mathcal{K}_v\}$$

Step 4: Use Hermite transformation to compute x^* from v^*

7: Use transformation (3.38) to obtain $x^*(\lambda, k)$ from $v^*(\lambda, k)$.

realization (3.17), two extreme rays were identified by CellNetAnalyzer [87]

$$[E_1, E_2] = \begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 1 & 0 \end{bmatrix}.$$

Thus, the expression of the flux equilibria (3.39) is given by

$$v^*(\lambda) = \sum_{i=1}^{2} \lambda_i E_i = \begin{bmatrix} \lambda_2 \\ \lambda_2 \\ \lambda_1 \\ \lambda_1 \end{bmatrix}. \tag{3.41}$$

<u>Step 2:</u> Next, we compute the flux equilibria defined by the toric variety $\mathcal{I}$ in (3.36). The ideal I formed by the flux vector v_i and the monomial in $v(x, k)$ is given by

$$I = \langle v_1 - k_1 x_1^2 x_2,\ v_2 - k_2 x_1,\ v_3 - k_3 x_1,\ v_4 - k_4 \rangle.$$

A Gröbner basis of I was computed using SINGULAR *[32] and is given by*

$$\mathbb{G}(I) = \{v_4 - k_4,\ k_3 v_2 - k_2 v_3,\ k_1 x_2 v_3^2 - k_3^2 v_1,\ k_3 x_1 - v_3\}.$$

The Gröbner basis of the toric ideal $\mathcal{I}$ in (3.34) was given those elements of $\mathbb{G}$ that only contain the unknown v, i.e.

$$\mathbb{G}_n(\mathcal{I}) = \{v_4 - k_4,\ k_3 v_2 - k_2 v_3\}.$$

The toric ideal, $\mathcal{I}$, is then given by $\mathcal{I} = \langle \mathbb{G}_n \rangle$ and its toric variety is defined as

$\mathbb{V}(\mathcal{I}) = \{\mathbb{G}_n(\mathcal{I}) = 0\}$. The flux equilibria obtained from $\mathbb{V}(\mathcal{I})$ therefore satisfy

$$v_4^* - k_4 = 0, \quad k_3 v_2^* - k_2 v_3^* = 0. \tag{3.42}$$

<u>Step 3:</u> We now compute the intersection of $v^*(k)$ and $v^*(\lambda)$. From (3.41), we have

$$v_1^* = v_2^* = \lambda_2, \quad and \quad v_3^* = v_4^* = \lambda_1.$$

The substitution of (3.41) into (3.42) gives the following equations

$$\begin{aligned} \lambda_1 - k_4 &= 0, \\ k_3 \lambda_2 - k_2 \lambda_1 &= 0. \end{aligned} \tag{3.43}$$

Thus, the flux equilibria can now be expressed either in terms of the convex parameters, λ, or the system's parameters, k, below.

$$v^*(\lambda, k) = \begin{bmatrix} \lambda_2 \\ \lambda_2 \\ \lambda_1 \\ \lambda_1 \end{bmatrix} = \begin{bmatrix} k_2 k_4 / k_3 \\ k_2 k_4 / k_3 \\ k_4 \\ k_4 \end{bmatrix}. \tag{3.44}$$

<u>Step 4:</u> Finally, we use the Hermite transformation (3.38) to compute the expression for the state equilibria, $x^*(\lambda, k)$ from the flux equilibria, $v^*(\lambda, k)$ in (3.44). For matrix Z in (3.18), the unimodular matrix U and the Hermite normal form H such that $UZ = H$ are

$$UZ = H \quad \Leftrightarrow \quad \begin{bmatrix} 0 & 1 \\ 1 & -2 \end{bmatrix} Z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}.$$

Using the transformation in (3.38), the equation $\mathrm{diag}(k)w^H = v^*(\lambda, k)$ with $v^*(\lambda, k)$

defined in (3.44) becomes

$$diag(k)w^H = \begin{bmatrix} k_1 w_2 \\ k_2 w_1 \\ k_3 w_1 \\ w_3 \end{bmatrix} = v^*(\lambda, k).$$

Solving the above equation for w and then using the relation $x^* = w^U$ in (3.38), the expression for the state equilibria $x^*(\lambda, k)$ is given by

$$x_1^* = \frac{\lambda_2}{k_3}, \quad x_2^* = \frac{k_3^2 \lambda_1}{k_1 \lambda_2^2}. \tag{3.45}$$

Note that x_i^*s in (3.45) are explicit functions of the system's and the convex parameters. Thus, for a given set of parameters, equation (3.45) can be used to compute the corresponding state equilibria. For example, let the reaction constants in the Brusselator system be specified as $k_i = 1$, $i = 1, \ldots, 4$. Using the expression for the intersection between $v^*(k)$ and $v^*(\lambda)$ in (3.43), the corresponding convex parameters are $\lambda_1 = \lambda_2 = 1$. The substitution of these parameters into (3.45) gives an equilibrium $x_1^* = x_2^* = 1$. To check this result, we simulated the Brusselator's ODE model using the specified parameters $k_i = 1$, $i = 1, \ldots, 4$. Figure 3.3 plots the system's trajectories for the specified parameters. This figure shows that the trajectories converge to a steady state $x_1^* = x_2^* = 1$, the same equilibrium obtained using (3.45).

In the next example, we illustrate the computation of flux and state equilibria for the tritrophic foodweb model discussed in Example 7.

Example 7 (continuing from p. 54). *In this example, we illustrate the computation of the equilibria for a kinetic realization of the tritrophic foodweb model in (3.23)-(3.24). Again, we follow the routines in Algorithm 2.*

Step 1: First, we compute the expression for the flux equilibria, v^*, in terms of
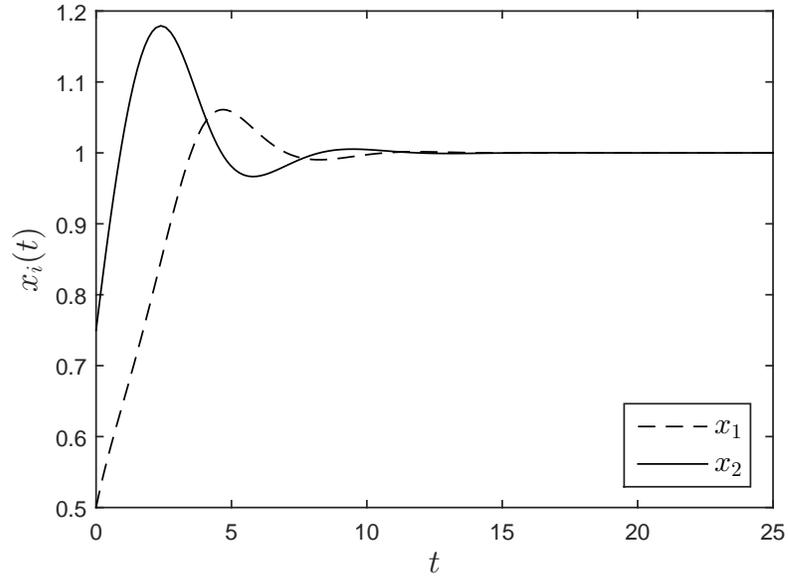
Figure 3.3. State trajectories of the Brusselator system for
$k_i = 1, \ i = 1, \ldots, 4.$

the convex parameters, λ. For the kinetic realization in (3.23), seven extreme rays

$(E_i, i = 1, \ldots, 7)$ were identified from CellNetAnalyzer*[87].*

$$
\begin{bmatrix} E_1 & E_2 & E_3 & E_4 & E_5 & E_6 & E_7 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.
$$

Thus, the expression of the flux equibria in (3.39) is

$$v^*(\lambda) = [\lambda_1 + \lambda_6, \ \lambda_2 + \lambda_7, \ \lambda_1, \ \lambda_2, \ \lambda_3, \ \lambda_3, \ \lambda_4, \ \lambda_5, \ \lambda_4 + \lambda_5, \ \lambda_6, \ \lambda_7]^T. \qquad (3.46)$$

Step 2: Next, we compute the flux equilibria from the toric variety $\mathbb{V}(\mathcal{I})$ in (3.36). The ideal I generated by the flux vector, v_i, and the monomials in $v(x, k)$ is given by

$$I = \langle \, v_1 - x_1, v_2 - x_1 x_4^2, v_3 - x_1^2, v_4 - x_1^2 x_4^2, v_5 - k_7 x_3, v_6 k_6 x_2 x_3, v_7 - k_5 x_2, v_8 - k_4 x_2 x_3,$$

$$v_9 - k_3 x_1 x_2 x_4, v_{10} - k_1 x_1 x_2 x_4, v_{11} - k_1 x_1 x_2 x_4^3 \, \rangle.$$

Using a Gröbner basis $G(I)$ of I computed from Singular [32] and applying eliminations on variables x in $G(I)$, a Gröbner basis for the toric ideal $\mathcal{I}$ is

$$\mathbb{G}_n(\mathcal{I}) = \{ \, k_6 v_7 v_8 - k_5 k_7 v_7, k_6 v_5 - k_4 v_7, k_5^2 v_4^2 - k_3^2 v_6^2 v_9, k_3 v_3 - k_1 v_4, k_5^2 v_2 - v_6^2,$$

$$v_1 v_{11} - v_3 v_{10}, v_1 v_{10} - v_9 \, \}. \quad (3.47)$$

The toric ideal is then defined as $\mathcal{I} = \langle \mathbb{G}_n \rangle$ and the toric variety is given by $\mathbb{V}(\mathcal{I}) = \{ v : \mathbb{G}_n = 0 \}$. Thus, the flux equilibria obtained from $\mathbb{V}(\mathcal{I})$ satisfy

$$v^*(k) = \{ v : \mathbb{G}_n(\mathcal{I}) = 0 \}. \qquad (3.48)$$

Step 3: By substituting $v(\lambda)$ in (3.46) into variables v in (3.47) and using some algebraic simplifications, the following expressions for the flux equilibria were obtained

$$v^*(\lambda, k) = [v_1^*, v_2^*, v_3^*, \lambda_3 + \lambda_4, \lambda_3, \lambda_4, k_6 \lambda_3 / k_4, k_6 \lambda_3 / k_4, \lambda_6, \lambda_6(1 + \lambda_2/(\lambda_1 - \lambda_2)),$$

$$\lambda_2 \lambda_6 / (\lambda_1 - \lambda_2)]^T,$$

where

$$v_1^* = (k_7/k_6)^2 - (k_1/k_6)(\lambda_3 + k_5 k_7/k_6),$$

$$v_2^* = (k_7/k_6)^2 - (k_1/k_6 - k_1/k_3)(\lambda_3 + k_5 k_7/k_6),$$

$$v_3^* = (k_1/k_3)(\lambda_3 + k_5 k_7/k_6).$$

Note that we used $\mu = (k_1, k_5, k_6, \lambda_3, \lambda_4, \lambda_5, \lambda_6, \lambda_7)$ *to parameterize the flux equilibria.*

<u>*Step 4:*</u>*We now use the Hermite transformation (3.38) to compute the state equilibria* $x^*(\lambda, k)$ *from* $v^*(\lambda, k)$. *A unimodular matrix* U *for matrix* Z *in (3.24) is*

$$U = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

Using (3.38), the following expressions of the state equilibria, x^*, *were obtained.*

$$x_1^* = \lambda_1 + \lambda_6, \quad x_2^* = \frac{k_7}{k_6}, \quad x_3^* = \frac{k_3 k_6 \lambda_6}{k_1 k_4 k_7} - \frac{k_5}{k_4}, \quad x_4^* = \frac{k_6 \lambda_6}{k_1 k_7 (\lambda_1 + \lambda_6)}. \tag{3.49}$$

The above result can be used to compute the system's equilibria for a given set of parameters. Consider a parameter set $k = [0.5, 0.5, 0.5, 0.1, 0.25, 0.25, 0.2]$. Using the intersection between $v^*(k)$ and $v^*(\lambda)$ defined in Step 3, the corresponding convex parameters were computed to be $\lambda_1 = 0.453, \lambda_2 = 0.25, \lambda_6 = 0.2$ and $\lambda_7 = 0.11$. Thus, the equilibria of the first three states for these parameters are $x_1^* = 0.653, x_2^* = 0.8, x_3^* = 0.332$. To check this result, we simulated the ODE of the tritrophic foodweb model using the specified parameters. Figure 3.4 plots the state trajectories of the model for the chosen parameters. It can be seen that the state trajectories converge to an equilibrium $x_1^* = 0.65, x_2^* = 0.8, x_3^* = 0.33$. These values are the same with that computed using (3.49).
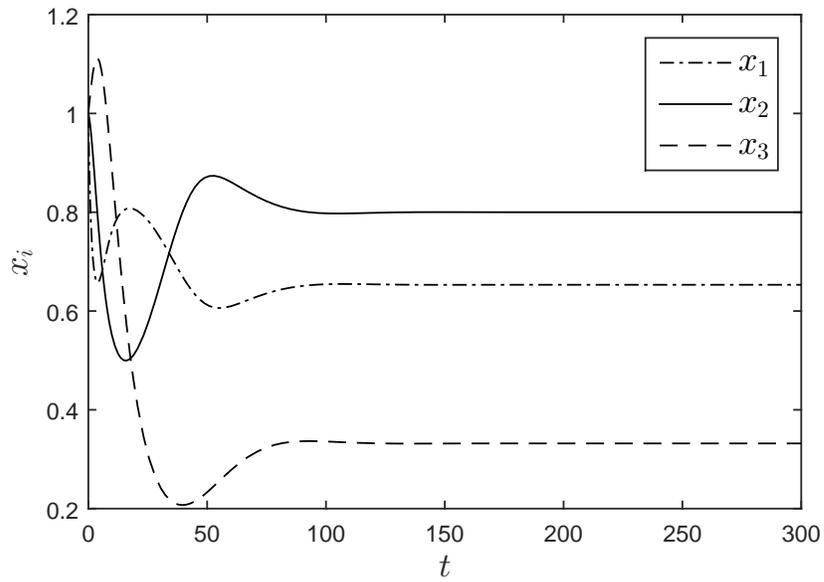
Figure 3.4. Trajectories of the tritrophic foodweb model for $k = [0.5, 0.5,$
$0.5, 0.1, 0.25, 0.25, 0.2]$.

3.4 Conclusion

This chapter has presented a method to compute an equilibrium parameterization of nonnegative systems with kinetic realizations. As will be shown in the Chapter 4, this equilibrium parameterization will be used to help simplify the minimum distance to bifurcation problem.

CHAPTER 4

FORECASTING BIFURCATION-INDUCED REGIME SHIFTS

4.1   Introduction

This chapter presents a method to forecast bifurcation-induced regime shifts. Recall that bifurcation-induced regime shifts occur because the system's equilibrium undergoes a bifurcation as a result of variations in the system's parameters. The method presented in this chapter characterizes such regime shifts as a minimum *distance to bifurcation* (D_2B) problem. The solution to this problem is the minimum distance $\gamma = \inf_k |k^* - k^0|$ from a nominal parameter k^0 to the closest critical parameter k^* at which a bifurcation occurs. The method discussed in this chapter uses this distance as a measure of how close a system is to a bifurcation-induced regime shift. This chapter focuses on the computation of γ in a class of nonnegative systems with kinetic realizations. A dynamical system $\dot{x}_i(t) = f_i(x, k)$ with polynomial vector fields $f_i(x, k)$ is said to have a kinetic realization if there exist polynomials $g_i(x, k)$ and $h_i(x, k)$ with nonnegative coefficients such that $f_i(x, k)$ can be rewritten as [67]

$$f_i(x, k) = g_i(x, k) - x_i h_i(x, k).$$

One important property of systems with kinetic realizations is that their special structure allows one to compute expressions for the system's equilibria as a rational function of the system's parameters and some convex parameters. We show that this property helps simplify the computation of a lower bound for the minimum D_2B.

The approach to compute the quantity γ discussed in this chapter consists of two main steps:

- First, we use the method to compute an expression of kinetic realizations' equilibria that simplifies the optimization problem for computing the D_2B. In particular, we express the constraints on the formulated optimization problems only in term of the system's parameters rather than the system's parameters and the equilibria.

- Second, by rewriting the constraints of the optimization problem as a semialgebraic set conditions in the parameter space, we show that we can recast the computation of γ as a polynomial optimization problem (POP). Using the SOS relaxation of this POP, we formulate an SOS optimization problem to compute bounds on γ. This SOS optimization problem is then solved using semidefinite programming solvers [121, 100, 69].

4.1.1 Background and Prior Work

To motivate the discussion, recall the model of the lake eutrophication process below

$$\dot{x} = a - bx + \frac{x^2}{1 + x^2}$$

where the state variable x denotes the Phosporus (P) concentration in the lake water column and the parameters $a \geq 0$ and $b \geq 0$ denote the rates of inflow and outflow of P into and out of the lake, respectively. The equilibria of this system are defined by those values of x at the intersections of the curves of production function $f(x)$ and loss function $g(x)$. This means that the equilibria are those values of x such that

$$\underbrace{a + \frac{x^2}{1 + x^2}}_{f(x)} = \underbrace{bx}_{g(x)}.$$

First, let us consider the characteristic of these equilibria when one of the system's parameter is varied. For example, assume that parameter b is held fixed while parameter a is varied. The values of the system's equilibria for different a can be

(a) One-parameter bifurcation diagram.

(b) Two-parameters bifurcation diagram.
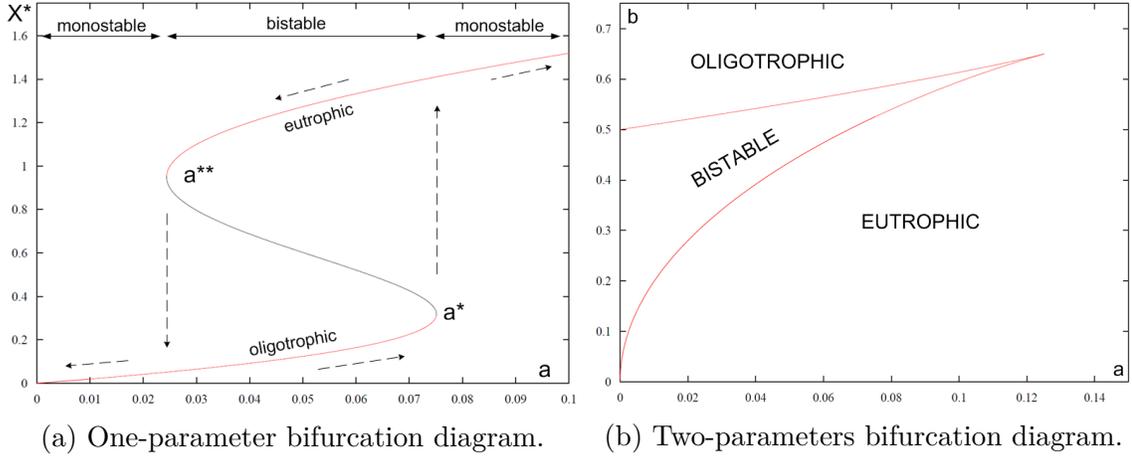
Figure 4.1. Bifurcation diagrams of the lake model.

traced from the one-parameter bifurcation diagram in Figure 4.1a that was generated using XPPAUT [41]. In this plot, one may see that the number of equilibria changes as the value of parameter a is increased. In particular, these changes are also followed by the change on the qualitative behaviors (i.e. stability type) of the equilibria. For a small inflow rate ($0 \leq a \leq 0.025$), the system has a single *oligotrophic* equilibrium (monostable). For larger inflow rate ($0.026 < a \leq 0.075 = a^*$), this equilibrium bifurcates into two stable equilibria: an oligotrophic one and a *eutrophic* one. Finally, for an even larger inflow rate ($a > a^*$), these two stable equilibria coalesce to a single eutrophic equilibrium. Once the system stays in the high P concentration equilibrium, the return to a low P concentration requires a decrease on parameter a down to the critical value $a^{**} \approx 0.025$ due to the hysteresis characteristic of the equilibrium curve shown in the bifurcation diagram. One may further analyze how the equilibria bifurcate when both parameters a and b are varied simultaneously. As can be seen in Figure 4.1b, the two-parameter bifurcation diagram [41] clearly shows the partition of the parameter space into regions where the system has single (oligotrophic or eutrophic) or multiple equilibria (bistable). The curve which encapsulates the bistable region in Figure 4.1b is known as the *bifurcation manifold* and it contains all criti-

cal parameters k^* at which transitions or regime shifts between different qualitative behaviors of the system occur [41]. Thus for any set of nominal parameters k^0, the minimum D_2B is defined as

$$\gamma = \inf_k |k^* - k^0|. \tag{4.1}$$

The computation of γ, however, is generally difficult since the bifurcation manifold is usually not known [41, 36]. In particular, numerical bifurcation analysis illustrated in the above example are limited to systems having at most two or three unknown parameters [41, 92]. For a dynamical system

$$\dot{x}(t) = f(x(t), k), \quad x(t) = x_0, \tag{4.2}$$

whose vector fields depend on parameter k, the bifurcation manifold consists of those critical parameters k^* that satisfy the bifurcation conditions as given in Table 4.1 [92]. The first row of the table shows necessary and sufficient conditions for a Hopf bifurcation to occurs. The transversality condition requires that the partial derivative (with respect to parameter k) of the real part of the characteristic polynomial's roots be not equal to zero. The other transversality conditions in this table are conditions on the various derivatives of the vector field in which w and v are the left and right eigenvectors, respectively, associated with the zero eigenvalue of the Jacobian matrix (see [92] for details). Each of these transversality conditions essentially describes an instance where the system undergoes change on its stability properties at the critical parameter k^*. Prior works have proposed several methods for computing γ in the context of robust stability analysis [88, 106, 110, 154] and voltage collapse problem in power systems [36, 36]. In general, these methods use numerical optimization techniques to search for the minimum γ subject to the constraints that the critical parameter k^* satisfy the condition in Table 4.1. These methods, however, are computationally demanding since the search for the minimum γ requires the computation

TABLE 4.1

NECESSARY CONDITIONS FOR LOCAL BIFURCATONS [92].

Type	Jacobian Eigenvalue	Transversality	
Hopf	simple imaginary pair	$D_k\{Re(s)\} \neq 0$	
Saddle-node	simple 0	$w\left(D_k f\big|_{x^*,k^*}\right) \neq 0,$	$w\left(D_x^2 f\big|_{(v,v)}\right) \neq 0$
Transcritical	simple 0	$w\left(D_k f\big|_{x^*,k^*}\right) \neq 0,$	$w\left(D_{x,k}^2 f\big|_{v,v}\right) \neq 0$
Pitchfork	simple 0	$w\left(D_k f\big|_{x^*,k^*}\right) \neq 0,$	$w\left(D_x^3 f\big|_{x^*,k^*}\right) \neq 0$

of system's equilibria x^* *numerically* for each values of the parameter and at every iteration.

4.1.2 Approach and Contribution

This chapter uses SOS relaxations to compute lower bounds on the minimum $D_2 B$ (4.1) in nonnegative systems with kinetic realizations. A dynamical system of the form (4.2) with polynomial vector fields is said to have a kinetic realization if there exist polynomials $g_i(x,k)$ and $h_i(x,k)$ with nonnegative coefficients such that

$$f_i(x,k) = g_i(x,k) - x_i h_i(x,k).$$

The above condition essentially guarantees that the system is a *mass action systems* for which a chemical reaction network (CRN) graph can be realized [67]. The existence of such realizations implies that the system's vector fields can be decomposed

as $f(x, k) = Nv(x, k)$ where N is a real-valued matrix and $v(x, k)$ is a nonnegative-valued vector of monomials in x. The special structure of these systems allows one to compute an expression for their equilibria in terms of the system's parameters and some convex parameters. Such an equilibrium expression simplifies the D_2B problem because the constraints that define the bifurcation conditions can now be expressed only in terms of the system's parameters, rather than the system's parameters and equilibria. By rewriting the eigenvalue conditions in Table 4.1 as a semialgebraic set in the parameter space, we formulate the D_2B problem as an SOS optimization problem that can be solved using semidefinite programming solvers.

The remainder of this chapter is structured as follows. Section 4.2 presents background on local bifurcation theory. Section 4.3 describes the necessary bifurcation conditions that will be used later in Section 4.4 to recast the D_2B problem as an SOS optimization problem. Section 4.5 illustrates an application of the proposed method. Final remarks are given in Section 4.6.

4.2 Local Bifurcation and D_2B Problem

4.2.1 General Nonlinear Systems

Consider the parameterized dynamical systems in (4.2) and assume for the moment that the system's parameters are fixed to some real numbers. Equation (4.2) may then be written as

$$\dot{x} = f(x), \qquad x(0) = x_0. \tag{4.3}$$

Let $\varphi(t) : \mathbb{R}^n \mapsto \mathbb{R}^n$ be an *evolution operator* that transforms the system's initial state $x_0 \in \mathbb{R}^n$ into a state $x(t) \in \mathbb{R}^n$ at time $t \in [0, T]$ such that

$$x(t) = \varphi(t)x_0.$$

The operator $\varphi(t)$ is known as the *flow* of the system and it characterizes the evolution of the state $x(t)$ of the system at any time t when initialized at x_0 [63]. For a dynamical system with a flow $\varphi(t)$, two basic geometric objects can be associated with it, namely its *orbits* in the state space and its *phase portrait* in the state space formed by these orbits. An orbit that starts at x_0 is an ordered subset of the state space at which the evolution operator $\varphi(t)x_0$ is defined. Examples of orbits include fixed points (equilibria), cycles, etc. The phase portrait is a partitioning of the state space into orbits and thereby provides a topological description about the qualitative dynamics of the system. Two dynamical systems are said to be *topologically equivalent* if their phase portraits are qualitatively similar, namely if there exists a homeomorphism between the orbits of the two phase portraits [63, 92].

The dynamics of a system are usually studied locally in a bounded region $\mathcal{X} \subset \mathbb{R}^n$ of the state space. This is particularly helpful for topological classification of the phase portrait near the equilibrium points because the local behaviors of a system near its equilibrium points can be studied from its linearization [92]. Let x^* be an equilibrium of (4.3) such that $f(x^*) = 0$ and let $J = \frac{\partial f}{\partial x}\big|_{x^*}$ denotes its Jacobian matrix evaluated at the equilibrium x^*. Let n_-, n_0 and n_+ be the number of eigenvalues of J with negative, zero and positive real parts, respectively. An equilibrium is called *hyperbolic* if $n_0 = 0$, that is, if there are no eigenvalues on the imaginary axis [92]. In the neighborhood of a hyperbolic equilibriums x^* of (4.3), the Grobman-Hartman Theorem [92] states that the qualitative dynamic of a nonlinear system (4.3) is *locally topologically equivalent* to its linearization $\dot{\xi} = J\xi$. One can then study the local topological equivalence of dynamical systems by analyzing the local phase portrait of its linearization around the equilibrium points.

Now assume that the vector fields of (4.2) depend on the parameters $k \in \mathbb{R}^p$ and consider the phase portrait of this system. If the values of parameters k are varied in the parameter space, $\mathbb{R}^p$, then the equilibria x^* and the phase portrait of the system

will also vary in the state space, $\mathbb{R}^n$. In particular, the phase portrait of the system may either remain topologically equivalent to the nominal one or it may also change to something else. The appearance of a topologically nonequivalent phase portrait under variation of the parameters is called a *bifurcation*. The values of the parameters at which a bifurcation occurs are called the bifurcation (critical) parameters [92]. Since the behavior of a nonlinear system is locally topologically equivalent with its linearization around a hyperbolic equilibrium, the onset of a bifurcation can be studied using the system's linearized model around that particular equilibrium. In this case, topological equivalence of the system in the presence of parameter variations can be studied by analyzing the impact of such variations on the topology of the system's equilibria [63, 92].

Let x^* be a nominal equilibrium of the systems and let y^* be the equilibrium when the parameter varies. The following theorem provides conditions under which the linearized system is locally topologically equivalent.

Theorem 4.2.1 ([92])**.** *The phase portrait of a system near two hyperbolic equilibria, x^* and y^* are locally topologically equivalent if and only if these equilibria have the same number n_- and n_+ of eigenvalues with negative and positive real part, respectively.*

Theorem 4.2.1 is the basic result from which the necessary conditions in Table 4.1 are constructed [92]. The search for parameter sets that satisfy these conditions, however, is not trivial because it requires knowledge of system's equilibria. In particular, the currently available numerical software tools [34, 41] for bifurcation analysis are limited to analyze at most two parameters simultaneously. As a result, the computation of minimum D_2B in (4.1) for systems that have more than two parameters cannot be done (cf. the review in [36]).

4.2.2 Nonnegative Systems with Kinetic Realizations

The computation of minimum D_2B can be simplified in a class of nonnegative systems that have kinetic realizations. Recall that system (4.2) with polynomial vector fields $f_i(x, k)$ has a *kinetic realization* if there exists polynomials $g_i(x, k)$ and $h_i(x, k)$ with nonnegative coefficients such that the vector fields satisfy condition (**??**). If a system's vector fields satify the condition in (**??**), then the system is a mass action system for which a chemical reaction graph can be realized [67]. In particular, these systems' vector fields satisfy a decomposition of the form

$$\dot{x}(t) = f(x, k) = Nv(x, k), \quad x(0) = x_0, \tag{4.4}$$

where N is a real-valued matrix and $v(x, k)$ is a nonnegative-valued flux vector that satisfies a decomposition $v(x, k) = \mathrm{diag}(k)x^Z$. An important property of systems with this special structure is that it allows one to compute an expression for the system's equilibria in terms of the system's parameters and some convex parameters (cf. Chapter 3).

Nonnegative systems with kinetic realizations exist for a large number of real world systems including compartmental, biological, and ecological systems [65, 64]. The restriction to polynomial systems is not overly restrictive since 1) any function with finite variations that is defined over a bounded interval can be approximated arbitrarily closely with a polynomial [138], 2) systems with rational vector fields can be transformed into polynomial systems [42], and 3) there exist a number of methods for extracting kinetic realizations from polynomial systems [67, 144, 145, 79–81].

As discussed in Chapter 3, the special structure of nonnegative systems with kinetic realizations has two major consequences, namely

(i) any equilibrium flux can be expressed as a function $v^*(\lambda)$ in terms of some convex parameters (λ), and

(ii) any equilibrium state can be expressed as a rational function $x^*(v^*, k)$ of the equilibrium fluxes (v^*) and the system parameters (k).

Using the expression of the equilibrium fluxes v^* in term of the convex parameters, one may compute an algebraic expression for the equilibrium state $x^*(k, \lambda) \in \mathbb{Q}^n(k, \lambda)$ as a rational function of the system's parameters (k) and some convex parameters (λ). This algebraic equation characterizes all system equilibria as a function of the system and convex parameters and it provides a critical starting point for characterizing the bifurcation constraints in the D_2B problem.

Another property of systems with kinetic realizations is that its Jacobian matrix can be parameterized as [57, 58]

$$J(\lambda, k) = N \operatorname{diag}(v^*(\lambda)) Z^T \operatorname{diag}(1/x^*(\lambda, k)). \tag{4.5}$$

This implies that the eigenvalue conditions in Table 4.1 can now be expressed in terms of the parameters k and λ without the need to compute the system's equilibria x^* for different values of k. Earlier works for solving the D_2B problem [35, 36, 15] always required that one solve for the equilibrium as part of the optimization; this approach is computationally expensive as it increases the number of decision variables in the formulation of the optimization problem. In Section 4.3, the Jacobian matrix (4.5) will be used to recast the necessary conditions for the occurrence of bifurcations.

4.3 Necessary Bifurcation Conditions

Consider the Jacobian matrix in (4.5). Let $p(s) = |sI - J|$ be the Jacobian's characteristic polynomial, i.e.

$$p(s) = a_0 s^n + a_1 s^{n-1} + \cdots + a_{n-1} s + a_n, \tag{4.6}$$

where the coefficients $a_i(\lambda, k)$ are functions of the parameters (λ, k). For notational convenience, we denote these parameters as $\mu = (\lambda, k)$. The eigenvalues of J are given by the roots of $p(s)$. The Jacobian J is asymptotically stable if and only if all of its eigenvalues have negative real parts and is unstable otherwise [63]. For $z = 1, \ldots, n$, the zth Hurwitz determinant, $\triangle_z$, associated with characteristic polynomial $p(s)$ is defined as

$$\triangle_z = \begin{vmatrix} a_1 & a_3 & a_5 & \ldots & a_{2z-1} \\ a_0 & a_2 & a_4 & \ldots & a_{2z-2} \\ 0 & a_1 & a_3 & \ldots & a_{2z-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & a_{z-2} & a_z \end{vmatrix}$$

such that

$$\triangle_1 = |a_1|, \quad \triangle_2 = \begin{vmatrix} a_1 & a_3 \\ a_0 & a_2 \end{vmatrix}, \quad \triangle_3 = \begin{vmatrix} a_1 & a_3 & a_5 \\ a_0 & a_2 & a_4 \\ 0 & a_1 & a_3 \end{vmatrix}, \quad \ldots.$$

Proposition 4.3.1 characterizes the conditions for the Jacobian (4.5) to have a simple zero eigenvalue.

Proposition 4.3.1. *Consider matrix J in (4.5) with characteristic polynomial $p(s)$ in (4.6). If the coefficients of $p(s)$ satisfy the conditions $a_n = 0$ and $a_{n-1} \neq 0$, then matrix J will have zero eigenvalue with multiplicity not greater than one.*

Proof. That $a_n = 0$ implies one of the roots of $p(s)$ is zero is clear. Now notice that $p(s)$ will have zero eigenvalue with multiplicity not greater than one if $\frac{\partial p(s)}{\partial s}|_{s=0} \neq 0$, which will be satisfied when $a_{n-1} \neq 0$. $\qquad \square$

The following lemma from [23] characterizes the condition for matrix J to have a simple pair of imaginary eigenvalues. The proof of this lemma is based on the Orlando

formula (cf. [52]).

Lemma 4.3.2 ([23]). *Consider matrix J in (4.5) with characteristic polynomial $p(s)$ in (4.6). If the $(n-1)$th Hurwitz determinant of $p(s)$ satisfies $\triangle_{n-1} = 0$, then matrix J will have a pair of imaginary eigenvalues with multiplicity not greater than one.*

Now let us express the eigenvalue conditions in Table 4.1 in terms of the coefficients of $p(s)$ in (4.6). Let q denote the number of parameters in μ. Let Ω^{SN} be the parameter set where a saddle-node (also pitchfork and transcritical) bifurcation occurs. Based on the conditions in Proposition 4.3.1, then

$$\Omega^{SN} = \left\{ \mu \in \mathbb{R}^q_{\geq 0} \mid a_n(\mu) = 0, \ a_{n-1}(\mu) \neq 0 \right\}. \tag{4.7}$$

In a similar way, Lemma 4.3.2 can be used to characterize the parameter set Ω^H where a Hopf bifurcation occurs.

$$\Omega^H = \left\{ \mu \in \mathbb{R}^q_{\geq 0} \mid \triangle_{n-1}(\mu) = 0 \right\}. \tag{4.8}$$

Equations (4.7)-(4.8) suggest that if a bifurcation occurs, then one may denote the parameter set Ω for which at least one type of bifurcation occurs as

$$\Omega = \Omega^{SN} \cup \Omega^H. \tag{4.9}$$

Note that the sets defined in (4.7)-(4.9) are algebraic sets which characterize those parameters for which a bifurcation may occurs. Equivalently, system (4.4) will *not* undergoes a bifurcation if the set Ω is empty. In the next section, a method for verifying the emptiness of this set will be discussed.

Remark 4.3.3. Note that the analysis in this chapter will only characterize the eigenvalue conditions for the occurrence of a Hopf and a saddle-node bifurcations (the

necessary conditions for the occurrence of pitchfork and transcritical bifurcations are the same as that for the saddle-node bifurcation, cf. Table 4.1). This implies that the semialgebraic descriptions for the transversality conditions in Table 4.1 will not be included in the formulation of the optimization problem for computing the minimum D_2B. Rather than embedding them in the formulated optimization problem, the satisfaction of these transversality conditions can be tested after the set of critical parameters k^* from the optimization is obtained.

4.4 Distance-to-Bifurcation Problem

From the discussions in Section 4.3, it should be clear that the non-existence of a particular bifurcation is equivalent to the emptiness of the corresponding bifurcation set (i.e. the sets Ω^{SN} and Ω^H in (4.7) and (4.8), respectively). In general, verifying the emptiness of the set Ω^{SN}, for example, can be difficult. However in recent years, it has been proven fruitful to consider convex relaxations of this problem in which one checks for the emptiness of the set $\tilde{\Omega}(\gamma) \cap \Omega^{SN}$, where $\tilde{\Omega}(\gamma)$ is a semi-algebraic set defined by a psd *certificate* function $V(\mu)$ [156].

In particular, let $\gamma > 0$ be a real-valued constant and let $\alpha(|\mu - \mu_0|)$ be a class $\mathcal{K}$ function (i.e. α is a continuous, strictly increasing function with $\alpha(0) = 0$) in which μ is the parameter set with known initial μ_0. We define the *certificate* set as

$$\tilde{\Omega}(\gamma) = \left\{ \mu \in \mathbb{R}^q \mid \alpha(|\mu - \mu_0|) \leq \gamma \right\}. \tag{4.10}$$

For a given $\gamma > 0$, if the intersection of the certificate set $\tilde{\Omega}(\beta)$ in (4.10) with the saddle-node bifurcation set Ω^{SN} in (4.7) is empty, then the D_2B cannot be less than $\alpha^{-1}(\gamma)$. The key point in formulating the problem in this way is that the conditions which specify whether the intersection $\tilde{\Omega}(\beta) \cap \Omega^{SN}$ is empty or not can be formulated using SOS relaxations. This fact is formally stated in Proposition 4.4.1 below.

Proposition 4.4.1. *For a constant $\gamma > 0$, let $\tilde{\Omega}(\gamma)$ be a certificate set defined in (4.10). Consider the saddle-node bifurcation set Ω^{SN} in (4.7). If there exist polynomials $V(\mu)$ and $r(\mu)$ such that*

$$a_{n-1}^2(\mu)(V(\mu) - \gamma) + r(|\mu|)a_n(\mu) \quad \text{is SOS}, \tag{4.11}$$

then $\Omega^{SN} \cap \tilde{\Omega}(\gamma) = \varnothing$.

Proof. Verifying the condition $\Omega^{SN} \cap \tilde{\Omega}^{SN} = \varnothing$ amounts to check the emptiness of the set

$$\{ \, \mu \mid a_n = 0, \ a_{n-1} \neq 0, \ V(\mu) - \gamma \neq 0, \ -(V(\mu) - \gamma) \geq 0 \, \}.$$

Using the positivstellensatz theorem [137, 10, see also Chapter 2], this set is empty if there exist SOS polynomials s_0, s_1 and polynomials $V(\mu)$, $t(\mu)$ such that

$$s_0 - s_1(V(\mu) - \gamma) + a_{n-1}^{2m} \left(V(\mu) - \gamma\right)^{2m} + t(\mu)a_n = 0.$$

Setting $s_0 = 0$, $m = 1$, and $t(\mu) = (V(\mu) - \gamma)r(\mu)$, the above equation becomes

$$s_1(V(\mu) - \gamma) = (V(\mu) - \gamma) \left(a_{n-1}^2(V(\mu) - \gamma) + r(\mu)a_n\right),$$

which is the SOS condition in (4.11). Now consider any parameter $\mu \in \Omega^{SN}$ for which the condition $a_n(\mu) = 0$ holds. Upon substitution with the SOS condition in (4.11), we have that

$$a_{n-1}^2(\mu)(V(\mu) - \gamma) \geq 0.$$

Since $a_{n-1}^2 > 0$, we have $V(\mu) - \gamma \geq 0$ which implies that any parameter $\mu \in \Omega^{SN}$ lie outside the level set defined by $V(\mu) \leq \gamma$. $\qquad\square$

In a similar way, an SOS relaxation for the non-existence of a Hopf bifurcation can be obtained as stated in Proposition 4.4.2 below.

Proposition 4.4.2. *For a constant $\gamma > 0$, let $\tilde{\Omega}(\gamma)$ be a certificate set as defined in (4.10). Consider the Hopf bifurcation set Ω^H in (4.8). If there exist polynomials $V(\mu)$, $r(\mu)$ such that*

$$V(\mu) - \gamma + r(\mu)\triangle_{n-1}(\mu) \quad \text{is SOS}, \tag{4.12}$$

then $\Omega^H \cap \tilde{\Omega}(\gamma) = \varnothing$.

Proof. Verifying the condition $\Omega^H \cap \tilde{\Omega}(\gamma) = \varnothing$ amounts to check the emptiness of the set

$$\{\, \mu \mid \triangle_{n-1} = 0, \ V(\mu) - \gamma \neq 0, \ -(V(\mu) - \gamma) \geq 0 \,\}.$$

Using the positivstellensatz theorem, this set is empty if there exist SOS polynomials s_0, s_1 and polynomials $V(\mu)$, $t(\mu)$ such that

$$s_0 - s_1(V(\mu) - \gamma) + (V(\mu) - \gamma)^{2m} + t(\mu)\triangle_{n-1} = 0.$$

Let $s_0 = 0$, $m = 1$, $t(\mu) = (V(\mu) - \gamma)r(\mu)$, then

$$s_1(V(\mu) - \gamma) = (V(\mu) - \gamma)\left[(V(\mu) - \gamma) + r(\mu)\triangle_{n-1}\right],$$

which is the SOS condition in (4.12). Now consider any parameter $\mu \in \Omega^H$ for which the condition $\triangle_{n-1}(\mu) = 0$ holds. Upon substitution with the SOS condition (4.12), we have that $V(\mu) > \gamma$ which implies that any $\mu \in \Omega^H$ will lie outside the level set defined by $V(\mu) \leq \gamma$. $\qquad\square$

Note that Propositions 4.4.1 and 4.4.2 characterize those values of γ for which the associated certificate set $\tilde{\Omega}(\gamma)$ contains no bifurcation parameters. Clearly, if one were to identify the maximum value of γ for which, say, Proposition 4.4.1 held, then this γ could be used to bound the minimum D_2B. In particular, let μ_0 be a known

initial parameter. Define the certificate function as $V(\mu) = \alpha(|\mu - \mu_0|)$ where α is a class $\mathcal{K}$ function. Let $\bar{\gamma}$ denotes the largest real constant for which, say, Proposition 4.4.1 holds. Then the minimum D_2B, γ, can be bounded below as

$$\gamma = |\mu^* - \mu_0| \geq \alpha^{-1}(\bar{\gamma}).$$

One obvious choice for α is to let it be $|\mu - \mu_0|$. This observation suggests that the computation of the minimum γ can now be formulated as a polynomial optimization problem which can be solved through the SOS relaxation method. This is formalized in Poposition 4.4.3 below which is stated for the case of saddle-node bifurcation (cf. Proposition 4.4.1). Clearly a similar result holds for the Hopf bifurcation case.

Proposition 4.4.3. *Consider system (4.4) with Jacobian matrix in (4.5). Let μ_0 be the set of initial parameters and let μ^* denotes the set of critical parameters at which a saddle-node bifurcation occurs. If there exist a constant $\bar{\gamma} > 0$, polynomials $\bar{V} = |\mu^* - \mu_0|$ and $r(\mu)$ such that the following SOS optimization problem*

$$\begin{aligned} max \quad &\bar{\gamma} \\ s.t. \quad &a_{n-1}^2(\mu)(\bar{V}(\mu) - \gamma) + r(\mu)a_n(\mu) \quad \text{is SOS,} \end{aligned}$$

has a feasible solution, then the D_2B is defined as $|\mu^* - \mu_0| \geq \bar{\gamma}$.

Proof. From the assumption in the proposition, we know that a saddle-node bifurcation exists and therefore the set Ω^{SN} is not empty. Now since no bifurcation takes place at μ_0, one can take the infimum of this set, say $\inf(\Omega^{SN})$. Note that the sets $\tilde{\Omega}(\gamma)$ are compact sets, so there exists $\bar{\gamma} = \inf(\Omega^{SN})$ such that for any $\gamma < \bar{\gamma}$ we know from Proposition 4.4.1 that no saddle-node bifurcation occurs. $\square$

Note that the SOS optimization problem in Proposition 4.4.3 can be solved using semidefinite programming solvers [121, 100, 111, 139]. Section 4.5 presents some

examples of the method discussed in this section.

4.5 Examples

This section uses the method discussed in Sections 4.3-4.4 to compute lower bounds on the minimum D_2B in nonnegative systems with kinetic realizations. The first example computes a lower bound on the minimum distance to a Hopf bifurcation in a kinetic realization of the Brusselator system. The second example computes a lower bound on the minimum distance to a saddle-node bifurcation in a kinetic realization of the tritrophic foodweb system. The kinetic realizations of these models were constructed in the examples discussed in Chapter 3.

4.5.1 The Brusselator Model (Hopf Bifurcation)

A kinetic realization of this system is given in (3.17)-(3.18) and the expression for its state equilibria is given in (3.45). Note that this model has four parameters and so the standard numerical bifurcation analysis cannot be used to analyze all parameters simultaneously.

The Jacobian matrix (4.5) for this system is given by

$$
\begin{aligned}
J(\lambda, h) &= N \mathrm{diag}(E\lambda) Z^T \mathrm{diag}(h), \\
&= \begin{bmatrix} (\lambda_1 - \lambda_2)h_1 & \lambda_1 h_2 \\ -\lambda_1 h_1 & -\lambda_1 h_2 \end{bmatrix},
\end{aligned}
$$

with $h_i = (1/x_i^*)$ and x_i^* $(i = 1, 2)$ are given in (3.45). The characteristic polynomial of J is given by

$$
p(s) = s^2 + (\lambda_1 h_2 + \lambda_2 h_1 - \lambda_1 h_1)s + \lambda_1 \lambda_2 h_1 h_2.
$$

Using the expression for the state equilibria in (3.45), then polynomial $p(s)$ can be

rewritten in terms of the system's parameters below

$$p(s) = s^2 + (k_2 - k_1 - k_3)s + k_1 k_3.$$

It can be seen that polynomial $p(s)$ will have a simple pair of imaginary eigenvalues if the following conditions are satisfied

$$k_1 k_3 > 0$$

$$k_2 - k_1 - k_3 = 0.$$

Since $k_i \geq 0$, $(i = 1, \ldots, 4)$, the first condition is automatically satisfied and so the eigenvalue condition for a Hopf bifurcation can be reduced to a single algebraic equation: $k_2 - k_1 - k_3 = 0$. Now define $F(k) = \Sigma_{i=1}^4 (k_i^* - k_i^0)^2$ where k_i^0 denotes the initial/nominal parameters. A lower bound on the minimum $D_2 B$ may then be obtained by solving the following SOS optimization problem.

$$\begin{aligned} \max \quad & \gamma, \\ \text{s.t.} \quad & F(k) - \gamma - \sigma(k)(k_2 - k_1 - k_3) \quad \text{is SOS.} \end{aligned}$$

Let us consider, for example, an initial set of parameters, $k_i^0 = 1$ for $i = (1, \ldots, 4)$, and initial states, $x_1(0) = x_2(0) = 1$. As shown in Figure 4.2a, these initial state and parameter values cause the system to have an asymptotically stable equilibrium at $x_1^* = x_2^* = 1$. We used SOSTOOLS [121] to solve the above SOS optimization problem and found a minimum $D_2 B$ $\gamma = 0.33$ that corresponds to a set of critical parameters $k^* = [2/3, 4/3, 2/3, 1]$. Figure 4.2b plots the state trajectories for this k^* and shows that a Hopf bifurcation occurs.

(a) Initial parameter k^0.

(b) Critical parameter k^*.
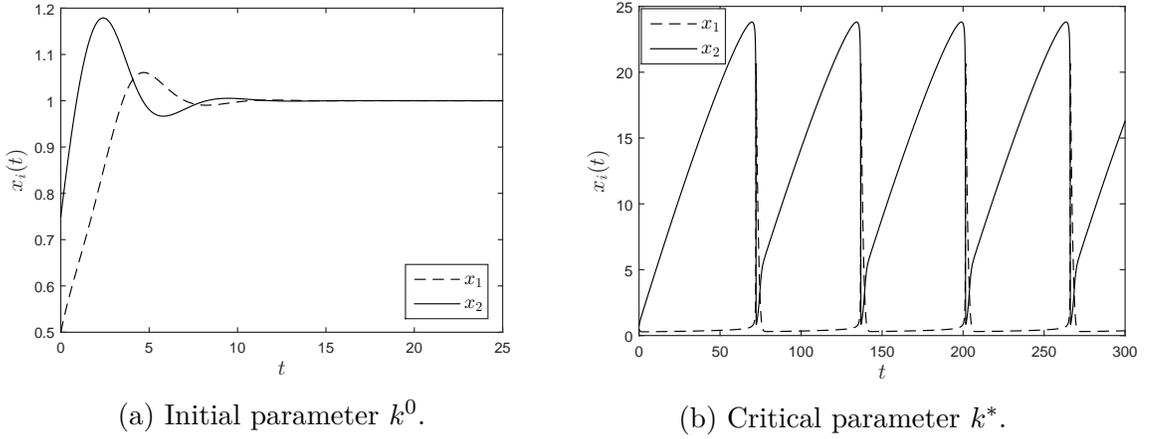
Figure 4.2. Hopf bifurcation in the Brusselator model.

4.5.2 Tritrophic Foodweb Model (Saddle-node Bifurcation)

A kinetic realization of this system is given in (3.23)-(3.24) and the expression of its state equilibria is given in (3.49). The Jacobian matrix (4.5) is given by

$$
\begin{aligned}
J(\lambda, h) &= N \mathrm{diag}(E\lambda) Z^T \mathrm{diag}(h) \\
&= \begin{bmatrix}
-h_1\lambda_1 & -h_2\lambda_6 & 0 & -h_4\lambda_6 \\
h_1(\lambda_4 + \lambda_5) & 0 & -h_3\lambda_5 & h_4(\lambda_4 + \lambda_5) \\
0 & h_2\lambda_3 & 0 & 0 \\
h_1\lambda_2 & h_2\lambda_7 & 0 & h_4\lambda_7
\end{bmatrix}
\end{aligned}
$$

with $h_i = (1/x_i^*)$ and x_i^* $(i = 1, \ldots, 4)$ are given in (3.49). The Jacobian's characteristic polynomial is given by

$$
p(s) = s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0,
$$

where

$$a_0 = h_1 h_2 h_3 h_4 \lambda_3 \lambda_5 (\lambda_2 \lambda_6 - \lambda_1 \lambda_7),$$

$$a_1 = h_2 [h_3 \lambda_3 \lambda_5 (h_1 \lambda_1 - h_4 \lambda_7) - h_1 h_4 (\lambda_1 \lambda_4 \lambda_7 - \lambda_2 \lambda_4 \lambda_6 + \lambda_1 \lambda_5 \lambda_7 - \lambda_2 \lambda_5 \lambda_6)],$$

$$a_2 = h_1 h_2 \lambda_6 (\lambda_4 + \lambda_5) - h_1 h_4 (\lambda_1 \lambda_7 + \lambda_2 \lambda_6) + h_2 h_3 \lambda_3 \lambda_5 - h_2 h_4 \lambda_7 (\lambda_4 - \lambda_5)$$

$$a_3 = h_1 \lambda_1 - h_4 \lambda_7.$$

Using the condition in Proposition 4.3.1, the eigenvalue condition for a saddle-node bifurcation to occur is

$$a_0 = h_1 h_2 h_3 h_4 \lambda_3 \lambda_5 (\lambda_2 \lambda_6 - \lambda_1 \lambda_7) = 0.$$

Since h_i, $i = 1, \ldots, 4$ and λ_j, $j = 1, \ldots, 7$ are always nonnegative, the above condition can be simplified into

$$\lambda_2 \lambda_6 - \lambda_1 \lambda_7 = 0. \tag{4.13}$$

As the above condition indicates, the saddle-node bifurcation in this system's kinetic realization depends only on the convex parameters $\lambda_1, \lambda_2, \lambda_6$ and λ_7. The SOS optimization problem for computing a lower bound on the minimum distance to a saddle-node bifurcation may then be formulated as

$$
\begin{aligned}
\max \quad & \gamma \\
\text{s.t.} \quad & F(\lambda) - \gamma - \sigma(\lambda)(\lambda_2 \lambda_6 - \lambda_1 \lambda_7) \quad \text{is SOS,}
\end{aligned}
\tag{4.14}
$$

with $F(\lambda) = (\lambda_1^* - \lambda_1^0)^2 + (\lambda_2^* - \lambda_2^0)^2 + (\lambda_6^* - \lambda_6^0)^2 + (\lambda_7^* - \lambda_7^0)^2$. Let us consider a set of nominal parameters $k^0 = [0.5, 0.5, 0.5, 0.1, 0.25, 0.25, 0.2]$. The set of convex parameters that correspond to k^0 is $\lambda^0 = [\lambda_1^0, \lambda_2^0, \lambda_6^0, \lambda_7^0] = [0.453, 0.15, 0.2, 0.31]$. Figure 4.3 plots the state trajectories of the nominal system when initialized from two different initial conditions, namely $x_0 = [0.1, 0.1, 0.1]^T$ (Figure 4.3a) and $x_0 =$

(a) Initial condition: $x_0 = [0.1, 0.1, 0.1]$. (b) Initial condition: $x_0 = [0.5, 0.5, 0.5]$.
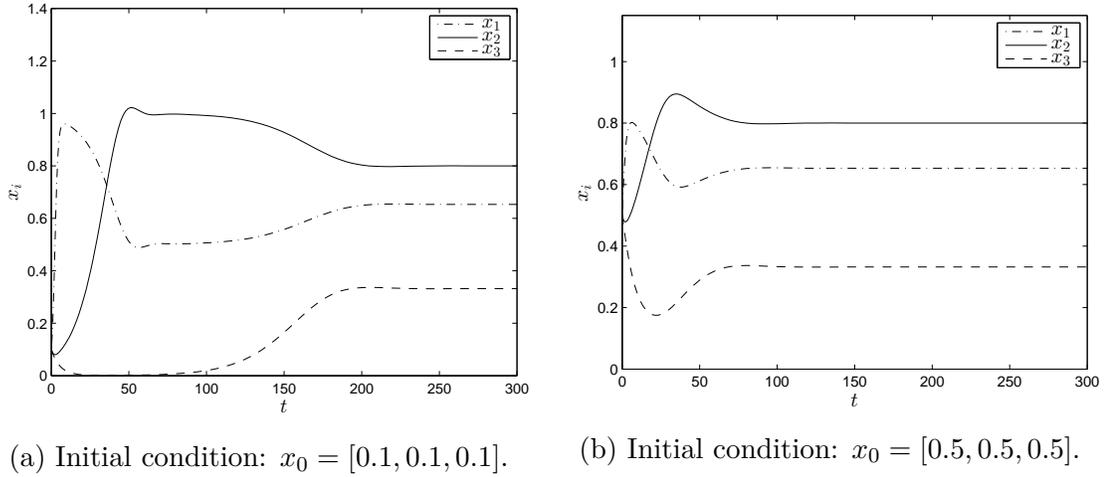
Figure 4.3. Trajectories for k^0 using two different initial conditions.

$[0.5, 0.5, 0.5]^T$ (Figure 4.3b). Note in Figure 4.3 that the trajectories for both initial conditions go to the same stable equilibrium $x^* = [0.65, 0.8, 0.33]^T$.

We solved the optimization problem in (4.14) using SOSTOOLS [121] and found a minimum value $\gamma(\lambda) = 0.0374$ for a set of critical convex parameters $\lambda^* = [0.393, 0.246, 0.283, 0.177]$. The set of critical system's parameters that corresponds to λ^* is $k^* = [0.411, 0.255, 0.5, 0.15, 0.26, 0.25, 0.2]$. Figure 4.4 plots the state trajectories for this k^* using two different initial conditions mentioned previously. The plots in Figure 4.4 show that the state trajectories follow two different qualitative behaviors (i.e. steady states) when initialized from different initial conditions. This suggests that the corresponding saddle-node bifurcation leads to the birth of alternative equilibria in the system's phase portrait.

4.6   Discussions and Remarks

This chapter has presented a method to predict bifurcation-induced regime shifts by solving the minimum D_2B problem. We formulated a polynomial optimization problem that can be used to compute a lower bound on the minimum D_2B in non-

94

(a) Initial condition: $x_0 = [0.1, 0.1, 0.1]$.   (b) Initial condition: $x_0 = [0.5, 0.5, 0.5]$.
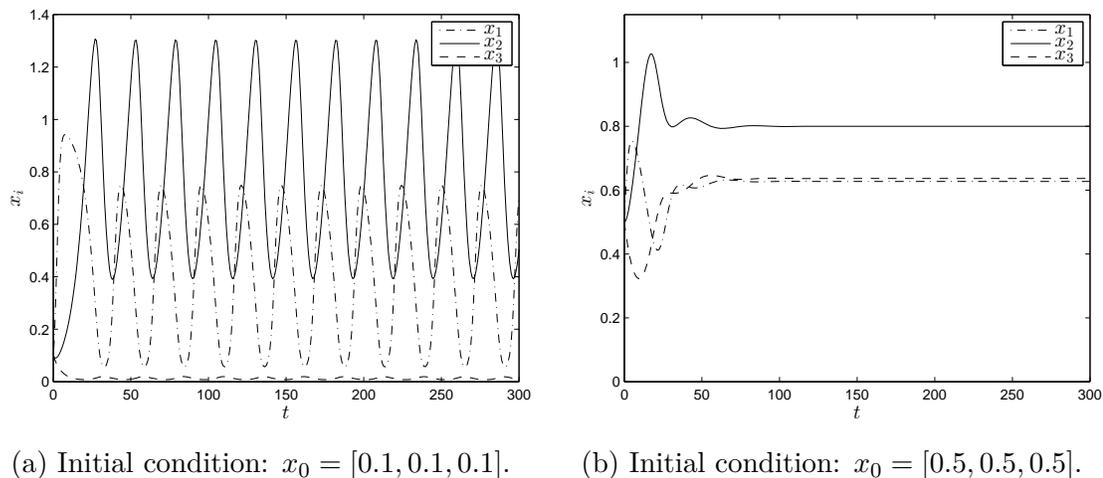
Figure 4.4. Trajectories for k^* using two different initial conditions.

negative systems with kinetic realizations. Example applications of the proposed method illustrate its effectiveness in analyzing local bifurcations caused by simultaneous variations in several parameters.

CHAPTER 5

FORECASTING NOISE-INDUCED REGIME SHIFTS

5.1   Introduction

This chapter presents the method to characterize noise-induced regime shifts. Recall from Chapter 1 that noise-induced regime shifts are defined using stochastic differential equation models of dynamical systems. In particular, such shifts occur because the underlying drift part of the system has multiple stable equilibria and perturbations from external noise push the system's sample paths from the region of attraction (ROA) of one stable equilibrium to the ROA of the alternative equilibria. One important characteristic that distinguishes noise-induced regime shifts from bifurcation-induced regime shifts is that perturbations that trigger the shifts do not change the phase portrait or the number of equilibria of the drift part of the nominal system. This chapter presents two probabilistic quantities that can be used to predict the onset of noise-induced regime shifts. These quantities are the *mean first passage time* (MFPT) and the *finite time safety probability*. The MFPT quantifies the average time required by the process' sample paths to cross the boundary of an ROA. The finite time safety probability characterizes the probability that, starting from the ROA of a stable equilibrium, the process' sample paths eventually reach the boundary of and leave that ROA in a finite time. We show how the computation of these quantities can be formulated and solved using SOS relaxation method (cf. Chapter 2).

5.1.1 Background and Prior Work

To motivate the discussion, consider the stochastically perturbed lake eutrophication model from Chapter 1. The stochastic differential equation (SDE) model of the system is defined as

$$dx(t) = f(x)dt + g(x)dw(t),$$
$$= \left(a - bx + \frac{x^2}{1 + x^2} \right) dt + \sigma dw(t), \tag{5.1}$$

In equation (5.1), $x(t)$ denotes the state of the random process $\{x(t)\}$, $f(x)$ and $g(x)$ are the *drift* and *diffusion* terms, respectively, and $\{w(t)\}$ is a Wiener process with a constant variance σ. The scalar SDE (5.1) can be written as a *stochastic gradient system* of the form [53]

$$dx = -\nabla V(x)dt + \sigma dw(t),$$

where $\nabla V(x) = \frac{dV(x)}{dx}$ and $V(x) = -\int f(x)dx$ is a *potential function* governing the drift of the system. This means that the system's state x can be viewed as a particle moving in a potential landscape defined by $V(x)$. For equation (5.1), this potential function is defined as

$$V(x) = -\int \left(a - bx + \frac{x^2}{1 + x^2} \right) dx = \frac{bx^2}{2} + \arctan(x) - (a + 1)x.$$

Let us consider the values $a = 0.06$ and $b = 0.525$ for the parameters. For this choice of parameters, the drift term has two stable equilibria (oligotrophic and eutrophic states) that are separated by one unstable equilibrium. Figure 5.1 plots the equilibria and the phase portrait of the drift term.

Figure 5.2a plots the potential $V(x)$ for the chosen parameters. This plots shows that $V(x)$ has two local minima (at $x_L^* = 0.16$ and $x_H^* = 1.33$) that are separated by one local maxima (at $x_U^* = 0.52$). In particular, the local minima x_L^* and x_H^* corre-
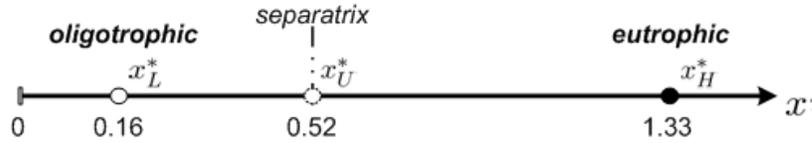
Figure 5.1. Phase portrait of the lake eutrophication model.
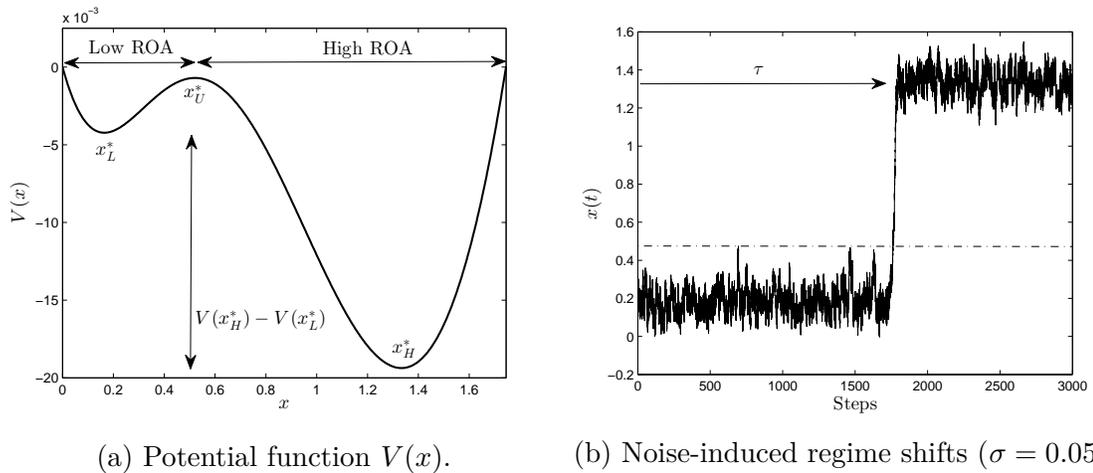
(a) Potential function $V(x)$.

(b) Noise-induced regime shifts ($\sigma = 0.05$).

Figure 5.2. Potential function and sample path of lake eutrophication model.

spond to the two stable equilibria of the system whereas the maxima x_U^* corresponds to the unstable equilibrium. The potential $V(x)$ also defines the ROAs 'Low ROA' and 'High ROA' of the stable equilibria x_L^* and x_H^*, respectively. These two ROAs are separated by a *separatrix* defined by the unstable equilibrium x_U^*. As a result, any trajectory of the drift term that starts from one of the ROAs will remain in that ROA and eventually settle to the equilibrium point of that ROA.

In the presence of the noise process $\{w(t)\}$, the system's sample paths can no longer be guaranteed to stay inside a particular ROA. In other words, *there is a positive probability* that the process' sample paths will reach the separatrix (indicated by dashed line in Figure 5.2b) of the competing ROAs *in a finite time*. Once the

sample path gets closer to the separatrix, the noise process may force the sample path to cross the separatrix and then reach the ROA of an alternative equilibrium. Figure 5.2b illustrates such *noise-induced* regime shifts when the noise process has a variance of $\sigma = 0.05$. This example illustrates that even small noise intensity may drive the process' sample paths to shift from the ROA of equilibrium x_L^* to the ROA of an alternative equilibrium x_L^*.

The statistics of process $\{x(t)\}$ can be studied from its probability density function, $\varphi(x,t)$, whose evolution satisfies the *Fokker-Planck* (FP) equation [54]

$$\frac{\partial \varphi(x,t)}{\partial t} = -\frac{\partial}{\partial x}\left(f(x)\varphi(x,t)\right) + \frac{1}{2}\frac{\partial^2}{\partial x^2}\left(\sigma^2\varphi(x,t)\right).$$

For a scalar stochastic gradient system as in (5.1), the corresponding FP equation can be solved explicitly. This solution can then be used to compute the average time, τ, required by $\{x(t)\}$ to reach the point x_U^* when initialized in the neighborhood of point x_L^* (i.e. MFPT). In the above example, the quantity τ is given by [54]

$$\tau(x_L^* \to x_U^*) = \frac{\pi}{(|\nabla^2 V(x_U^*)|\,\nabla^2 V(x_L^*))^{1/2}} exp\left\{\frac{V(x_U^*) - V(x_L^*)}{\sigma^2}\right\}.$$

The time τ would then indicates the expected time at which the system undergoes a regime shift. One may also compute the probability that the sample paths of $\{x(t)\}$ starting in the ROA $\mathcal{X}_L$ of the equilibrium point x_L^* will eventually leave that ROA through its boundary, $\partial \mathcal{X}_L$, in a finite time $t \leq \tau(x \notin \mathcal{X}_L)$. For the lake model in (5.1), this probability is given by [53]

$$\mathbb{P}\{x(t) \notin \mathcal{X}_L | x(0) \in \mathcal{X}_L\} = \frac{\int_{x_L^*}^{\partial \mathcal{X}_U} \psi(x)dx}{\int_0^{\partial \mathcal{X}_U} \psi(x)dx}, \quad \text{where} \quad \psi(x) = exp\left(\int_0^{x_L^*} \frac{-\nabla V(x)}{\sigma^2}dx\right),$$

which can be used to characterize the likelihood of regime shifts occurring.

The above example suggests that the characterization of noise-induced regime

shifts can be formulated either as an MFPT problem or a stochastic safety analysis as stated below.

- *MFPT problem:* Let $\{x(t)\}$ be a stochastic process whose state $x(t)$ at time $t \geq 0$ takes values on a bounded open subset $\mathcal{X} \subseteq \mathbb{R}^n$ of the Euclidean space with smooth boundary $\partial \mathcal{X}$. Let $\mathcal{X}_0 \subset \mathcal{X}$ be an initial set such that $x(0) = x_0 \in \mathcal{X}_0$. The time at which the sample paths of $\{x(t)\}$ hit the set $\partial \mathcal{X}$ is a random variable τ called the first passsage time and is defined as

$$\tau \equiv \inf_t \{t \geq 0 \,|\, x(t) \in \partial \mathcal{X} \}. \tag{5.2}$$

  Thus, the MFPT problem concerns with the computation of the expected value $\mathbb{E}\{\tau\}$ of τ.

- *Stochastic safety analysis:* Let $\{x(t)\}$ be a stochastic process whose state $x(t)$ at time $t \geq 0$ takes values on a bounded open subset $\mathcal{X} \subseteq \mathbb{R}^n$ of the Euclidean space. Let $\mathcal{X}_0 \subset \mathcal{X}$ be an initial set such that $x(0) = x_0 \in \mathcal{X}_0$ and let $\mathcal{X}_s \subseteq \mathcal{X}$ denote an arbitrary safe/desired set within the state space such that $\mathcal{X}_0 \subset \mathcal{X}_s$. The finite time stochastic safety analysis characterizes the probability that, starting from inside the initial set $\mathcal{X}_0$, the sample paths $x(t)$ of the process leave the safe set $\mathcal{X}_s$, at least once, in a finite time $t \in [0, T]$. Formally, this analysis seeks to compute $\beta \in [0, 1]$ such that

$$\mathbb{P}\{x(t) \notin \mathcal{X}_s, \text{ for some } 0 \leq t \leq T \,|\, x(0) \in \mathcal{X}_0\} \leq \beta. \tag{5.3}$$

Note that the method used for solving these two problems in the lake model (5.1) does not scale up to systems with dimensionality greater than one. This is because the method requires analytical solutions to the corresponding FP equation. It is widely acknowledged that solving the FP equation for systems with dimensionality greater than one is generally intractable as it involves solving a set of partial differential equations with appropriate boundary conditions [125, 49, 53]. As a result, stochastic simulations using Monte Carlo (MC) methods [89] have become the commonly used approach to compute (5.2)-(5.3). However, the MC method is also computationally expensive as it requires exhaustive simulations of the process' sample paths.

5.1.2 Approach and Contribution

One alternative approach to approximate the quantitites (5.2)-(5.3) is using the Lyapunov-like methods in stochastic stability analysis [122, 91]. An important feature of this method is that it does not rely on the solution of the FP equation or require exhaustive simulations through MC methods. This method essentially searches for a positive semidefinite function $V(x(t))$, called a *barrier certificate*, from which the upper bounds for (5.2)-(5.3) can be deduced. In particular, if the drift and diffusion terms of the process's model are polynomial functions, then the computation of these bounds can be formulated and solved using SOS relaxation method [122]. This approach was recently used in [122] for solving the stochastic reachability problem in systems that are driven by simple Brownian motion.

One should realize that many cases of noise-induced regime shifts occur due to the extreme or abnormal events that cause jumps or discontinous changes on the system's states. Examples of these events include storms that wash organisms out of the lakes or rivers [132], abnormal variations of stock prices that lead to market crash [3] or natural disaster and human exploitation that destroy an ecosystem [132]. These events are no longer suitable to be modeled by a Wiener process but are better characterized as a stochastic renewal process. Forecasting regime shifts that are induced by these 'shock' noises is also important and so it is valuable to extend the basic approach in [91, 122] to systems that are modeled as jump diffusion processes.

This chapter presents an extension of the methods introduced in [91, 122] to compute upper bounds for (5.2)-(5.3) in systems that are modeled as jump diffusion processes. As in the case of diffusion processes [91, 122], the techniques discussed in this chapter are also based on searching for a barrier certificate, $V(x(t))$, that generates a supermartingale from which the bounds can be deduced. The main contribution in this chapter is a polynomial characterization of the jump diffusion process' infinitesimal generators that allows the use of SOS optimization techniques

to search for the appropriate barrier certificate $V(x(t))$.

The remainder of this chapter is structured as follows. Section 5.2 gives background on jump diffusion processes. Section 5.3 derives upper bounds for (5.2)-(5.3) in systems that are modeled as jump diffusion processes. The formulation of a SOS optimization problem for computing these bounds is also discussed. Section 5.5 illustrates example using the proposed method for managing fish population in freshwater ecosystems. Section 5.6 gives remarks and suggestions for future works.

Notational Conventions: Let $\{x(t)\}$ denotes a random process whose state $x(t) \in \mathcal{X}$ at time $t \in \mathbb{R}_+$ takes values in an open subset $\mathcal{X} \subseteq \mathbb{R}^n$ of the Euclidean space. $\mathbb{E}\{\cdot\}$ and $\mathbb{E}\{\cdot|\cdot\}$ denote the total and conditional expectations of a random variable, respectively, while $\mathbb{P}\{\cdot\}$ and $\mathbb{P}\{\cdot|\cdot\}$ denote the total and conditional probabilities of an event, respectively. If $\{x(t)\}$ has a distribution $F(x)$, its nth moment is $\mathbb{M}_x^n = \int x^n dF(x)$.

The binomial coefficient of n-dimensional multi-indices α and β is defined as

$$\binom{\alpha}{\beta} = \binom{\alpha_1}{\beta_1} \cdots \binom{\alpha_n}{\beta_n} = \frac{\alpha!}{\beta!(\alpha - \beta)!}.$$

Given n-dimensional vectors $x, y \in \mathbb{R}^n$ and n-dimensional multi-indices α and β, the multi-index binomial theorem states

$$(x + y)^{[\alpha]} = \sum_{0 \leq \beta \leq \alpha} \binom{\alpha}{\beta} x^{[\alpha - \beta]} y^{[\beta]}.$$

It can be shown that

$$\partial^{[\alpha]} x^{[\beta]} = \begin{cases} \frac{\beta!}{(\beta - \alpha)!} x^{[\beta - \alpha]}, & \text{if } \alpha \leq \beta, \\ 0, & \text{otherwise.} \end{cases}$$

Given a bounded real-valued function $V(x) : \mathbb{R}^n \to \mathbb{R}$ and an n-dimensional multi-

index α, the αth order partial derivative of V with respect to x is defined as $\partial^{[\alpha]}V =$
$\frac{\partial^{\alpha_1}V}{\partial x_1^{\alpha_1}} \frac{\partial^{\alpha_2}V}{\partial x_2^{\alpha_2}} \cdots \frac{\partial^{\alpha_n}}{\partial x_n^{\alpha_n}}$.

5.2 Jump Diffusion Processes

Let $\{\Omega, \mathcal{F}, \mathbb{P}\}$ be a complete probability space with filtration $\{\mathcal{F}_t\}_{t \geq 0}$ that satisfies the usual conditions (cf. [124, Ch. I.5]): (i) $\mathcal{F}_t$ contains $\mathbb{P}-$negligible sets for all t, (ii) $\mathcal{F}_t$ is right continuous, i.e. $\mathcal{F}_{t+} = \mathcal{F}_t$, $\forall t$ (i.e. the totality of information is observable by time t). Consider an adapted JDP $\{x(t)\}$

$$dx(t) = f(x(t))dt + \sigma(x(t))dw(t) + dJ(t), \quad x(0) = x_0, \tag{5.4}$$

where $f(\cdot) : \mathbb{R}^n \to \mathbb{R}^n$ and $\sigma(\cdot) : \mathbb{R}^n \to \mathbb{R}^n$ are Lipschitz continuous functions with linear growth, $\{w(t)\}$ is a Wiener process, $\{J(t)\}$ is a shot noise process defined as [124]

$$J(t) = \sum_{\ell=1}^{N(t)} y_\ell e^{-\delta(t-\tau_\ell)}, \quad \ell \in \mathbb{Z}_+. \tag{5.5}$$

In equation (5.5), $N(t)$ is a Poisson process with intensity ρ, $\{\tau_\ell\}$ are the event times of a Poisson jump, $\{y_\ell\}$ is an i.i.d. random process with distribution $F(y)$ describing the ℓ-th jump's size, and δ is a real positive constant representing the rate of exponential decay after a jump. The JDP in (5.4) is understood in Itô's sense and processes $\{w(t)\}$ and $\{J(t)\}$ are assumed to be independent from each other.

Let $Y(\tau_\ell, y_\ell) = y_\ell e^{\delta \tau_\ell}$, then $J(t)$ in (5.5) may be written as

$$J(t) = e^{-\delta t} \int_0^t \int_{\mathbb{R}^n} Y(\tau, y) N(d\tau, dy), \tag{5.6}$$

where $N(d\tau, dy)$ is a Poisson random measure with $\mathbb{E}\{N(d\tau, dy)\} = \rho dt F(dy)$ and $F(dy)$ is the distribution of jump's size. We define the increment of $J(t)$ as $dJ(t) = J(t + dt) - J(t)$ where dt is an infinitesimal time increment. Using (5.6) to expand

out $dJ(t)$ and retaining the first order terms in dt, the jump process increment can be written as

$$dJ(t) = -\delta J(t)dt + \int_{\mathbb{R}^n} yN(dt, dy), \tag{5.7}$$

where the second term in (5.5) is known as a compound Poisson process [124]. Using the expression for the jump increment in (5.5), the JDP in (5.4) can be rewritten as

$$dx(t) = (f(x(t)) - \delta J(t))\, dt + \sigma(x(t))dw(t) + \int_{\mathbb{R}^n} yN(dt, dy), \quad x(0) = x_0. \tag{5.8}$$

Since $\{J(t)\}$ and $\{w(t)\}$ in (5.7) are independent Markov processes and by the assumed conditions on the filtration $\mathcal{F}_t$, one may conclude that the solution of the JDP in (5.8) is a Markov process with right continuous sample paths (cf. [124]).

Now consider a Markov process $\{x(t)\}$ with right continuous sample paths and consider any function $V(x(t)) : \mathbb{R}^n \to \mathbb{R}$ that generates some statistics of $\{x(t)\}$. The (infinitesimal) *generator* of $\{x(t)\}$ is an operator, $\mathcal{L}$, whose action on $V(x(t))$ is defined by

$$\mathcal{L}V(x(t)) = \lim_{h \downarrow 0_+} \frac{\mathbb{E}\{V(x(h))|V(x_0)\} - V(x_0)}{h} \qquad \text{(if the limit exists)},$$

where $\downarrow$ means that the limit is taken from the right. For a diffusion process $\{x(t)\}$ that satisfies stochastic differential equation $dx(t) = f(x(t))dt + \sigma(x(t))dw(t)$ and a function $V(x(t)) \in C^2(\mathbb{R}^n)$ that is twice continuously differentiable in x and bounded for all $x \in \mathbb{R}^n$ (denote this class of functions as $C^2(\mathbb{R}^n)$), its generator, $\mathcal{L}_{DP}$, is given by [128]

$$\mathcal{L}_{DP}V(x(t)) = \frac{\partial V(x(t))}{\partial x}f(x(t)) + \frac{1}{2}\text{Tr}\left(\sigma^T(x(t))\frac{\partial^2 V(x(t))}{\partial x^2}\sigma(x(t))\right). \tag{5.9}$$

For the jump process $\{x(t)\}$ in (5.8) and a function $V(x(t)) \in C^2(\mathbb{R}^n)$, one can show

that its generator $\mathcal{L}_{JP}$ is defined as [148, 115]

$$\mathcal{L}_{JP}V(x(t)) = \rho \int_0^\infty \left(V(x+y) - V(x)\right) dF(y) - \frac{\partial V(x(t))}{\partial x} \delta J(x). \qquad (5.10)$$

Combining the above generators of diffusion and jump processes, one may conclude that the generator, $\mathcal{L}$, of the JDP in (5.8) is given by

$$\begin{aligned}
\mathcal{L}V(x(t)) &= \frac{\partial V(x(t))}{\partial x}\left(f(x(t)) - \delta J(t)\right) + \frac{1}{2}\mathrm{Tr}\left(\sigma^T(x(t))\frac{\partial^2 V(x(t))}{\partial x^2}\sigma(x(t))\right) \\
&\quad + \rho \int_0^\infty (V(x+y) - V(x))dF(y). \qquad (5.11)
\end{aligned}$$

Dynkin's formula for JDP in (5.8) which can be used to characterize a supermartingale $V(x(t))$ is stated below.

Lemma 5.2.1 ([115]). *Consider the JDP in (5.8) defined on a bounded open set $\mathcal{X} \subseteq \mathbb{R}^n$ with smooth boundary $\partial\mathcal{X}$. Let $\tau := \inf\{t \in \mathbb{R}_+ \mid x(t) \in \partial\mathcal{X}\}$ with $\tau < \infty$ be a stopping time. For $V(x(t)) \in C^2(\mathbb{R}^n)$, suppose that $\mathbb{E}\left\{|V(x(\tau))| + \int_0^\tau |\mathcal{L}V(x(s))|ds\right\} < \infty$. Then*

$$V(x(\tau)) = V(x_0) + \int_0^\tau \mathcal{L}V(x(s))ds. \qquad (5.12)$$

Now recall that a process $\{V(x(t))\}$ is said to be a supermartingale with respect to the filtration $\{\mathcal{F}_t\}_{t\geq 0}$ generated by process $\{x(t)\}$ if: (i) $\forall t \geq 0$, $V(x(t))$ is $\mathcal{F}_t$-measurable, (ii) $\mathbb{E}\{|V(x(t))|\} < \infty$, and (iii) $\mathbb{E}\{V(x(t_2))|V(x(t_1))\} \leq V(x(t_1))$ for $0 \leq t_1 \leq t_2 \leq \tau$ (cf. [128]). For a function $V(x(t)) \in C^2(\mathbb{R}^n)$ with $x(t) \in \mathcal{X}$ that takes values in a bounded open set $\mathcal{X} \subseteq \mathbb{R}^n$, it is known that the generated $\{V(x(t))\}$ is a martingale with respect to $\{x(t)\}$ and satisfies conditions (i) and (ii) (cf. [124, Ch. IV]). If $V(x(t))$ also satisfies $\mathcal{L}V(x(t)) \leq 0$, $\forall x \in \mathcal{X}$ with JDP's generator $\mathcal{L}V(x(t))$ is as defined in (5.11), then Dynkin's formula in (5.12) implies $\{V(x(t))\}$ also satisfies condition (iii). One may then conclude that $V(x(t)) \in C^2(\mathbb{R}^n)$ with $\mathcal{L}V(x(t)) \leq 0$, $\forall x(t) \in \mathcal{X}$ generates a supermartingale with respect to $\{x(t)\}$. In this note, we will

consider nonnegative supermartingale for which the following inequality holds.

Lemma 5.2.2 ([90]). *Let $\{V(x(t))\}$ be a supermartingale with respect to the process $\{x(t)\}$ where $x(t) \in \mathcal{X} \subseteq \mathbb{R}^n$ and $0 \leq t \leq \tau := \inf\{t : x(t) \notin \mathcal{X}\}$. Let $V(x(t))$ be nonnegative in $\mathcal{X}$. Then for a constant $\theta > 0$ and any initial condition $x_0 \in \mathcal{X}$,*

$$\mathbb{P}\left\{\sup_{0 \leq t \leq \tau} V(x(t)) \geq \theta \,\middle|\, x(0) = x_0\right\} \leq \frac{V(x_0)}{\theta}. \tag{5.13}$$

5.3 Upper Bounds of MFPT and Safety Probability

Using the generator in (5.11), Dynkin's formula in (5.12), and the supermartingale inequality in (5.13), we now present methods to compute upper bounds for (5.2)-(5.3) in JDP (5.8). The technique used in these computational methods is inspired by the Lyapunov-like techniques in stochastic stability analysis of Markov processes [90].

5.3.1 Upper Bound for MFPT

Proposition 5.3.1 below characterizes an upper bound $\theta \geq 0$ for the MFPT of JDP (5.8).

Proposition 5.3.1. *Consider the JDP in (5.8) defined on a bounded open subset $\mathcal{X} \subset \mathbb{R}^n$ with smooth boundary $\partial\mathcal{X}$. Let the initial condition $x(0) = x_0$ be a random variable that takes values in $\mathcal{X}_0 \subset \mathcal{X}$. If there exists a real-valued function $V(x(t)) \in C^2(\mathbb{R}^n)$ and a constant $\theta \geq 0$ such that*

$$
\begin{aligned}
V(x(t)) &\geq 0, &\forall x \in \mathcal{X}, \\
V(x(t)) &\leq 0, &\forall x \in \partial\mathcal{X}, \\
V(x(t)) &\leq \theta, &\forall x \in \mathcal{X}_0, \\
\frac{\partial V(x(t))}{\partial t} + \mathcal{L}V(x(t)) &\leq -1, &\forall x \in \mathcal{X},
\end{aligned}
$$

where $\mathcal{L}V(x(t))$ is defined in (5.11), then $\mathbb{E}\{\tau\} \leq \theta$ with $\tau = \inf_t\{t \geq 0 : x(t) \in \partial\mathcal{X}\}$.

Proof. The technique of the proof is similar to that in [122]. Itô's lemma provides a stochastic differential equation for $V(x(t))$

$$dV(x(t)) = \left(\frac{\partial V(x(t))}{\partial t} + \mathcal{L}V(x(t)) \right) dt + \sum_{k=1}^{m} \left(\sum_{i=1}^{n} \frac{\partial V(x(t))}{\partial x_i} \sigma_{ik} \right) dw_k(t).$$

Let $\tau \equiv \inf\{t \geq 0 : x(t) \in \partial \mathcal{X}\}$ and define $\tau \wedge t = \min\{\tau, t\}$. Integrating $dV(x(t))$ over the time interval $[0, \tau \wedge t]$ and taking the expectation yields

$$\mathbb{E}\{V(x(\tau \wedge t))\} = V(x(0)) + \mathbb{E}\left\{ \int_0^{\tau \wedge t} \left(\frac{\partial V(x(s))}{\partial s} + \mathcal{L}V(x(s)) \right) ds \right\}.$$

Taking the limit of the above equation as $t \to \infty$ and using the last condition in the proposition's statement, one finds

$$\mathbb{E}[V(x(\tau))] \leq V(x_0) - \mathbb{E}\left[\int_0^{\tau} ds \right] = V(x_0) - \mathbb{E}[\tau].$$

τ is the first time the state trajectory hits the boundary set $\partial \mathcal{X}$ and so the above equation implies that the MFPT satisfies

$$\mathbb{E}\{\tau\} \leq V(x_0) - \mathbb{E}[V(x(\tau))].$$

Boundary points of $\mathcal{X}$ are limit points of $\mathcal{X}$ and since $V(x(t)) \geq 0$ on $\mathcal{X}$, this means $V(x(t)) = 0$ on $\partial \mathcal{X}$. One may therefore conclude that $\mathbb{E}\{V(x(\tau))\} = 0$ which implies $\mathbb{E}\{\tau\} \leq V(x_0)$. By the third condition in the proposition, we know that $V(x_0) \leq \theta$ on $\mathcal{X}_0$ which implies $\mathbb{E}\{\tau\} \leq \theta$. $\qquad \square$

Remark 5.3.2. From the proof of this proposition, we see that $\mathbb{E}\{V(x(t))|V(x_0)\} \leq V(x_0)$ for $0 \leq t \leq \tau$. Since $\mathcal{X}$ is a bounded set, this implies $\mathbb{E}\{V(x(t))\} < \infty$ which along with the requirement that $V(x(t)) \geq 0$ for all x implies the stochastic process generated by $V(x(t))$ is a supermartingale.

Proposition 5.3.3 below is a simple extension of Proposition 5.3.1 and characterizes an upper bound for the MFPT of diffusion processes. The proof of Proposition 5.2 is similar with that of Proposition 5.3.1 except that we need to use the corresponding generator $\mathcal{L}_{DP}V(x(t))$ of diffusion processes defined in (5.9).

Proposition 5.3.3. *Consider a diffusion process $dx(t) = f(x(t))dt + \sigma(x(t))dw(t)$ defined on a bounded open subset $\mathcal{X} \subset \mathbb{R}^n$ with smooth boundary $\partial\mathcal{X}$. Assume the initial condition satisfies $x(0) = x_0 \in \mathcal{X}_0 \subset \mathcal{X}$. If there exists a function $V(x(t)) \in C^2(\mathbb{R}^n)$ and a constant $\theta > 0$ such that*

$$
\begin{aligned}
V(x(t)) &\geq 0, && \forall x \in \mathcal{X}, \\
V(x(t)) &\leq 0, && \forall x \in \partial\mathcal{X}, \\
V(x(t)) &\leq \theta, && \forall x \in \mathcal{X}_0, \\
\frac{\partial V(x(t))}{\partial t} + \mathcal{L}_{DP}V(x(t)) &\leq -1, && \forall x \in \mathcal{X},
\end{aligned}
$$

where $\mathcal{L}_{DP}V(x(t))$ is given in (5.9), then $\mathbb{E}\{\tau\} \leq \theta$ with $\tau = \inf_t\{t \geq 0 : x(t) \in \partial\mathcal{X}\}$.

Section 5.4 presents SOS relaxation methods for searching function $V(x(t))$ in Propositions 5.3.1 and 5.3.3.

5.3.2   Upper Bound for Safety Probability

We now present a characterization of the safety probability for JDP in (5.8). In particular, Proposition 5.3.4 below characterizes a function $V(x(t))$ that bounds such probability. The proof of this proposition is based on the proof in [90, Theorem 1] (except that we use the JDP's generator in (5.9)). The proof uses Dynkin's formula in (5.12) and the supermartingale inequality in (5.13). Using the polynomial expression of the JDP's generator in (5.9), a two-stage SOS optimization method to compute the probability bound characterized in Proposition 5.3.4 is presented in Section 5.4.

Proposition 5.3.4. *Consider the set* $\Omega_{V,\gamma} = \{x \in \mathcal{X} \mid V(x) < \gamma\}$ *defined by a constant* $\gamma > 0$ *and a nonnegative function* $V(x(t)) \in C^2(\mathbb{R}^n)$. *Let* $\{x(t)\}$ *with* $x_0 \in \Omega_{V,\gamma}$ *be a right continuous JDP in (5.8) defined on* $\Omega_{V,\gamma}$ *until at least some time* $\tau > \tau_\gamma \doteq \inf\{t \in \mathbb{R}_+ \mid x(t) \notin \Omega_{V,\gamma}\}$. *Let* $\mathcal{L}V(x(t))$ *be the JDP's generator in (5.9) and let* $V(x(t))$ *be in the domain of* $\mathcal{L}V(x(t))$. *For a constant* $\alpha > 0$ *and finite interval* $t \in [0, T]$, *assume the condition below holds (with probability 1) in* $\Omega_{V,\gamma}$.

$$\mathcal{L}V(x(t)) \leq -\alpha V(x(t)) + \beta(t), \tag{5.14}$$

where $\beta(t)$ *is a continuous, strictly positive function on* $[0, T]$ *with a maximum* $\epsilon = \max_{0 \leq t \leq T} \beta(t)$. *Let* $\mathcal{B}(t) = \int_0^t \beta(s)ds$ *for* $t \in [0, T]$ *and define* $\theta = \alpha/\epsilon$. *Then for any* $x_0 \in \Omega_{V,\gamma}$

$$\mathbb{P}\left\{ \sup_{0 \leq t \leq T} V(x(t)) \geq \gamma \mid x_0 \right\} \leq \frac{V(x_0) + \frac{1}{\theta}(e^{\theta \mathcal{B}(T)} - 1)}{\gamma e^{\theta \mathcal{B}(t)} + \frac{1}{\theta}(e^{\theta \mathcal{B}(T)} - e^{\theta \mathcal{B}(t)})}. \tag{5.15}$$

Proof. In the interval $t \in [0, T]$, define

$$W(x(t)) = e^{\frac{\alpha t}{\beta(t)}} V(x(t)) + \frac{\beta(t)}{\alpha} \left(e^{\frac{\alpha \mathcal{B}(T)}{\beta(t)}} - e^{\frac{\alpha \mathcal{B}(t)}{\beta(t)}} \right), \tag{5.16}$$

Let $\lambda > 0$ be such that if $V(x(t)) > \gamma$, then $W(x(t)) \geq \lambda$. Consider the set

$$\Omega_{W,\lambda} \triangleq \{x \in \mathbb{R}^n, t \in \mathbb{R}_+ : W(x(t)) < \lambda, \ t < T\}. \tag{5.17}$$

Let $\tau \doteq \inf_t\{t \in \mathbb{R}_+ \mid x(t) \notin \Omega_{W,\lambda}\}$ and consider the process $\{x(s)\} = \{x(t \wedge \tau)\}$ that takes values on $\Omega_{W,\lambda}$, with $t \wedge \tau = \min(t, \tau)$. Let $\tilde{\mathcal{L}}$ be the infinitesimal generator of $\{x(s)\}$ acting on the function $W(x(s))$ defined on $\Omega_{W,\lambda}$. From the definition of $W(x(s))$ in (5.16), we have

$$\tilde{\mathcal{L}}W(x(s)) = \theta \beta(s) e^{\theta \mathcal{B}(s)} V(x(s)) + e^{\theta \mathcal{B}(s)} \mathcal{L}V(x(s)) - \beta(s) e^{\theta \mathcal{B}(s)},$$

$$= e^{\theta \mathcal{B}(s)} \left(\theta \beta(s) V(x(s)) + \mathcal{L}V(x(s)) - \beta(s) \right).$$

109

Since $\theta\beta(s) = \alpha\beta(s)/\epsilon \leq \alpha$ and using $\mathcal{L}V(x(t))$ in (5.14), then

$$\tilde{\mathcal{L}}W(x(s)) \leq e^{\theta\mathcal{B}(s)}\left(\alpha V(x(s)) - \alpha V(x(s)) + \beta(s) - \beta(s)\right) \leq 0.$$

Applying Dynkin's formula to function $W(x(t))$ gives

$$\mathbb{E}\{W(x(t))\} = \mathbb{E}\{W(x(0))\} + \mathbb{E}\left\{\int_0^s \tilde{\mathcal{L}}W(x(t))\,dt\right\} \leq W(x_0),$$

which implies $W(x(t))$ is a supermartingale. Thus, for $\lambda \geq 0$

$$\mathbb{P}\left\{\sup_{0\leq t\leq T} W(x(t)) \geq \lambda \,\Big|\, x(0) = x_0\right\} \leq W(x_0)/\lambda$$
$$\leq V(x_0)/\lambda + (e^{\theta\mathcal{B}(T)} - 1)/\lambda\theta. \tag{5.18}$$

Now since $W(x(t)) \geq \lambda$ implies $V(x(t)) \geq \gamma$, then

$$W(x(t)) = e^{\theta\mathcal{B}(t)}V(x(t)) + (e^{\theta\mathcal{B}(T)} - e^{\theta\mathcal{B}(t)})/\theta \geq \lambda$$

can be rearranged to obtain

$$V(x(t)) \geq e^{-\theta\mathcal{B}(t)}\left(\lambda - (e^{\theta\mathcal{B}(T)} - e^{\theta\mathcal{B}(t)})/\theta\right).$$

Thus, the condition that $V(x(t)) \geq \gamma$ implies

$$\gamma = e^{-\theta\mathcal{B}(t)}\left(\lambda - (e^{\theta\mathcal{B}(T)} - e^{\theta\mathcal{B}(t)})/\theta\right).$$

Solving the above equation for λ and then substituting it back into (5.18) gives the probability bound in equation (5.15). $\qquad\square$

Remark 5.3.5. Note that the bound in (5.15) corresponds to the probability that the process' sample paths fail to stay inside the set $\Omega_{V,\gamma}$. One may notice that the

condition in (5.14) is less restrictive than the requirement $\mathcal{L}V(x) \leq 0$ used in [122]. This suggests that a better estimate of (5.15) can be achieved by choosing the function $\beta(t)$ in (5.14) to be a strictly decreasing function with small maxima (cf. Section 5.4 for one choice of $\beta(t)$).

5.4 SOS Optimization

As discussed in Section 5.1, provided that the drift, diffusion and the jump terms in JDP (5.8) are polynomial functions and the sets $\mathcal{X}, \partial\mathcal{X}, \mathcal{X}_0, \mathcal{X}_s$ in equations (5.2)-(5.3) are semialgebraic, then the search for a barrier certificate, $V(x(t))$, can be formulated as an SOS optimization problem. In this SOS optimization, $V(x(t))$ is a polynomial function whose coefficients are the decision variables that will be determined during the optimization task. Thus, our goal is to formulate polynomial representations for the conditions (given in the previous subchapter) that guarantee the process $\{V(x(t))\}$ is a supermartingale. One issue in formulating such a representation comes from the integral term in the JDP's generator in (5.11). Proposition 5.4.1 below shows how to address this issue.

Proposition 5.4.1. *Let $y \in \mathbb{R}^n$ be an n-dimensional independent random variable with distribution $F(y)$. Let $V(x) = \sum_{|\alpha| \leq p} c_\alpha x^{[\alpha]}$ be a multi-index representation of polynomial function $V(x)$. Then*

$$\int \left(V(x+y) - V(x) \right) dF(y) = \sum_{1 \leq |\beta| \leq p} \frac{1}{\beta!} \partial^{[\beta]} \left[V(x) \right] \mathbb{M}^{|\beta|}, \qquad (5.19)$$

and the generator in (5.11) can be rewritten as

$$\mathcal{L}V(x(t)) = \frac{\partial V(x(t))}{\partial x} \left(f(x(t)) - \delta J(t) \right) + \frac{1}{2} \mathrm{Tr} \left(\sigma^T(x(t)) \frac{\partial^2 V(x(t))}{\partial x^2} \sigma(x(t)) \right)$$
$$+ \sum_{1 \leq |\beta| \leq p} \frac{1}{\beta!} \partial^{[\beta]} \left[V(x) \right] \mathbb{M}^{|\beta|}. \quad (5.20)$$

111

Proof. We only need to show that equation (5.19) holds since its substitution into the integral term in equation (5.11) gives the generator in equation (5.20). Let us write

$$V(x+y) = \sum_{|\alpha| \leq p} c_\alpha (x+y)^{[\alpha]} = \sum_{|\alpha| \leq p} c_\alpha \sum_{0 \leq |\beta|, \beta \leq \alpha} \binom{\alpha}{\beta} x^{[\alpha-\beta]} y^{[\beta]},$$

$$= \sum_{|\alpha| \leq p} c_\alpha \left[x^{[\alpha]} + \sum_{1 \leq |\beta|, \beta \leq \alpha} \binom{\alpha}{\beta} x^{[\alpha-\beta]} y^{[\beta]} \right].$$

For notational convenience, let us denote the difference $V(x+y) - V(x)$ as $\Delta V(x,y)$. Using the above sum, one can write this difference as

$$\Delta V(x,y) = \sum_{|\alpha| \leq p} c_\alpha \sum_{1 \leq |\beta|, \beta \leq \alpha} \binom{\alpha}{\beta} x^{[\alpha-\beta]} y^{[\beta]},$$

and since

$$\partial^{[\beta]} \left[x^{[\alpha]} \right] = \begin{cases} \dfrac{\alpha!}{(\alpha-\beta)!} x^{[\alpha-\beta]} & \text{if } \beta \leq \alpha, \\[2mm] 0 & \text{otherwise,} \end{cases}$$

the expression for $\Delta V(x,y)$ defined above can then be rewritten as

$$\Delta V(x,y) = \sum_{|\alpha| \leq p} c_\alpha \sum_{1 \leq |\beta|, \beta \leq \alpha} \frac{1}{\beta!} \partial^{[\beta]} \left[x^{[\alpha]} \right] y^{[\beta]}.$$

Now expand out the first summation to obtain

$$\Delta V(x,y) = \sum_{|\alpha|=1} c_\alpha \sum_{|\beta|=1} \frac{1}{\beta!} \partial^{[\beta]} \left[x^{[\alpha]} \right] y^{[\beta]} + \sum_{|\alpha|=2} c_\alpha \sum_{1 \leq |\beta| \leq 2} \frac{1}{\beta!} \partial^{[\beta]} \left[x^{[\alpha]} \right] y^{[\beta]} + \cdots$$

$$+ \sum_{|\alpha|=p} c_\alpha \sum_{1 \leq |\beta| \leq p} \frac{1}{\beta!} \partial^{[\beta]} \left[x^{[\alpha]} \right] y^{[\beta]}.$$

The order of the summations can now be interchanged since α and β are no longer

112

directly coupled to yield

$$\Delta V(x,y) = \sum_{|\beta|=1} \frac{1}{\beta!} \left[\sum_{|\alpha|=1} c_\alpha \partial^{[\beta]} \left[x^{[\alpha]} \right] \right] y^{[\beta]} + \sum_{1 \le |\beta| \le 2} \frac{1}{\beta!} \left[\sum_{|\alpha|=2} c_\alpha \partial^{[\beta]} \left[x^{[\alpha]} \right] \right] y^{[\beta]} + \cdots$$
$$+ \sum_{1 \le |\beta| \le p} \frac{1}{\beta!} \left[\sum_{|\alpha|=p} c_\alpha \partial^{[\beta]} \left[x^{[\alpha]} \right] \right] y^{[\beta]}.$$

Reordering the terms in the first summation yields,

$$\Delta V(x,y) = \sum_{|\beta|=1} \frac{1}{\beta!} \left[\sum_{1 \le |\alpha| \le p} c_\alpha \partial^{[\beta]} \left[x^{[\alpha]} \right] \right] y^{[\beta]} + \sum_{|\beta|=2} \frac{1}{\beta!} \left[\sum_{2 \le |\alpha| \le p} c_\alpha \partial^{[\beta]} \left[x^{[\alpha]} \right] \right] y^{[\beta]} + \cdots$$
$$+ \sum_{|\beta|=p} \frac{1}{\beta!} \left[\sum_{|\alpha|=p} c_\alpha \partial^{[\beta]} \left[x^{[\alpha]} \right] \right] y^{[\beta]}.$$

Because $\partial^{[\beta]} \left[x^{[\alpha]} \right] = 0$ when $\alpha \le \beta$, the summation limits of the inner sums can be extended from 1 to p, thereby yielding

$$\Delta V(x,y) = \sum_{|\beta|=1} \frac{1}{\beta!} \left[\sum_{1 \le |\alpha| \le p} c_\alpha \partial^{[\beta]} \left[x^{[\alpha]} \right] \right] y^{[\beta]} + \sum_{|\beta|=2} \frac{1}{\beta!} \left[\sum_{1 \le |\alpha| \le p} c_\alpha \partial^{[\beta]} \left[x^{[\alpha]} \right] \right] y^{[\beta]} + \cdots$$
$$+ \sum_{|\beta|=p} \frac{1}{\beta!} \left[\sum_{1 \le |\alpha| \le p} c_\alpha \partial^{[\beta]} \left[x^{[\alpha]} \right] \right] y^{[\beta]}. \quad (5.21)$$

Now note that

$$\partial^{[\beta]} V(x) = \partial^{[\beta]} \left[\sum_{|\alpha| \le p} c_\alpha x^{[\alpha]} \right] = \sum_{1 \le |\alpha| \le p} c_\alpha \partial^{[\beta]} \left[x^{[\alpha]} \right],$$

which is simply the inner sum in (5.21) and so the difference becomes

$$\Delta V(x,y) = \sum_{1 \le |\beta| \le p} \frac{1}{\beta!} \partial^{[\beta]} \left[V(x) \right] y^{[\beta]}.$$

Integrating both sides with respect to $F(y)$, and since each component of y is inde-

pendent from each other, gives

$$\int \Delta V(x,y)dF(y) = \sum_{1\leq|\beta|\leq p} \frac{1}{\beta!}\partial^{[\beta]}\left[V(x)\right]\int y^{|\beta|}dF(y), = \sum_{1\leq|\beta|\leq p} \frac{1}{\beta!}\partial^{[\beta]}\left[V(x)\right]\mathbb{M}^{|\beta|},$$

where we have noticed that the integral $\int y^{|\beta|}dF(y) = \mathbb{M}^{|\beta|}$ is the $|\beta|$-th moment of y. Finally, the substitution of the above expression to the integral term in (5.11) gives the JDP generator defined in (5.20). $\qquad\square$

Using the polynomial representation of the JDP's generator in (5.20), we now formulate the SOS optimization problems for computing upper bounds of the quantities stated in Propositions 5.3.1 - 5.3.4.

5.4.1 Computation of MFPT Upper Bound

Proposition 5.4.2 below formulates the SOS optimization problem to compute an upper bound for the MFPT in Proposition 5.3.1. In particular, this proposition formulates the search of a barrier certificate $V(x(t))$ in Proposition 5.3.1 from which an upper bound of the MFPT, θ can be deduced.

Proposition 5.4.2. *Consider the JDP in (5.8) with initial condition $x_0 \in \mathcal{X}_0$. Let the sets $\mathcal{X}$, $\mathcal{X}_0$, $\partial\mathcal{X}$ be described by $\mathcal{X} = \{x \in \mathbb{R}^n : g_{\mathcal{X}}(x) \geq 0\}, \mathcal{X}_0 = \{x \in \mathbb{R}^n : g_{\mathcal{X}_0}(x) \geq 0\}, \partial\mathcal{X} = \{x \in \mathbb{R}^n : g_{\partial\mathcal{X}}(x) = 0\}$, respectively, where the g's are polynomial functions. Consider the polynomial parameterization $\mathcal{V} \in C^2(\mathbb{R}^n)$ of $V(x(t))$, and define $\tau \equiv \inf\{t \geq 0 : x(t) \in \partial\mathcal{X}\}$. If there exists a function $V(x(t)) \in \mathcal{V}$, constants $\theta, \epsilon > 0$, and SOS polynomials $\sigma_{\mathcal{X}}(x), \sigma_{\mathcal{X}_0}(x), \sigma_{\partial\mathcal{X}}(x)$ such that the SOS optimization*

$$\min \quad \theta$$

$$
\begin{aligned}
\text{such that} \quad & V(x(t)) - \sigma_{\mathcal{X}}(x)g_{\mathcal{X}}(x) - \epsilon && \text{is SOS,} \\
& -V(x(t)) - \sigma_{\partial\mathcal{X}}(x)g_{\partial\mathcal{X}}(x) && \text{is SOS,} \qquad (5.22) \\
& -V(x(t)) + \gamma - \sigma_{\mathcal{X}_0}(x)g_{\mathcal{X}_0}(x) && \text{is SOS,} \\
& -\frac{\partial V(x(t))}{\partial t} - \mathcal{L}V(x(t)) - \sigma_{\mathcal{X}}(x)g_{\mathcal{X}}(x) - 1 && \text{is SOS,}
\end{aligned}
$$

has a feasible solution, then $\mathbb{E}\{\tau\} \leq \theta$.

Proof. The conditions for $V(x(t))$ in the above SOS program are the SOS relaxations of inequality constraints in Proposition 5.3.1. In particular, such a relaxation is constructed using SOS polynomial multipliers $\sigma_{(.)}(x)$. Thus for a choice of SOS polynomial parameterization, any feasible θ obtained from the minimization in Proposition 5.4.2 will serve as the tightest bound for the MFPT of the JDP in (5.8). $\qquad \square$

5.4.2 Computation of Safety Probability's Upper Bound

This section presents a two-stage SOS optimization method to compute the probability bound in Proposition 5.3.4. The first stage on this method computes a Lyapunov function $V(x)$ for the drift term of the JDP in (5.8) that maximizes a subset $\Omega_{V,\gamma}$ of the system's region of attraction (ROA). The second stage on the method then uses the obtained Lyapunov function $V(x)$ to estimate the values of constant α and function $\beta(t)$ in (5.14). Each of these stages is discussed below and illustrated in Algorithm 3.

5.4.2.1 Computation of $V(x)$ and $\Omega_{V,\gamma}$

Let $\dot{x} = f(x)$, $x(0) = x_0$ with $x \in \mathbb{R}^n$, $f(x) \in \mathbb{R}[x]$ be a polynomial system and assume $x = 0$ is a locally asymptotically stable equilibrium. The ROA of this system is defined as

$$\mathcal{R} := \{x_0 \mid \text{if } x(0) = x_0 \text{ then } \lim_{t \to \infty} x(t) = 0\}.$$

An exact computation of $\mathcal{R}$ is generally hard and one often limits the search to estimate a largest subset $\Omega_{V,\gamma} \subset \mathcal{R}$ [84, 150]. The method described in this section considers the set $\Omega_{V,\gamma} := \{x \mid V(x) \leq \gamma\}$ to be an ellipsoid defined by a constant $\gamma > 0$ and a function $V(x) = x^T M x$ with $M \in \mathcal{S}_+^n$. Thus the objective is to compute $\underline{\gamma} > 0$ in the following optimization problem.

$$\underline{\gamma} := \max \gamma, \quad \text{s.t.} \ \Omega_{V,\gamma} \subset \mathcal{R}. \tag{5.23}$$

We solve this problem by computing $\underline{\gamma}$ which guarantees that $\Omega_{V,\gamma}$ is a maximal inner approximation $\Omega_{V,\gamma} \subset \Omega_\omega$ of the bounded level set Ω_ω of a Lyapunov function $\mathcal{V}(x)$ defined as

$$\Omega_\omega = \{x \mid \mathcal{V}(x) \leq \omega\} \subset \{x \mid \nabla \mathcal{V}(x) f(x) < 0\} \cup \{0\},$$

for a constant $\omega > 0$. Note that any feasible $\underline{\gamma}$ will guarantees that $\Omega_{V,\gamma} \subset \Omega_\omega \subset \mathcal{R}$ and the corresponding $V(x)$ is also a Lyapunov function. The optimization problem in (5.23) can then be formulated as two optimization problems of the form

$$\omega^* := \max \omega, \quad \text{s.t.} \ \Omega_\omega \subset \mathcal{R},$$
$$\underline{\gamma} := \max \gamma, \quad \text{s.t.} \ \Omega_{V,\gamma} \subset \Omega_\omega,$$

which can be reformulated as the following two SOS optimization problems

$$\omega^* := \max \omega, \quad \text{s.t.} \ -(\omega - \mathcal{V})s_1 - (\nabla \mathcal{V} f(x) + \sigma(x)) \in \Sigma(x),$$
$$\mathcal{V}(0) = 0, \ \mathcal{V}(x) \in \Sigma(x).$$
$$\underline{\gamma} := \max \gamma, \quad \text{s.t.} \ -(\gamma - V)s_2 - (\omega^* - \mathcal{V}) \in \Sigma(x),$$
$$-(\omega^* - V)s_1 - (\nabla V f(x) + \sigma(x)) \in \Sigma(x),$$
$$V(0) = 0, \ V(x) \in \Sigma(x),$$

where $s_1, s_2 \in \Sigma(x)$ and $\sigma(x) = \varepsilon\|x\|$ for a small constant $\varepsilon > 0$. The constraints on ω^* are the SOS relaxations for the Lyapunov function property of $\mathcal{V}(x)$ whereas the constraints on $\underline{\gamma}$ are the SOS relaxations for the Lyapunov function property of $V(x)$ and the set containment condition $\Omega_{V,\gamma} \subset \Omega_\omega$ (cf. [151, Lemma 1]). As shown in Stage 1 of Algorithm 3 (cf. [134, 21]), these optimizations can be solved iteratively by bisections on both ω and γ using $\mathcal{V}(x) = \mathcal{V}_0(x)$ for initialization in which $\mathcal{V}_0(x)$ is a Lyapunov function of the linearized system. In particular, $\mathcal{V}_0(x) = x^T P x$ for a matrix $P \in \mathcal{S}_+^n$ that satisfies the Lyapunov equation $A^T P + PA = -Q$ with $A := \frac{\partial f(x)}{\partial x}|_{x=0}$ and $Q \in \mathcal{S}_+^n$. Note that the maximization of ω^* is bilinear in the pairs $(\mathcal{V}, s_1)$ and (ω, s_1) while the maximization of $\underline{\gamma}$ is bilinear in the pairs $(\gamma, s_2), (V, s_1)$ and (V, s_2). Algorithm 3 shows that these bilinearities are avoided by holding one decision variable fixed while searching for the other one [21]. The largest $\Omega_{V,\gamma}$ is then searched for by iterating Stage 1 until $\underline{\gamma}$ is no longer increasing.

5.4.2.2   Computation of α and $\beta(t)$

Based on the values of $V(x)$ and $\Omega_{V,\gamma}$ obtained in Stage 1, then Stage 2 searches for a constant $\alpha > 0$ and a function $\beta(t)$ which satisfy (5.14). From the discussion in Remark 5.3.5, we choose $\beta(t)$ to be a rational function of time of the form $\beta(t) = t^{q-1}/(c + t^{2q})$ with a maxima $\epsilon > 0$ where $c > 0$ is a constant and $q \in \mathbb{Z}_+$. Since ϵ is required to be small (cf. Remark 5.3.5), the condition in (5.14) suggests that one needs to find a maximum α which satisfies equation (5.14) for a given $V(x)$ and a prespecified ϵ. If such an α is feasible then the corresponding $\beta(t)$ can be determined by choosing a maximum c such that $\max \beta(t) \leq \epsilon$. The following two-step SOS optimization algorithm summarizes the computation of constant α and function $\beta(t)$ (cf. Stage 2 in Algorithm 3).

1. Fix $\epsilon > 0$ and solve for α in the following optimization.

$$\alpha^* := \max \alpha, \text{ s.t. } \epsilon - \mathcal{L}V - \alpha V \in \Sigma(x).$$

Algorithm 3 Two-stage SOS optimization.

Stage 1 - Computation of $V(x)$ and $\Omega_{V,\gamma}$.

Input: $f(x), \mathcal{V}_0(x)$ and $V(x)$

1: Fix $\mathcal{V}$, bisect ω, solve the optimization below for $s_1 \in \Sigma(x)$.

$\omega^* := \max \omega$, s.t. $-(\omega - \mathcal{V})s_1 - (\nabla\mathcal{V}f(x) + \sigma(x)) \in \Sigma(x)$

2: Fix $\mathcal{V}, \omega^*$, bisect γ, solve the optimization below for $s_1 \in \Sigma(x)$.

$\underline{\gamma} := \max \gamma$, s.t. $-(\gamma - V)s_2 - (\omega^* - \mathcal{V}) \in \Sigma(x)$

3: Fix $s_1, s_2, \underline{\gamma}, \omega^*$ and find $V(x)$ such that

$V(0) = 0, \quad V(x) \in \Sigma(x),$

$-(\omega^* - V)s_2 - (\nabla\mathcal{V}f(x) + \sigma(x)) \in \Sigma(x),$

$-(\underline{\gamma} - V)s_1 + (\omega^* - \mathcal{V}) \in \Sigma(x).$

4: Set $\mathcal{V} = V$ and repeat from line 1 until $\underline{\gamma}$ stop increasing.

Stage 2 - Computation of α and c.

Input: $V(x)$, $\epsilon_0 > \underline{\epsilon}$

5: **procedure** $[\alpha, c] = $ BOUND $(V, \Omega_{V,\gamma}, \epsilon_0)$

6:      set $\underline{\epsilon} > 0, \epsilon \leftarrow \epsilon_0$

7:      **while** $\epsilon \geq \underline{\epsilon}$ **do**

8:          max α, s.t. $[-\mathcal{L}V - \alpha V + \epsilon] \in \Sigma(x)$

9:          **if** α exists **then**

10:             max c, s.t. $[t^{q-1} - (c + t^{2q})(\mathcal{L}V + \alpha V)] \in \Sigma(x, t)$

11:          **end if**

12:          $\epsilon \leftarrow \epsilon/2$ ▷ Bisection on ϵ

13:      **end while**

14:      **return** α^* **and** c^* ▷ The optimal α and c

15: **end procedure**

---

2. Fix α and solve for c in the following optimization.

$$c^* := \max c, \text{ s.t. } t^{q-1} - (c + t^{2q})(\mathcal{L}V + \alpha V) \in \Sigma(x, t),$$

where $\Sigma(x, t)$ denotes the set of SOS polynomials in variables x and t. Note that the constraint in the second step is an SOS representation of the condition in (5.14). As illustrated in Stage 2 of Algorithm 3, this two-step optimization is solved iteratively through bisection on ϵ. The specified small constant $\underline{\epsilon} > 0$ in Algorithm 3 can be used as a stopping criteria in searching for a feasible α. In particular, Remark 5.3.5 suggests that a small value of $\underline{\epsilon}$ will gives a better estimate for the bound in (5.15).

Remark 5.4.3. The feasibility of α and c in Stage 2 depends on $V(x)$ obtained in Stage 1. If a solution to α is not feasible for a fixed order $V(x)$, one may try repeat the search from Stage 1 using higher degree $V(x)$ but at the expense of higher computational effort. This computational complexity is still an active research problem in SOS optimization method [117, 151]. Note also that the obtained ROA estimate can be improved by using simulation data [151] or using the level-set method [155].

5.5 Examples

5.5.1 MFPT Approximation

This section illustrates the use of MFPT analysis discussed in the previous sections in ecosystems management. In particular, we consider the problem of choosing a harvesting strategy to manage the bass-crayfish population in freshwater lakes. Bass-crayfish interaction is an intraguild predation system in which both species compete for the same resource while also predating on one another. The model presented in this section has two equilibria; one in which the bass dominate the ecosystem and the other in which the crayfish dominate the ecosystem. An outbreak of crayfish is undesirable as it can suppress the bass population. If such an outbreak occurs, management strategies are needed to shift the crayfish-dominated equilibrium point to the bass-dominated equilibrium point. One method to achieve this management objective is to permit the harvesting of crayfish by anglers. In general, this harvesting

process can be modeled as a jump process in which the size and the intensity of harvesting events are variables that the ecosystem manager needs to set.

This example was drawn from a paper [38] that used MFPT as a basis for management decisions. The underlying nondimensionalized model is given by the following state equations,

$$
\begin{aligned}
\dot{x}_1(t) &= x_1(k_1 - k_{11}x_1 - k_{12}x_2) - \frac{k_{12}^* x_2 x_1^2}{K_1^2 + x_1^2} - \sum_{i=1}^{N_t} y_i \delta(t - \tau_i), \\
\dot{x}_2(t) &= r_{21}x_2(k_2 - k_{22}x_2 - k_{21}x_1) + \eta \frac{k_{12}^* x_2 x_1^2}{K_1^2 + x_1^2},
\end{aligned}
\tag{5.24}
$$

where the biomass of the crayfish and bass are denoted as x_1 and x_2, respectively, k_i and k_{ii} are the intrinsic growth rate and the strength of density dependence of the ith species, respectively, k_{ij} is the competition rate on resource between species i and j, k_{ij}^* is the attack rates of species j on i, K_i is the carrying capacity, η is the conversion efficiency, and r_{ij} is the ratio of growth rate between species i and j. The parameter values are $k_i = k_{ii} = 1, k_{12} = 0.7, k_{21} = 0.9, k_{12}^* = 0.075, r_{21} = 1.5, \eta = 0.01, K_1 = 0.1$. The last term in the first equation of (5.24) models crayfish harvesting as a compound Poisson process in which the harvest size $\{y_i\}_{i=1}^{N_t}$ and the harvest times, $\{\tau_i\}_{i=1}^{N_t}$ are i.i.d with exponential distribution of intensity μ and λ, respectively, and N_t is the number of harvest events in the interval $[0, t]$.

Figure 5.3 plots the isoclines for equation (5.24), identifies the two stable equilibria and their regions of attraction (ROA), and marks the separatrix between the two ROAs. Assuming that the system's current state lies in the ROA dominated by the crayfish, we are interested in computing the MFPT required by the process' sample paths to finally reach the ROA dominated by the the bass population.

To compute an upper bound for the MFPT using the SOS program in Proposition
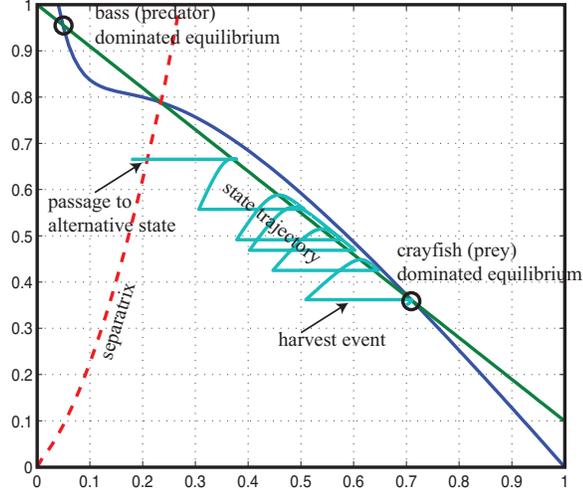
Figure 5.3. ROA in bass-crayfish interaction model (5.24) [37].

5.4.2, we define the following sets.

$$\mathcal{X} = \left\{ x \in \mathbb{R}_+^2, \, t \in \mathbb{R}_+ \, \middle| \, x_1(1 - x_1) \geq 0, \, x_2(1 - x_2) \geq 0, t(T - t) > 0 \right\},$$

$$\mathcal{X}_0 = \left\{ x \in \mathbb{R}_+^2 \, \middle| \, (x_1 - 0.72)^2 + (x_2 - 0.36)^2 \leq 10^{-4} \right\},$$

$$\partial \mathcal{X} = \{ x \in \mathbb{R}_+^2 \, \middle| \, 0.27x_1 \geq x_1^2, \, x_2 \geq x_2^2, x_2 - 0.14x_1^3 - 9.5x_1^2 - 1.1x_1 + 3.10^{-4} = 0 \}.$$

Region $\mathcal{X}$ characterizes a unit square in $\mathbb{R}_+^2$ over the time interval $[0, T]$. The initial region $\mathcal{X}_0$ is a disk of radius 0.01, centered at the crayfish-dominated equilibrium. The boundary region $\partial \mathcal{X}$ is the separatrix shown in Figure 5.3.

Figure 5.4a shows the MFPT approximation (circle) for $\mu = 0.1$ and $\lambda \in [0, 6]$ obtained using SOSTOOLS. This plot also shows the MFPT obtained using a Monte Carlo (MC) simulation with a 95% confidence interval and estimates (plus) obtained in [37]. The estimates in [37] were based on a linearization and were only valid for small mean harvest sizes (μ). As a result, the estimates from [37] under approximate the actual MFPT seen in MC simulations, whereas our results provide reasonable upper bounds on the MFPT.

(a) MFPT comparison for various λ. (b) MFPT as a function of μ and λ..
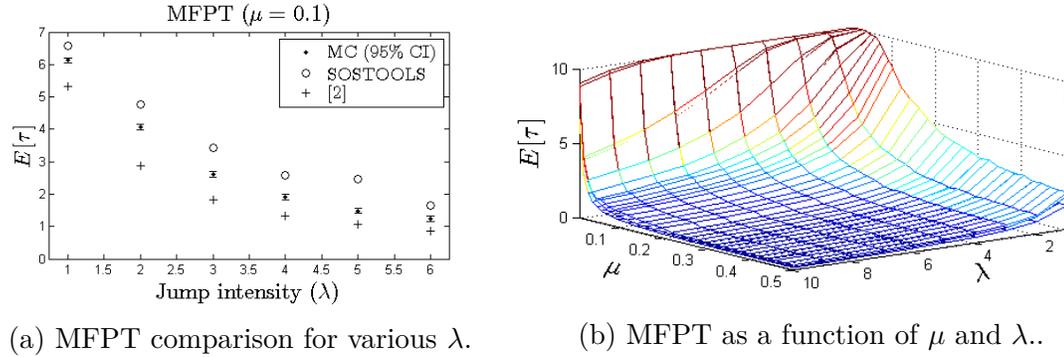
Figure 5.4. Result of MFPT approximation.

Figure 5.4b shows a more complete surface plot of our MFPT approximations for a range of μ and λ. Clearly, one can maximize the likelihood of a regime shift by simply increasing the intensity of harvesting. In general, one would want to limit such harvesting intensity since large harvesting intensity may drive the crayfish population to extinction. While crayfish may be considered to be a "nuisance", the extinction of a species in the eco-system reduces overall bio-diversity and often makes such systems more prone to collapse from extreme events [73]. Thus for management purposes, a reasonable choice on harvesting strategy involves limiting the harvesting rates λ and μ to minimize the likelihood of crayfish extinction while still achieving a regime shift over a specified time interval.

5.5.2 Safety Probability

This example illustrates the use of stochastic safety analysis in ecosystems management. In particular, we are interested in bounding the probability that a coexisting bass-crayfish population fails to maintain a desired bass-dominated state in the presence of harvesting activities.

The density of the bass population in many freshwater lakes has been observed to decline significantly due to sport harvesting activities. In many cases, overfishing

of bass may shift the lake from a bass-dominated state to an undesired crayfish-dominated one whereupon the bass population is further suppressed due to their loss in competition for food and shelter. In order to evaluate the sustainability of the bass population under such harvesting pressure, we need methods to evaluate the likelihood that a coexisting bass-crayfish population fails to stay within the desired bass-dominated region. One such method can be developed by modeling the harvested bass-crayfish interaction as a jump diffusion process in which the jump process part of the model is used to describe the harvesting activities of certain intensity and size parameters. For a given set of harvesting parameters, the objective is then to bound the probability that the coexisting bass-crayfish population fails to stay within the bass-dominated region.

A normalized model of crayfish (x_1) and bass (x_2) interaction under harvesting events is given by [37]

$$
\begin{aligned}
dx_1(t) &= \left(x_1(1 - x_1^2 - 0.65x_2) - \frac{0.65x_2x_1^3}{0.01 + x_1^4} \right) dt + \sigma x_1 dw_1(t), \\
dx_2(t) &= \left(1.5x_2(1 - x_2^2 - x_1) + \frac{0.01x_2x_1^3}{0.01 + x_1^4} \right) dt + \sigma x_2 dw_2(t) - x_2 dJ(t),
\end{aligned}
\tag{5.25}
$$

where $(x_1, x_2) = x \in \mathcal{X}$ and $\mathcal{X}$ is a unit square. The Wiener processes $\{w_i(t)\}$, $i = 1, 2$ of intensities $\sigma = 0.05$ describe small fluctuations in each population due to variations in growth rate or other environmental factors. The jump process $\{J(t)\}$ models crayfish harvesting as a shot noise process (5.6) in which $N(t)$ is the number of harvest events in the interval $[0, t]$, the harvest time $\{\tau_\ell\}_{\ell=1}^{N_\ell}$ and size $\{y_\ell\}_{\ell=1}^{N_\ell}$ are i.i.d. with exponential distribution of intensity $\rho = 0.2$ and mean $\mu = 0.075$, respectively, and $\delta = 0.25$. Since the noise processes may drive the system persistently outside $\mathcal{X}$, we consider the stopped process of (5.25) defined up to a stopping time $\tau := \inf\{t \in \mathbb{R}_+ \mid x(t) \in \partial\mathcal{X}\}$ so that $x(t) = 0, \forall t > \tau$.

In the absence of processes $\{w(t)\}$ and $\{J(t)\}$, model (5.25) has two stable equi-

libria: a bass-dominated equilibrium E_1 at $x = (0.19, 0.89)$ and a crayfish-dominated equilibrium E_2 at $x = (0.75, 0.5)$. Fig. 5.5a plots the isoclines, identifies the two stable equilibria, and marks the separatrix between the two ROAs. Fig. 5.5a also plots one realization of (5.25) when initialized from the ROA of E_1. This realization shows that each harvesting event causes a step decrease in the bass population, after which the system begins relaxing back to E_1. There is a finite probability that repeated harvesting events will drive the system state across the separatrix, whereupon the system's equilibrium state shifts to E_2. Assuming that the system starts inside a safe region $\mathcal{X}_s$ defined by the ROA of E_1 (shaded area in Fig. 5.5a), we are interested in bounding the probability that its sample paths leave (exit) the set $\mathcal{X}_s$ in a finite time interval $t \in [0, T]$.

A circle of radius 0.05 centered at $x = (0.01, 0.5)$ is used as initial set $\mathcal{X}_0 \in \mathcal{X}_s$. To bound the exit probability of $\{x(t)\}$ in (5.25) from $\mathcal{X}_s$, we implemented Algorithm 3 in MATLAB using software tools SOSOPT [134], SOSTOOLS [121] and SeDuMi [139]. A dual core 2.2GHz PC with 4GB RAM was used for simulations.

After shifting E_1's coordinate to the origin, we ran Stage 1 of Algorithm 3 and found a fourth order Lyapunov function $V(x)$ whose ROA's level set $\Omega_{V,\gamma} = \{x \in \mathcal{X} \mid V(x) \leq \gamma^*\}$ for a lower bound $\gamma^* = 6.96$ is plotted in Fig. 5.5a as a bold line ellipse. The closest intersection between $V(x)$'s level set and the separatrix is at $V(x) = \gamma = 14.55$ and so the probability $\mathbb{P}\{x(t) \notin \mathcal{X}_s, t \leq T\}$ of leaving $\mathcal{X}_s :=$ $\{x \mid V(x) < \gamma\}$ in a finite time T is no greater than $\mathbb{P}\{\sup_{0 \leq t \leq T} V(x(t)) \geq \gamma \mid x_0 \in \mathcal{X}_0\}$. We choose $\beta(t) = t/(c + t^4)$. Using the obtained $V(x)$ and setting $\underline{\epsilon} = 10^{-4}$ with $\epsilon_0 = 0.1$, we ran Stage 2 of Algorithm 3 and found $\alpha = 0.117$ and $c = 5.79 \times 10^3$. These results can then be used in (5.15) to bound $\mathbb{P}\{x(t) \notin \mathcal{X}_s, t \leq T\}$. Our simulation results indicate that both steps are solved optimally (feasibility ratio ≥ 1, duality gaps $\approx 10^{-10}$, cf. [139]).

Figure 5.5b (dashed curve UB(γ)) shows the probability bound obtained using

(a) Phase portrait and ROA. (b) Probability bound.
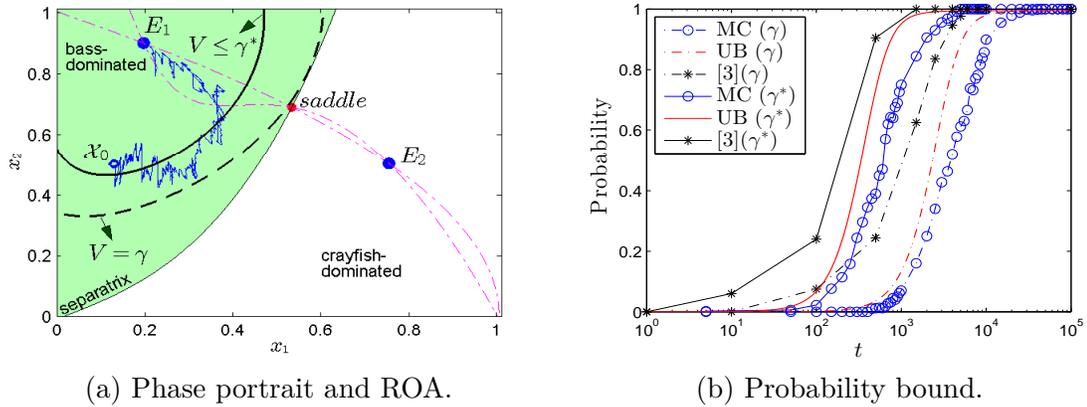
Figure 5.5. Phase portrait, ROA and probability bound.

the proposed method when evaluated over a finite time $T = 10^5$. This plot suggests that for $t \geq 2 \times 10^3$ and the specified harvesting parameters, the probability that the coexisting population fails to stay within the bass-dominated region is greater than or equal to 0.5. This information can be used to design lake management strategies. For example, in order to ensure the probability that the lake fails to maintain its safe bass-dominated state is lower than 0.5, a lake manager may choose to allow the sport harvesting of bass only up to time $t = 2 \times 10^3$. Alternatively, the manager can also choose to regulate the allowed harvesting parameters over different time intervals. The design of such management strategies is beyond the scope of this note and is left for future work.

Figure 5.5b also plots the estimates of the probability bounds obtained using the method described in [122] (dashed star curve $[122](\gamma)$) and from 500 realizations of a Monte Carlo (MC) simulation (dashed circle curve $\text{MC}(\gamma)$) based on the Euler-Maruyama scheme with 95% confidence interval and uniformly distributed initial condition $\mathcal{X}_0$. It can be seen that the result obtained using the proposed method upper bounds the MC simulation result (average difference ≈ 0.1 over the given T) and is tighter than that obtained using the method described in [122]. This suggests

125

that the use of less restrictive (time-varying) condition on the JDP's generator in our proposed method helps provide better upper bounds.

Note that Figure 5.5b also compares the obtained results when the safe set is $\mathcal{X}_s^* := \{x \,|\, V(x) < \gamma^*\}$ (solid lines [MC](γ^*), [UB](γ^*), [3](γ^*)). This plot suggests that an upper bound for the safety probability can still be obtained using the proposed method (possibly more conservative) even if the separatrix that defines the boundary of the system's true ROA is not known.

The main advantage of the SOS optimization method as compared to the MC simulation method can be seen in terms of the computation time and the memory usage. In our simulations, the computation of the MC(γ) bound in Figure 5.5b takes about eight minutes and uses up to 95 megabytes (MB) of memory whereas the computation of the SOS bounds takes an average of one minute and only use up to 15 MB of memory. This illustrates the effectiveness of the proposed method to verify the safety of a process without requiring exhaustive simulations of its sample paths.

5.6   Remarks and Future Works

This chapter presented a computational framework to predict the occurrence of noise-induced regime shifts for systems that are perturbed by jump/shock processes. Such predictions are formulated either as mean first passage time problem or as stochastic safety analysis whose solutions can be obtained using SOS optimization method. We presented examples of using the proposed method in the studies of ecosystem regime shift management.

Future works: One possible extension of the method discussed in this chapter is its application to predict large scale regime shifts or *phase transitions* in networks of interconnecting stochastic processes. The method presented in this chapter, however, cannot be applied directly to this large scale problem due to the limitation of the SOS optimization method which only capable of solving small to medium scale
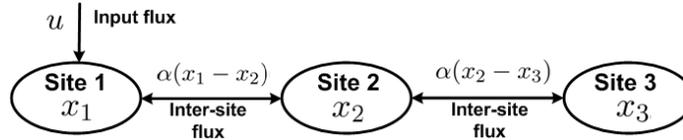
Figure 5.6. Network of lake systems.

problems. A method to address this issue is by using the *divide-and-conquer* method to decompose the network into smaller subsystems. If an appropriate decomposition of the network can be obtained, the method presented in this chapter can be used to analyze each subsystem.

To illustrate this approach, let us consider a hypothetical networked systems in Figure 5.6. This networked system assumes that each subsystem or site is governed by the scalar SDE of the lake eutrophication model discussed in Section 5.1. Thus, one may view this networked system as a spatial model for the Phosporus (P) concentration in three connected sites of a lake. The influx (u) of P from the environment enters the lake in site 1 and then distributed to other sites through inter-site fluxes of intensity α. The Wiener process $w_i(t)$, $(i = 1, \ldots, 3)$ in the model of each site illustrates stochasticity that occurs due to small variation in the amount of input or inter-site fluxes that enter each site. We assume that the drift term of the model of each site is bistable and that the intensity σ of the Wiener process in each site is relatively small such that each stable equilibria of the site is stochastically stable with probability 1. Moreover, the network is assumed to be *weakly coupled* so that the assumption that each subsystem is stochastically stable guarantees that the network system also stochastically stable [107, 108].

To analyze the system in Figure 5.6, we view the network as an interconnection of *isolated subsystems* that are coupled through the inter-site fluxes. In this case, the

dynamic of each site is governed by SDE of the form

$$dx_i(t) = [f_i(x_i(t)) + g(x_{-i}(t))]dt + \sigma dw_i(t), \quad (i = 1, 2, 3),$$

in which the subscript $-i \in \mathcal{N}_i$ denotes the ith site's nearest neighbors $\mathcal{N}_i$. This modeling approach results in the following SDE model of the network.

$$
\begin{aligned}
dx_1(t) &= \left(u - (1 + \alpha)x_1 + \frac{x_1^3}{\theta^3 + x_1^3} + \alpha x_2 \right) dt + \sigma dw_1(t), \\
dx_2(t) &= \left(-(1 + 2\alpha)x_2 + \frac{x_2^3}{\theta^3 + x_2^3} + \alpha(x_1 + x_3) \right) dt + \sigma dw_2(t), \qquad (5.26) \\
dx_3(t) &= \left(-(1 + \alpha)x_3 + \frac{x_3^3}{\theta^3 + x_3^3} + \alpha x_2 \right) dt + \sigma dw_3(t).
\end{aligned}
$$

Notice in model (5.26) that the last term in the drift part of the ith site's is only a function of its nearest neighbors' states. This suggests that networked system (5.26) can be viewed as a random process evolving on a graph and the dynamics of the network can be studied using Markov Random Field (MRF) formalism [86, 13].

Let $G = (S, E)$ be an undirected graph of networked system in Figure (5.6) which consists of a finite number of sites S whose edges $E \subset S \times S$ characterize the sites that are adjacent to each other. Consider a random process evolving on the graph G and let w_i be random variables describing the *configuration* of the ith site. In our case, the set of configurations are the two possible stable states that the system may have (i.e. oligotrophic and eutrophic states). Assume the probability of site ith configuration w_i taking some value can be specified as a conditional probability (or *local specification*) [86, 13] of the form

$$\mathbb{P}(w_i | w_j, j \in S) = \mathbb{P}(w_i | w_{-i}, -i \in \mathcal{N}_i). \qquad (5.27)$$

The above relation suggests that the probability of site i taking a particular configu-

ration w_i depends only on the configuration of its neighboring sites. In other words, the probability of the ith site's state to switch between the two possible configuration (i.e. from oligotrophic to eutrophic and vice versa) depends only on the size of the inter-site fluxes with its neighboring sites. This exactly is the stochastic safety problem discussed in the previous section and so the local specification in (5.27) can be evaluated using the SOS optimization techniques discussed in Section 5.3.

Figure 5.7 shows a preliminary result supporting the idea of using MRF *abstraction* to study regime shifts in coupled SDEs. The sample trajectories of the coupled SDEs (5.26) generated using direct integration method is plotted in Figure 5.7a. This figure is generated using an input flux $u = 0.1$ which is large enough to trigger regime shifts between the stable states of site 1. One may see from this figure that the shift from low to high P levels in site 1 is followed by the same shifts in sites 2 and 3, causing a *phase transition* of the network from low to high P concentrations. Figure 5.7b plots the Monte Carlo (MC) simulations of the MRF model discussed previously. The transition probability (5.27) used in the MC simulation of the MRF is constructed based on the solution of the stochastic safety analysis discussed in Section 5.3. One may see that the result from the MRF abstraction is capable of capturing

(a) Direct integration. (b) MRF abstraction.
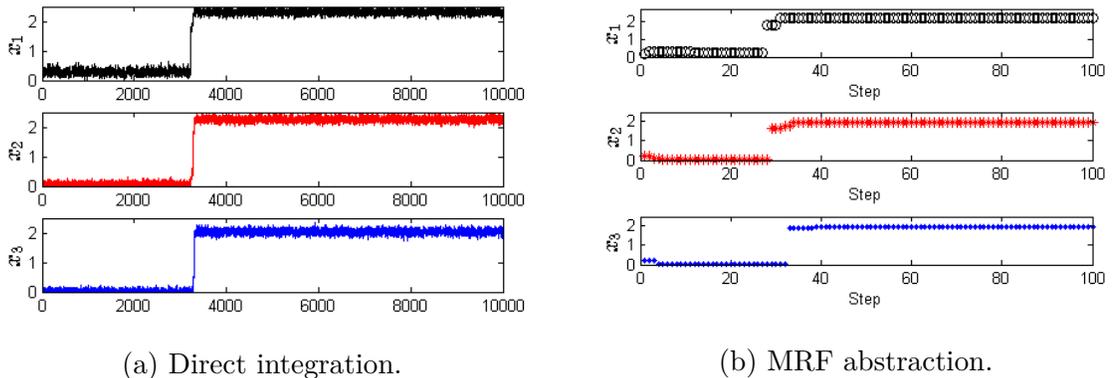
Figure 5.7. Simulation comparison of the coupled SDEs (5.26).

the *qualitative* properties of the network's transition from low to high P level. The main advantage in using the MRF abstraction is that its required computation effort is less than that required by direct integration method (i.e. simulation time of the MRF abstraction is 1/100 th of the simulation time required by the direct integration method). The MRF abstraction method therefore provides a means to reduce the computation cost required in analyzing a large scale interconnected SDEs.

Future research directions that could be pursued include the development of a theoretical framework for explaining the large scale regime shifts phenomena in interconnected systems as well as the construction of efficient algorithm that connects the SOS optimization method and the sampling methods in MC simulation of MRF.

CHAPTER 6

FORECASTING REGIME SHIFTS IN A MICROBIAL PREDATOR AND PREY
SYSTEM

This chapter describes an application of regime shift analysis in a mathematical model identified in an experiment on an ecological system test bed. The test bed is a laboratory scale chemostat that cultures a microbial predator and prey system. The prey is green algae, *Chlorella vulgaris* and the predator is a rotifer, *Brachionus calyciflorus*. The test bed was based on a similar experiment reported in [50, 157]. In that experiment, it was shown that this particular predator and prey system setup exhibited different alternative dynamics including population extinction, stable coexistence and sustained oscillation.

The initial objective of the work reported in this chapter was to evaluate the potential use of the D_2B analysis from Chapter 4 for studying regime shifts in models of real life systems. The chemostat experiment reported in [50, 157] is a suitable candidate for this purpose. This is not only because the experiment was relatively simple but also because the models of the *C. vulgaris* and *B. calyciflorus* interaction identified in [50, 157] have kinetic realizations.

To achieve our objective, a laboratory chemostat similar to that reported in [50, 157] was constructed. Using data obtained from the chemostat, we identified several differential equation models for the *C. vulgaris* and *B. calyciflorus* interaction. We found that the data was best predicted by models with multiple clones of *C. vulgaris* that differ in palatability. This finding was consistent with that suggested in [135, 157]. One of the identified models was then used to study regime shifts in the *C.*

vulgaris and *B. calyciflorus* interaction.

We first attempted to use the D_2B analysis to study the system's robustness under parametric variations. We found that the analysis could not be applied to the identified model because both the computation of the Gröbner basis and the characterization of the bifurcation conditions were intractable.

To address the limitation of the D_2B analysis, we developed another approach to study regime shifts [98]. The approach essentially formulated the regime shift analysis as robust stability analysis of affine parameter dependent systems [51]. This formulation was based on a modification of the method described in Chapter 4. The approach combined the effectiveness of linear matrix inequality (LMI) methods [12] and the symbolic-numerical algorithms [31]. The approach was applied to one of the identified models to predict regime shifts in the presence of parametric variation. We found that the analysis results from the proposed method provide lower bound for the distance to regime shifts obtained from direct numerical simulations of the system's ODE model. This suggests that the used method indeed provides a way of characterizing how robust a system may be to parametric variations and therefore can be used as a measure of the distance to regime shifts.

This chapter is structured as follows. Section 6.1 presents basic chemostat concept and the model of the predator and prey system. Section 6.2 describes the materials and methods used for constructing the chemostat test bed. Model identification of the *C. vulgaris* and *B. calyciflorus* interaction using maximum likelihood estimation method [123, 59] is also discussed. Section 6.3 presents the results on model identification and robustness analysis of the identified model. Section 6.4 gives final remarks.

6.1 Introduction

This section presents background on the chemostat and discusses a model for the predator and prey system to be cultured inside the chemostat. A chemostat is a well-mixed, continuous culture device that is often used in experiments to study the dynamics of nutrient-limited microorganisms or bacteria [70, 136]. It is a useful platform for hypothesis testing purposes because many environmental factors that affect microbial growth can be systematically controlled during the experiments.

6.1.1 Chemostat

Figure 6.1 illustrates the basic set up of a chemostat. The set up consists of three connected vessels namely the *feed*, the *culture* and the *collection* vessels. The feed vessel is filled with a media (liquid) that contains all nutrients that are required for bacterial growth. All of these nutrients are available in excess except one that is often referred to as the *limiting nutrient* [136]. By limiting a particular nutrient, the chemostat can be used to study how variations in this limiting nutrient affect the growth of the cultured bacteria. The culture vessel is the place where bacteria grow under the supply of media from the feed vessel. Other environmental factors such as light or sterile air are also provided to the culture vessel. The media in the culture is assumed to be well mixed in the sense that the content from the feed, the light illumination, and the air supply are uniformly dispersed throughout the vessel. The collection vessel is where the products of the culture are collected for measurement or other purposes.

The basic operation of a chemostat is as follows. The media in the feed vessel is pumped into the culture vessel at a steady state flow rate F and the media emerges from the culture vessel at the same rate. Thus, the residence time of a particle in the culture vessel with volume V is determined by the *dilution rate* $\delta = F/V$ which is defined as the number of complete volume-changes/hour [136]. This implies that the
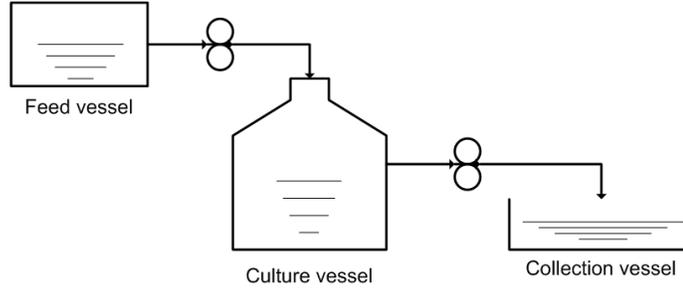
Figure 6.1. Basic schematic of a chemostat.

mean residence time of any particle in the culture vessel is equal to $1/\delta$. Thus, if the concentration of an organism in the culture vessel is denoted as x, then its *dilution rate* from the culture vessel is defined as $dx/dt = -\delta x$.

Now consider a single species bacteria that is cultured in the chemostat with a constant inflowing media containing one limiting nutrient N at concentration N_{in}. Assume that all other nutrients required for bacterial growth are available in excess within the media and that the culture vessel of the chemostat is well aerated and sufficiently illuminated. Assume further that an experimenter may only control (i) the concentration N_{in} of the limiting nutrient in the media supplied from the feed vessel and (ii) the dilution rate δ at which the the media emerges from the culture vessel. Using the law of mass conservation, the rate of change of the bacteria (x) and the limiting nutrient (N) concentrations in the culture vessel satisfy the following differential equation [136]

$$\begin{aligned}
\frac{dN}{dt} &= \delta(N_{in} - N) - \frac{\mu x}{\epsilon}\left(\frac{N}{K_x + N}\right), \\
\frac{dx}{dt} &= x\left(\frac{\mu N}{K_x + N} - \delta\right),
\end{aligned} \tag{6.1}$$

where μ denotes the rate of nutrient uptake by the bacteria and ϵ is an efficiency factor describing fraction of bacteria increase for the available amount of nutrient. Model (6.1) uses the Michaelis-Menten kinetics with a half saturation constant K_x

to describe the nutrient uptake by the bacteria. The differential equation (6.1) is a basic model for the dynamics of a single species bacteria cultured in the chemostat. It can be extended to model interactions of multiple bacteria that are simultaneously cultured in the chemostat.

6.1.2 *C. Vulgaris* and *B. Calyciflorus* Interaction in the Chemostat

One example of microbial interaction in the chemostat is the predator and prey interaction between *C. vulgaris* and *B. calyciflorus* [50]. *C. vulgaris* is a unicellular green algae that grows and multiplies through photosynthesis with the help of carbon dioxide, water, light, and some nutrients. The rotifer, *B. calyciflorus*, is a freshwater zooplankton whose growth rate depends on the available concentration of algae such as *C. vulgaris*. Culturing both *B. calyciflorus* and *C. vulgaris* simultaneously in a chemostat therefore sets a microbial predator and prey interaction. By limiting the concentration of a primary nutrient such as nitrogen in the supplied media, the growth of the prey species (*C. vulgaris*) will be limited by the available limiting nutrient concentration whereas the growth of the predator species (*B. calyciflorus*) will be limited by the available prey species concentration.

A model for the *C. vulgaris* and *B. calyciflorus* interaction was proposed in [50] and is given below.

$$
\begin{aligned}
\frac{dN}{dt} &= \delta(N_{in} - N) - F_C(N)C, \\
\frac{dC}{dt} &= F_C(N)C - F_B(C)\frac{B}{\epsilon} - \delta C, \\
\frac{dR}{dt} &= F_B(C)R - (\delta + \mu + m)R, \\
\frac{dB}{dt} &= F_B(C)R - (\delta + m)B.
\end{aligned}
\tag{6.2}
$$

In equation (6.2), variables N, C, R and B denote the concentrations of nitrogen (limiting nutrient), *C. vulgaris*, reproducing *B. calyciflorus* and nonreproducing *B. calyciflorus*, respectively, in the culture vessel. The parameters δ, N_{in} and ϵ are the

dilution rate of the chemostat, the nitrogen concentration in the supplied media and the conversion efficiency of the nutrient uptake by the *C. vulgaris*, respectively. Functions $F_C(N)$ and $F_B(C)$ denote the monoid type response functions of *C. vulgaris* and *B. calyciflorus*, respectively, defined as follows.

$$F_C(N) = \frac{b_C N}{K_C + N} \quad \text{and} \quad F_B(C) = \frac{b_B C}{K_B + C}.$$

The parameters b_C (b_B) and K_C (K_B) in these functions denote the maximum birth rate and half saturation constant of the *C. vulgaris* (*B. calyciflorus*), respectively. Note that model (6.2) classifies the rotifer population into those that can reproduce (R) and cannot reproduce (B). This classification was based on experiments reported in [50] where it was observed that the decline in rotifer population was not only caused by their exit from the culture (δ) but also due to the loss of fecundity while senescent. These additional loss effects are described in model (6.2) through parameters m and λ that describe the instantaneous mortality and the senescence rate of the rotifers, respectively.

Experimental data was used in [50] to estimate equation (6.2)'s parameters. In the experiment reported in [50], the media was set to have a limited nitrogen concentration and observations of both species concentrations were conducted for different values of dilution rates. The experimental observations suggested that the *C. vulgaris* and *B. calyciflorus* interaction result in a stable equilibrium where both the predator and the prey coexist at a steady state concentration. This occured at both low and high dilution rates. The system also exhbited a limit cycle behavior for medium dilution rate. As described in [50], model (6.2) correctly predicted these different dynamics for the corresponding values of dilution rates. However for the case of population cycle, model (6.2) failed to capture the observed period and phase of the population cycle.

Further analyses of the experimental data were conducted to find possible mechanisms that were excluded in model (6.2) [135]. In particular, four modifications of model (6.2) were proposed to account for the observed cycles. Each model encapsulated a hypothesis about the mechanism responsible for the feature of the cycles. The hypothesized mechanisms were (i) viability of rotifers' eggs increases with the available food source, (ii) algae's nutritional value increases with nitrogen availability, (iii) algal physiological state varies with toxins' accumulation in the chemostat and (iv) algae evolves in response to predation [135]. It was found that the fourth hypothesis on algae evolution best predicted the observed period and phase of the cycle [135, 76]. By incorporating this evolutionary effect into model (6.2), a modified model in (6.3) was shown in an experiment to resolve such a discrepency [157].

$$
\begin{aligned}
\frac{dN}{dt} &= \delta(V N_{in} - N) - \sum_{i=1}^{k} F_{C,i}(N)C_i, \\
\frac{dC_i}{dt} &= \eta_C F_{C,i}(N)C_i - F_{B,i}B - \delta C_i, \\
\frac{dR}{dt} &= \eta_B F_B(C_i)R - (\delta + m + \lambda)R, \\
\frac{dB}{dt} &= \eta_B F_B(C_i)R - (\delta + m)B,
\end{aligned}
\tag{6.3}
$$

where $i = 1, 2, \ldots, k$ denotes the k number of different clones in the *C. vulgaris* population and

$$
F_{C,i}(N) = \frac{b_C N}{K_C(p_i) + N}, \qquad F_{B,i} = \frac{b_B p_i C_i}{K_B + \sum_{i=1}^{k} C_i}, \qquad F_B(C_i) = \frac{\sum_{i=1}^{k} C_i p_i}{K_B + \sum_{i=1}^{k} C_i p_i}.
$$

The main difference between models (6.3) and (6.2) is that the *C. vulgaris* population in (6.3) is now partitioned into k number of different clones. Each clone is characterized by a *palatability* parameter $p_i \in [0, 1]$ which indicates its relative food value to the *B. calyciflorus*. Clones with low p_i are less prone to predation but are also less competitive than clones with higher p_i in term of nutrient utilization. This char-

acteristic therefore suggests a trade off between food value and competitive ability among clones in the *C. vulgaris* population [157].

6.2 Materials and Methods

This section describes the materials and methods used to construct a chemostat with similar setup as that reported in [50, 157]. A maximum likelihood estimation method [123, 59] for identifying differential equation models for the *C. vulgaris* and *B. calyciflorus* interaction is then presented.

6.2.1 Materials

Figure 6.2 shows the constructed chemostat and its schematic. The set up consists of one feed vessel that pumps media containing the limiting nutrient (i.e. nitrogen) into five plastic culture vessels with volume $V = 400$ mL. The media emerging from each culture vessel is collected in a collection vessel.

The feed vessel continuously supplies the media throughout the experiment. The media is a 50% Bold's Basal Medium (BBM-Medium, modified) [75] with a limited 100 micromolar (μM) nitrogen concentration and additional 5 gram per liter (g/L) NaCl crystal (see Appendix D.1). The dilution rates of the media from the feed to the cultures are adjusted using two peristaltic pumps (Cole-Palmer's Masterflex L/S). One pump sets a low dilution rate to three culture vessels (C1 - C3) while another pump sets a medium dilution rate to the other two culture vessels (C4 - C5).

Continuous light illumination and inflow of sterile air are provided to each culture vessel to prevent light or CO_2 limitation in microbial growth. Each culture vessel is equipped with a magnetic stirrer bar to ensure complete mixing of the nutrients and the microorganisms inside it.

The *C. vulgaris* (Item # 151955) and *B. calyciflorus* (Item # 162860) species used in the experiment were obtained from the Carolina Biological Supply Company
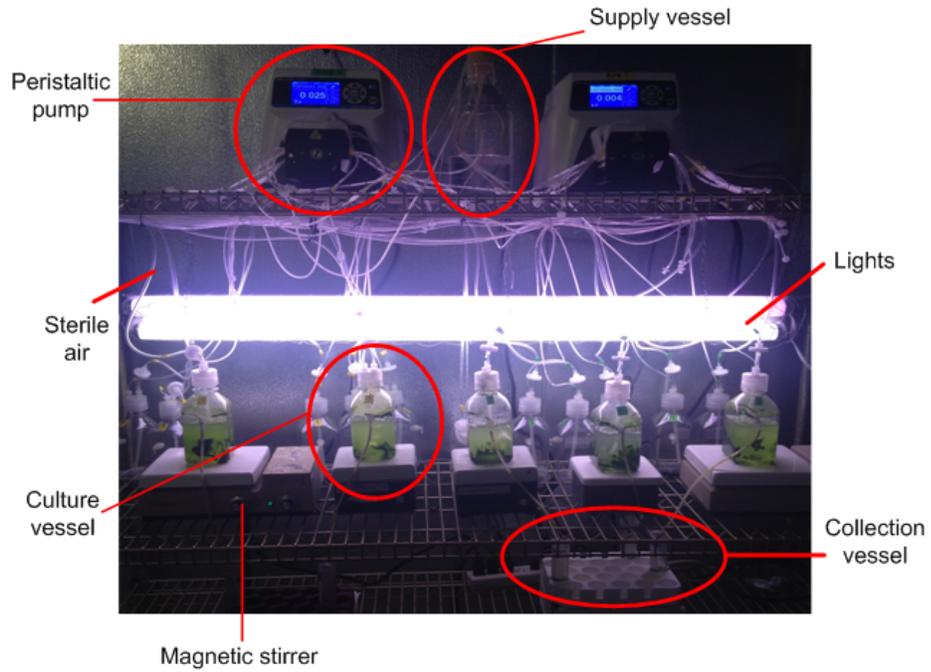
(a) The constructed chemostat.
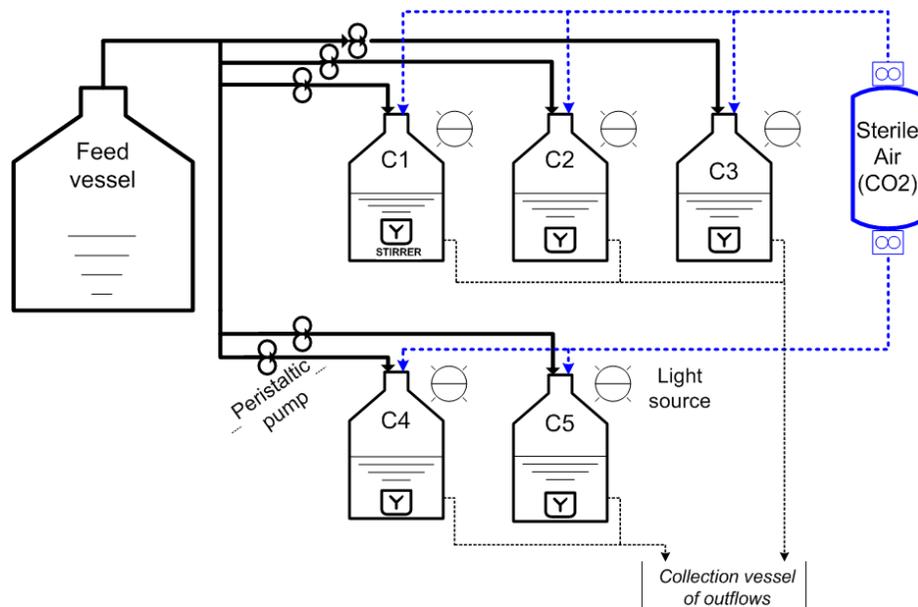
(b) Schematic of the chemostat.

Figure 6.2. Chemostat used in the experiment.

(`http://www.carolina.com`).

6.2.2   Methods

6.2.2.1   Experiment and Measurement

Experiment was started by setting both supply pumps at low dilution rates ($\delta = 0.04\ V$/day). Five mL *C. vulgaris* were added to each culture vessel and the chemostat was left to operate for five days until each culture contained a high density of *C. vulgaris* ($\geq 10^6$ cell/mL). One of the pumps (supplying C1 - C3) was then switched to a dilution rate $\delta = 0.1\ V$/day while the other (supplying C4 - C5) was set to a dilution rate $\delta = 0.95\ V$/day. The chemostat was then left to run for another five days to ensure that the cultures still contained a high density of *C. vulgaris* ($\geq 10^6$ cell/mL) for the two dilution rates. After that, five (5) mL *B. calyciflorus* (1 mL ≈ 20 cells) were added to the culture vessels C2 - C5. Note that the *B. calyciflorus* was not added into the culture vessel C1 because the vessel was only used to stock the *C. vulgaris* population.

The measurements were done daily for 45 days to obtain a measure of the daily variation in the population of both microorganisms. The measurements were obtained from culture vessels C2 - C5 by sampling through their top covers. The samples were taken using serological pippette. For each sample, the number of cells of both *C. vulgaris* and *B. calyciflorus* were measured. The total number of *B. calyciflorus* cells were counted using a dissecting microscope whereas the total number of *C. vulgaris* cells were counted using a compound microscope. Figure 6.3 plots the data for the two dilution rates (cf. Appendix D.2 for the raw data). For each dilution rate, the data were averages of the measurements from the corresponding two culture vessels. Figure 6.3 shows that both microorganisms coexisted in the chemostat for the two dilution rates. For a low dilution rate $\delta = 0.1$, the microorganisms coexisted in a stable steady state. For a medium dilution rate ($\delta = 0.95$), the microorganisms

(a) Dilution rate $\delta = 0.1$.   (b) Dilution rate $\delta = 0.95$.
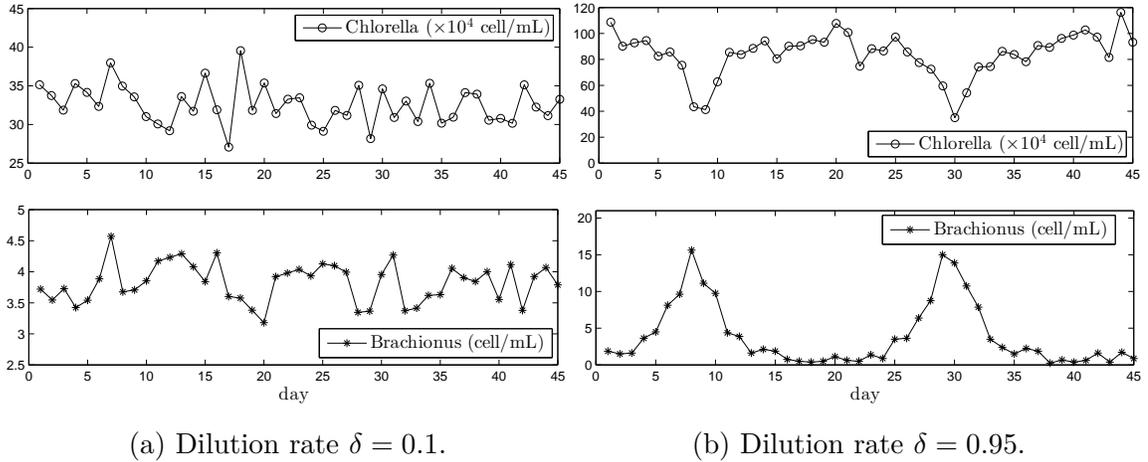
Figure 6.3. Measurements of *C. vulgaris* and *B. calyciflorus*: (a) $\delta = 0.1$,
(b) $\delta = 0.95$.

coexisted in a limit cycle. These observations were in agreement with that reported
in [50]. This data set was used to estimate the parameters in models (6.2) and (6.3).

6.2.2.2 Model Identification

The data obtained from the chemostat was used in model identification. The
model identification was done to estimate the parameters of equations (6.2) and
(6.3). This section describes a maximum likelihood estimation method [123, 59] that
was used to fit models (6.2) and (6.3) to the data.

Consider a dynamical system with n state variables $x(t) = [x_1, \ldots, x_n]^T$ and p
parameters $k = [k_1, \ldots, k_p]$. Assume that the system is modeled as a parameterized
ordinary differential equation (ODE)

$$\dot{x}(t) = f(x(t), k), \quad x(0) = x_0. \tag{6.4}$$

It is often the case that only some of the state variables can be measured directly
in the experiment. In particular, it can be the case that only combination or rel-

ative quantities of some state variables that can be measured. This is true in our experiment where only the total number of rotifer cells ($R + B$ in equation (6.1)) were measured rather than the individual number of reproducible (R) and nonreproducible (B) rotifer cells. In order to take such cases into consideration, let us define an observation function $g : \mathbb{R}^m \to \mathbb{R}^m$ that maps the state variables $x(t)$ to a set of m measurement $y(t) = [y_1, \ldots, y_m]^T$ such that

$$y(t) = g(x(t), \eta) \tag{6.5}$$

where η denotes the measurement parameter. The state and observation equations in (6.4) and (6.5), respectively, therefore take into account the case when the system states are only partially observable, i.e. when $m < n$. In reality, the measurements are usually corrupted with noise. Thus, if y_{ij} denotes the jth measurement data taken at time t_i, $(i = 1, \ldots, T)$, one may assume that it contains the true measurement $y_j(t_i)$ and a random process $e_j(t_i)$, i.e.

$$y_{ij} = y_j(t_i) + e_j(t_i). \tag{6.6}$$

In practical applications, the process $e_j(t_i)$ is often assumed to be an i.i.d. random variables with a Gaussian distribution [123].

Given the system model in (6.4)-(6.5) and the measurement data of the form (6.6), the maximum likelihood estimation (MLE) method seeks an optimal parameter k^* that maximizes the likelihood of the data y for the given parameter k [123]. This is achieved by searching for parameter k that maximizes the *likelihood function*. If the measurement error has a Gaussian distribution with a zero mean and a time-dependent variance σ_{ij}^2, then the likelihood function L is given by

$$L(y|k) = \prod_{i=1}^{T} \prod_{j=1}^{m} \frac{1}{\sigma_{ij}\sqrt{2\pi}} \exp\left(-\frac{1}{2}\frac{(y_{ij} - g_j(x(t_i, k), k))^2}{\sigma_{ij}}\right) \tag{6.7}$$

142

In practice, the maximization of the likelihood function L is done through minimization of its negative logarithm, i.e.

$$- \log \left[L(y|k) \right] = \sum_{i=1}^{T} \sum_{j=1}^{m} \frac{1}{2} R_{ij}(k)^2 + \log \left[\sigma_{ij} \sqrt{2\pi} \right], \tag{6.8}$$

where

$$R_{ij}(k) = \frac{y_{ij} - g_j(x(t_i, k), k)}{\sigma_{ij}} \tag{6.9}$$

is often called the residual function [123]. Thus, the optimal parameter, k^* obtained by the MLE method is given by

$$k^* := \arg \min_{k} \sum_{i=1}^{T} \sum_{j=1}^{m} \frac{1}{2} R_{ij}(k)^2. \tag{6.10}$$

Note that the second term on the right hand side of (6.8) can be excluded in the minimization (6.10) as it is independent of the unknown parameter k.

One approach to solve the minimization in (6.10) uses the gradient-based optimization method described in [123, 59]. In this approach, parameter updates that minimize (6.10) are computed iteratively. These parameter updates are obtained based on the gradient of the residual R_{ij} with respect to parameter k that is computed at each step of the iteration. In particular, the gradient of the R_{ij} is defined as

$$\begin{aligned} \frac{\partial}{\partial k} R_{ij}(k) &= \frac{-1}{\sigma_{ij}} \frac{\partial g_j(x(t_i, k), k)}{\partial k} \\ &= \frac{-1}{\sigma_{ij}} \left(\sum_{i=1}^{n} \frac{\partial g_j}{\partial x_n} \bigg|_{t_i} \frac{\partial x_n}{\partial k} \bigg|_{t_i} + \frac{\partial g_j}{\partial k} \bigg|_{t_i} \right). \end{aligned} \tag{6.11}$$

Note that the terms $\frac{\partial g_j}{\partial x_n}$ and $\frac{\partial g_j}{\partial k}$ are the Jacobian matrices of the output equation (6.5). Since the form of function (6.5) is specified, these Jacobians can be computed symbolically beforehand. The term $\frac{\partial x_n}{\partial k}$ in (6.11) is known as the sensitivity S_k of the state variables of (6.1). Since the analytical solution $x(t, k)$ of (6.1) is gener-

ally unknown, the sensitivity S_k is usually obtained by numerically integrating the sensitivity equation defined below [123]

$$\frac{dS_k}{dt} = \sum_{i=1}^{n} \frac{\partial f}{\partial x_i} \frac{\partial x_i}{\partial k} + \frac{\partial f}{\partial k} \qquad (6.12)$$

in parallel while numerically integrating the ODE in (6.1). At each integration step, the value of the residual's gradient (6.12) is computed. Algorithm 4 summarizes the gradient-based optimization for the minimization of (6.10) [59]. The minimization in Step 3 of Algorithm (4) can be done using function 'lsqnonlin' in MATLAB.

Algorithm 4 Minimization of (6.10) [59]

Input: Models (6.4)-(6.5) and an initial parameter k

    LOOP

1: Using the current parameter value, integrate ODEs in (6.1) and (6.12).
2: Compute the residual R_{ij} in (6.9) and the gradient in (6.11)
3: Compute k^* in (6.10) using gradient-based optimization method (such as function lsqnonlin in MATLAB)
4: **if** convergence criteria is satisfied **then**
5:     **BREAK**
6: **else**
7:     Update the parameter vector k
8: **end if**

    ENDLOOP

6.3 Results

6.3.1 Model Identification

The MLE method described in Section 6.2.2.2 was used to fit the *C. vulgaris* and *B. calyciflorus* measurements to models (6.2) and (6.3). Model (6.2) will be called the single clone model (as it does not distinguishes different clones of the *C. vulgaris* population) while model (6.3) will be called the multi clone model. We further considered two types of the multi clone models: one which assumes there are two different clones (2-clone model) and one which assumes there are three different clones (3-clone model) in the *C. vulgaris* population. In total, three model candidates were fitted to the data.

For the 2-clone model, the corresponding ODE model is

$$
\begin{aligned}
\dot{N} &= \delta(N_{in} - N) - \frac{b_C N C_1}{K_{c1} + N} - \frac{b_C N C_2}{K_{c2} + N} \\
\dot{C_1} &= \frac{b_C N C_1}{K_{c1} + N} - \frac{p_1 b_B C_1 B}{\epsilon(K_B + p_1 C_1 + p_2 C_2)} - \delta C_1 \\
\dot{C_2} &= \frac{b_C N C_2}{K_{c2} + N} - \frac{p_2 b_B C_2 B}{\epsilon(K_B + p_1 C_1 + p_2 C_2)} - \delta C_2 \\
\dot{R} &= \frac{b_B(p_1 C_1 + p_2 C_2)}{K_B + p_1 C_1 + p_2 C_2} R - (\delta + m + \alpha) R \\
\dot{B} &= \frac{b_B(p_1 C_1 + p_2 C_2)}{K_B + p_1 C_1 + p_2 C_2} B - (\delta + m) B.
\end{aligned}
\tag{6.13}
$$

In model (6.13), we assumed that the two clones have extremely different palatability values: the first clone (C_1) has a high palatability ($p_1 = 1$) while the second clone (C_2) has a low palatability ($p_2 = 0.15$). These values of p_1 and p_2 were chosen arbitrarily

during model identification. For the 3-clone model, the corresponding ODE model is

$$\dot{N} = \delta(N_{in} - N) - \frac{b_C N C_1}{K_{c1} + N} - \frac{b_C N C_2}{K_{c2} + N} - \frac{b_C N C_3}{K_{c3} + N}$$

$$\dot{C}_1 = \frac{b_C N C_1}{K_{c1} + N} - \frac{p_1 b_B C_1 B}{\epsilon(K_B + p_1 C_1 + p_2 C_2 + p_3 C_3)} - \delta C_1$$

$$\dot{C}_2 = \frac{b_C N C_2}{K_{c2} + N} - \frac{p_2 b_B C_2 B}{\epsilon(K_B + p_1 C_1 + p_2 C_2 + p_3 C_3)} - \delta C_2$$

$$\dot{C}_3 = \frac{b_C N C_3}{k_{c3} + N} - \frac{p_2 b_B C_3 B}{\epsilon(K_B + p_1 C_1 + p_2 C_2 + p_3 C_3)} - \delta C_3 \qquad (6.14)$$

$$\dot{R} = \frac{b_B(p_1 C_1 + p_2 C_2 + p_3 C_3)}{K_B + p_1 C_1 + p_2 C_2 + p_3 C_3} R - (\delta + m + \alpha)R$$

$$\dot{B} = \frac{b_B(p_1 C_1 + p_2 C_2 + p_3 C_3)}{K_B + p_1 C_1 + p_2 C_2 + p_3 C_3} R - (\delta + m)B.$$

In model (6.14), we also assumed that the three clones have different palatability values: the first clone (C_1) has high palatability ($p_1 = 1$), the second clone (C_2) has a medium palatability ($p = 0.5$) and the third clone (C_3) has a low palatability ($p_3 = 0.15$). The MLE method described in Section 6.2.2.2 was used to estimate the parameters in each model. The initial guesses of the parameters used in model identification were $b_C = 3.3$, $K_{C_1} = 4.3$, $b_B = 2.25$, $K_B = 15$, $\epsilon = 0.25$, $m = 0.055$, $\alpha = 0.4$. These were values reported in [50, 157].

Table 6.1 summarizes the estimated parameters of each model for data with dilution rate $\delta = 0.95$. Figure 6.4 compares the trajectories of each estimated model and the corresponding data. One may see in Figure 6.4a that the estimated single clone model poorly predicted the cycle period shown by the data. Nevertheless, the model correctly captures the oscillatory feature of the data for the specified dilution rate (δ) and the media's nutrient content (N_{in}). This result is in agreement with that reported in [50] where similar characteristics of the fitted model were observed. On the other hand, the estimated 2-clone and 3-clone models fit well with the data. Both models correctly captured the cycle periods observed in the data. Based on these results, we concluded that the *C. vulgaris* population used in our experiment

TABLE 6.1

ESTIMATED PARAMETERS IN DIFFERENT MODELS ($\delta = 0.95$).

MODEL	δ	N_{in}	b_C	K_{C_1}	b_B	K_B	ϵ	m	α	K_{C_2}	K_{c3}
1-clone	0.95	100	7.15	11.16	4.72	9.99	0.066	0.0003	1.806	-	–
2-clone	0.95	100	2.73	6.75	2.58	25.39	0.304	0.0025	0.406	11.24	–
3-clone	0.95	100	2.33	6.08	3.03	21.78	0.45	0.005	0.778	8.15	10.29

contain at least two different clones.

Due to its simplicity, the single clone model was chosen for regime shifts analysis presented in the next section. The analyses of the other identified models are left for future work.

6.3.2 Distance to Regime Shifts

This section describes regime shifts analysis in a kinetic realization of the single clone model (6.2). This analysis searches for parameter values that give rise to a shift in the the system's qualitative behavior. A necessary condition for this shift to occurs is a change in the stability type of any of its equilibria. This section will be focused on predicting the occurrence of such a shift when the dilution rate, δ of the chemostat is varied. This is because the chemostat's dilution rate can be easily varied in the experiment. Furthermore, its impact on the *C. vulgaris* and *B. calyciflorus* interaction can be studied/observed from the measurement data.

In our analysis, we first constructed a kinetic realization for model (6.2). We
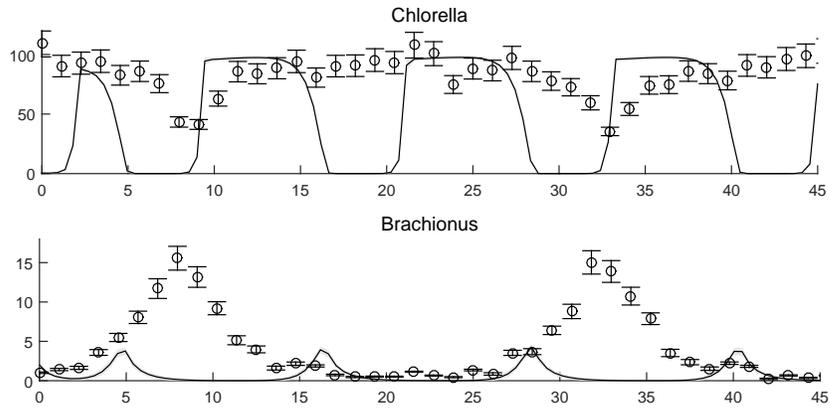
(a) Identified single clone model.
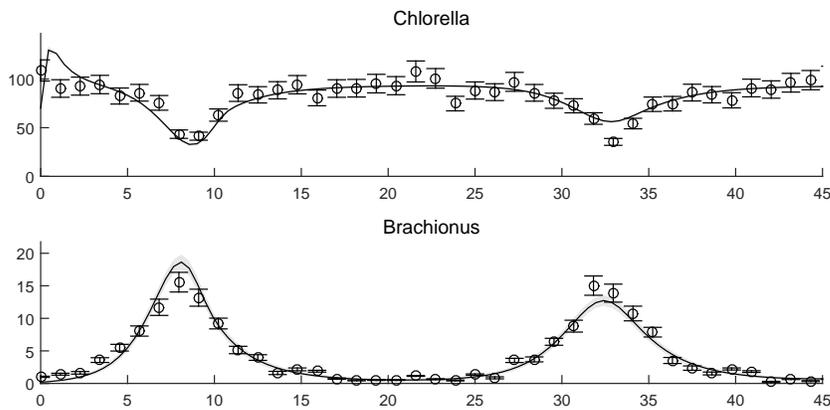
(b) Identified 2-clone model.
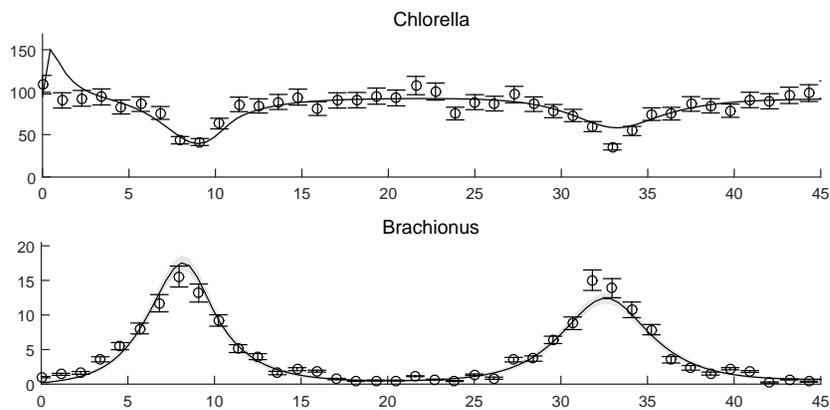
(c) Identified 3-clone model.

Figure 6.4. Comparison between the data and the identified models.

148

introduced a time scaling $\tau = t/(K_C + N)(K_B + C)$ and defined

$$\dot{x}_1 = \frac{dN}{d\tau}, \quad \dot{x}_2 = \frac{dC}{d\tau}, \quad \dot{x}_3 = \frac{dR}{d\tau}, \quad \dot{x}_4 = \frac{dB}{d\tau}. \tag{6.15}$$

The ODE (6.2) can then be written in a time scaling form below

$$
\begin{aligned}
\dot{x}_1 &= \delta(N_{in} - x_1)(K_C + x_1)(K_B + x_2) - b_C(K_B + x_2)x_1 x_2, \\
\dot{x}_2 &= b_C(K_B + x_2)x_1 x_2 - \frac{b_B}{\epsilon}(K_C + x_1)x_2 x_4 - \delta(K_C + x_1)(K_B + x_2)x_2, \\
\dot{x}_3 &= b_B(K_C + x_1)x_2 x_3 - (\delta + m + \alpha)(K_C + x_1)(K_B + x_2)x_3, \\
\dot{x}_4 &= b_B(K_C + x_1)x_2 x_3 - (\delta + m)(K_C + x_1)(K_B + x_2)x_4.
\end{aligned}
\tag{6.16}
$$

We further defined the following parameters

$$
\begin{aligned}
&k_1 = \delta, && k_2 = \delta N_{in}, && k_3 = b_C, \quad k_4 = K_C, \\
&k_5 = b_B, && k_6 = K_B, && k_7 = b_B/\epsilon, \quad k_8 = \delta + m, \\
&k_9 = \delta + m + \alpha, && k_{10} = \delta N_{in} - b_C K_B - \delta K_C, && k_{11} = b_C - \delta, \\
&k_{12} = (b_C - \delta)K_B, && k_{13} = (b_B - (\delta + m + \alpha))K_C, && k_{14} = b_B - (\delta + m + \alpha), \quad (6.17)
\end{aligned}
$$

such that model (6.16) is reparameterized into the following polynomial system.

$$
\begin{aligned}
\dot{x}_1 &= k_2 k_4 k_6 + k_2 k_4 x_2 + k_2 k_6 x_1 + k_{10} x_1 x_2 - x_1(k_1 k_4 k_6 + k_1 x_1 x_2 + k_1 k_6 x_1 + k_3 x_2^2), \\
\dot{x}_2 &= k_{11} x_1 x_2^2 + k_{12} x_1 x_2 - x_2(k_1 k_4 x_2 + k_1 k_4 k_6 + k_4 k_7 x_4 + k_7 x_1 x_4), \\
\dot{x}_3 &= k_{13} x_2 x_3 + k_{14} x_1 x_2 x_3 - x_3(k_4 k_6 k_9 + k_6 k_9 x_1), \\
\dot{x}_4 &= k_4 k_5 x_2 x_3 + k_5 x_1 x_2 x_3 - x_4(k_4 k_6 k_8 + k_4 k_8 x_2 + k_6 k_8 x_1 + k_8 x_1 x_2).
\end{aligned}
\tag{6.18}
$$

It can be seen that system (6.18) is a mass action system (cf. [67]) and so its kinetic realization exists and can be constructed. In our analysis, we constructed a kinetic

realization of the form

$$\dot{x} = Nv(x,k) = N\mathrm{diag}(k)x^Z, \tag{6.19}$$

where the corresponding matrix N and vector $v(x,k)$ are

$$N = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 1 & -1 & 0 & 0 & -1 & 1 \end{bmatrix}$$

$$\begin{matrix} -1 & 1 & 1 & 1 & -1 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{matrix}, \tag{6.20}$$

$$v(x,k) = [k_{14}x_1x_2x_3,\ k_{13}x_2x_3,\ k_{12}x_1x_2,\ k_{11}x_1x_2^2,\ k_{10}x_1x_2,\ k_8x_1x_2x_4,\ k_7x_1x_2x_4,\ k_6k_9x_1x_3$$

$$k_6k_8x_1x_4,\ k_5x_1x_2x_3,\ k_4k_8x_2x_4,\ k_4k_7x_2x_4,\ k_4k_6k_9x_3,\ k_4k_6k_8x_4,\ k_4k_5x_2x_3,\ k_3x_1x_2^2$$

$$k_2k_6x_1,\ k_2k_4x_2,\ k_2k_4k_6,\ k_1x_1^2x_2,\ k_1k_6x_1^2,\ k_1k_4x_2^2,\ k_1k_4k_6x_2,\ k_1k_4k_6x_1]^T. \tag{6.21}$$

The matrices $\mathrm{diag}(k)$ and Z such that $v(x,k) = \mathrm{diag}(k)x^Z$ are

$$\mathrm{diag}(k) = \mathrm{diag}\Big([k_{14},\ k_{13},\ k_{12},\ k_{11},\ k_{10},\ k_8,\ k_7,\ k_6k_9k_6k_8,\ k_5,\ k_4k_8,\ k_4k_7,\ k_4k_6k_9,$$

$$k_4k_6k_8,\ k_4k_5,\ k_3, k_2k_6,\ k_2k_4x_2,\ k_2k_4k_6,\ k_1,\ k_1k_6,\ k_1k_4,\ k_1k_4k_6,\ k_1k_4k_6]\Big), \tag{6.22}$$

150

$$Z = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 2 & 2 & 0 & 0 & 1 \\ 1 & 1 & 1 & 2 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 2 & 0 & 1 & 0 & 1 & 0 & 2 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{6.23}$$

The D_2B analysis described in Chapter 4 can now be applied to (6.19)-(6.23).

6.3.2.1 Limitation of the D_2B Analysis

Recall from Chapter 4 that the computation of the minimum D_2B for a kinetic realization (6.19) involves two main steps.

- The first step is to compute the expression for the flux equilibria $v^*(k, \lambda)$ from which the state equilibria $x^*(k, \lambda)$ can be obtained. In particular, the flux equilibria v^* are computed from the intersection $v^* := \mathcal{K}_v \cap \mathbb{V}(\mathcal{I})$ between the flux equilibria defined by a convex polyhedral cone $\mathcal{K}_v$

$$v^* \in \mathcal{K}_v := \ker(N) \cap \mathbb{R}_{\geq 0}^m = \Sigma_\ell \lambda_\ell E_\ell, \quad \ell = 1, \ldots, q, \tag{6.24}$$

and the flux equilibria defined by the toric variety $\mathbb{V}(\mathcal{I})$

$$v^* \in \mathbb{V}(\mathcal{I}) := \{v : \mathbb{G}(\mathcal{I}) = 0\} \tag{6.25}$$

of a binomial ideal $\mathcal{I} := \langle v - v(x, k) \rangle \subset \mathbb{R}(k)[v]$. This means that the computation of v^* requires the variety of the binomial ideal $\mathcal{I}$ contains only finitely many points. In other words, $\mathcal{I}$ has to be a zero dimensional ideal [25]. An ideal $\mathcal{I} \subset \mathbb{R}(k)[v_1, \ldots, v_m]$ with a Gröbner basis $\mathbb{G}(\mathcal{I})$ is said to be zero dimensional if and only if for each $1 \leq i \leq m$ there exists an element in $\mathbb{G}(\mathcal{I})$ whose leading terms is a pure power of v_i [25].

- The second step is to characterize the necessary bifurcation condition from the Jacobian

$$J(\lambda, h) = N \text{diag}(E\lambda) Z^T \text{diag}(h), \tag{6.26}$$

where E and λ, respectively, are the matrix of extreme generators and its associated vector of convex parameters that define $\mathcal{K}_v$ in (6.24) and $h = 1/x^*$ where x^* is the state equilibria obtained from the first step. The obtained condition is then used to formulate an SOS program for computing the minimum D_2B.

We first attempted to use the above method to compute the minimum D_2B for system (6.18). We found that it could not be applied to system (6.18) due to the following

difficulties:

- For the kinetic realization in (6.19)-(6.23), the associated binomial ideal $\mathcal{I} := \langle v - v(x,k) \rangle$ was not a zero dimensional ideal. This made computing an analytical expression for the variety $\mathbb{V}(\mathcal{I})$ in (6.25) intractable. As a result, the flux equilibria defined by the intersection $v^* := \mathcal{K}_v \cap \mathbb{V}(\mathcal{I})$ could not be computed.

- For the kinetic realization in (6.19)-(6.23), there were 36 extreme generators (cf. Appendix D.3, equation (D.1)) and so the convex polyhedral cone $\mathcal{K}_v$ in (6.24) was also formed by 36 convex parameters (cf. Appendix D.3, equation (D.2)). As a result, the Jacobian matrix (6.26) was a dense matrix with 36 variables (cf. equation (6.35)). The coefficients of this Jacobian's characteristic polynomials were very long symbolic expressions. This not only made characterizing necessary bifurcation conditions impossible but also caused computing the minimum $D_2 B$ using semidefinite programming tools intractable.

Because of these difficulties, we proposed another approach to study regime shifts in system (6.18). The approach used a robust asymptotic stability analysis method described in [98]. The analysis takes advantage of the Jacobian reparameterization (6.26) in terms of the convex parameter, λ. Under this parameterization, the associated linear system is a linear affine parameter dependent (APD) system whose robust stability can be examined using linear matrix inequalities (LMI) [51]. In this approach, a regime shift was defined as an event in which the robust stability of a system's nominal operating point could no longer be assured.

6.3.3 Robust Asymptotic Stability of Kinetic Realizations

As mentioned in Section 6.3.2, a necessary condition for the shift in a system's qualitative behavior is a change in the stability type of its equilibria. This section examines the local asymptotic stability of a kinetic realization's equilibrium point under parametric variations. The analysis uses the Jacobian reparamerization in (6.26) and combined the effectiveness of linear matrix inequality (LMI) methods [51, 12] and the symbolic-numerical algorithms for solving polynomial equations [31].

6.3.3.1   Robust Stability of Affine Parameter Dependent Systems

Note that the Jacobian matrix in (6.26) is bilinear with respect to the convex parameter λ and the reciprocal of the state equilibria $1/x^*$. We argue that (6.26) can be reparameterized to be affine in a new parameter. In particular, we introduce a parameter vector, μ, whose elements are formed from the product of the convex parameters, λ, and the reciprocal of the state equilibria, $h = 1/x^*$. More specifically, let us introduce the following notational convention

$$I_h = \text{diag}(1/x^*), \qquad \tilde{J} = N\text{diag}(E\lambda)Z^T \qquad (6.27)$$

so that the Jacobian (6.26) may be written as $J(\lambda, x^*) = \tilde{J}(\lambda)I_h$. With this change of notation, the linearized system equations becomes

$$\frac{d}{dt}(x - x^*) = \tilde{J}(\lambda)I_h(x - x^*). \qquad (6.28)$$

Introducing a change of variables, $z - \mathbf{1} = I_h(x - x^*)$ and noting that $\dot{z} = I_h\dot{x}$, the linearized system can be rewritten as

$$\dot{z} = I_h\tilde{J}(\lambda)z \qquad (6.29)$$

with state equilibrium at $z^* = \mathbf{1}$.

Given the system in (6.29), we introduce parameters, $h_i = 1/x_i^*$ for $i = 1, \ldots, n$ and let λ_ℓ be the system's convex parameters ($\ell = 1, \ldots, q$). Note that I_h is a diagonal matrix in h and that $\tilde{J}(\lambda)$ is an APD matrix with respect to λ. If we let M denote the set of parameters μ_j ($j = 1, \ldots, s$) formed from the products $h_i\lambda_\ell$ that appear in $I_h\tilde{J}(\lambda)$, then it should be apparent that $I_h\tilde{J}(\lambda)$ will also be an APD matrix with respect to the new parameters $\mu_j \in M$. So there will exist $s = (qn + 1)$ real $n \times n$

matrices $A_0, A_1, \ldots, A_s$ such that

$$I_h \tilde{J} = A_0 + \mu_1 A_1 + \cdots + \mu_s A_s \equiv A(\mu), \qquad (6.30)$$

thereby establishing that the linearized system in (6.29) may be rewritten as a linear APD system

$$\dot{z} = A(\mu)z \qquad (6.31)$$

with respect to the new parameter set μ.

Remark 6.3.1. In term of kinetic realizations (6.19), the new parameters μ are *elementary flux mode* (EFM) levels normalized with respect to the nominal state equilibria x^*. In particular, each EFM controls the fate of a particular species in the system [133].

The work in [51] established sufficient conditions for the APD system in (6.31) to be asymptotically stable for all parameters μ defined within a polytope, Γ. These conditions are in the form of linear matrix inequalities (LMIs) characterizing a parameter dependent function,

$$P(\mu) = P_0 + \mu_1 P_1 + \cdots + \mu_s P_s \qquad (6.32)$$

that serves as a Lyapunov function at each vertex of the parameter polytope, Γ. This leads to the following proposition from [51] which is stated without proof.

Proposition 6.3.2 ([51]). *Consider the APD system in equations (6.30)-(6.31) with time-invariant nomional parameter μ_0. Let Γ be a polytope in the parameter space defined by vertices $\gamma_1, \gamma_2, \ldots, \gamma_N$ that contain the parameter μ_0. This system's state equilibrium is asymptotically stable if $A(\mu_0)$ is Hurwitz and there exist $s+1$ symmetric matrices $P_0, P_1, \ldots, P_s$ such that the parameter dependent matrix $P(\mu)$ in equation*

(6.32) satisfies

$$A(\gamma_i)^T P(\gamma_i) + P(\gamma_i)A(\gamma_i) < 0 \tag{6.33}$$

for all $i = 1, 2, \ldots, N$ and

$$A_i^T P_i + P_i A_i \geq 0 \tag{6.34}$$

for all $i = 0, 1, 2, \ldots, s$.

Remark 6.3.3. The sufficient conditions in equations (6.33)-(6.34) are LMIs with with matrix variables $P_0, P_1, \ldots, P_s$. Since LMI problems are a class of convex optimization problem for which efficient interior point algorithm exist [12], it means it is computationally practical to evaluate the robust stability of kinetic realizations with a relatively large number of normalized flux parameters, μ. Moreover, this problem can be solved using one of the standard function in MATLAB's LMI toolbox. This function, `pdlstab`, is described in [51] and is used in the next section to examine the regime shift in the kinetic realization (6.18).

6.3.3.2   Computation of Equilibria Using a Triangular Decomposition

Note that the method described in Section 6.3.3.1 requires the knowledge about the system's equilibria, x^* (or the zeros of the system's vector fields). Clearly, one way to compute x^* is by using numerical simulations to integrate the system's ODE model. However, numerical methods are prone to numerical errors and often unstable [25, 31]. Moreover, numerical simulations only provide local information about the system's equilibria around the chosen nominal parameters and initial condition. In contrast, the symbolic computation method based on Gröbner basis gives global information about the system's equilibria. Specifically, it provides the correct information about the structure (i.e. the number and the (real and complex) values) of the equilibria [31]. With regard to the robust stability analysis described in Section 6.3.3.1, these information can be used to preprocess the numerical simulations of the formulated

LMI problem.

Consider a polynomial dynamical system whose state trajectories are modeled by differential equation $\dot{x}(t) = f(x)$ where $f(x) = \{f_1(x), \ldots, f_n(x)\} \in \mathbb{K}[x]$ and $x = [x_1, \ldots, x_n]^T$. Suppose we are interested in computing the system's equilibria, x^* defined by the zeros of its vector fields, i.e. $x^* = \{x : f_1(x) = \cdots = f_n(x) = 0\}$. This is equivalent with the problem of characterizing the varieties $\mathbb{V}(I)$ of the ideal $I = \langle f_1, \ldots, f_n \rangle$ generated by the system's vector fields. If I is a zero dimensional ideal, then the Gröbner basis method can be used to approximate the numerical values of x^* through what is known as a symbolic-numerical approach [31]. In this approach, the symbolic part aims at computing a Gröbner basis $\mathbb{G}(I)$ of the ideal I. In particular, the computed Gröbner basis is then decomposed into a set of bases with special forms. One such form is known as a *triangular basis* defined below.

Definition 4 ([31]). *Let $I \subset \mathbb{K}[x]$ be a zero dimensional ideal. A set of polynomials $\mathcal{T} = \{T_1, \ldots, T_n\} \subset \mathbb{K}[x]$ is called a triangular basis if, for each $j = 1, \ldots, n$,*

(i) $T_j \in \mathbb{K}[x_{n-j+1,\ldots,x_n}] \backslash \mathbb{K}$, and

(ii) the leading monomial of T_j with respect to the lexicographic order is of the form $x_{n-j+1}^{m_j}$ for some $m_j \geq 1$.

A list of triangular bases $\mathcal{T}_1, \ldots, \mathcal{T}_t$ is called a triangular decomposition of I if

$$\mathbb{V}(I) = \mathbb{V}(\mathcal{T}_1) \cup \cdots \cup \mathbb{V}(\mathcal{T}_t).$$

Definition 4 essentially means that each triangular basis, $\mathcal{T}_\ell$ has at least one univariate polynomial, say T_1, in its elements. As such, the zeros/varieties of such a T_1 can be computed using numerical root finding technique. The computed zeros can then be back substituted to another polynomial, say T_2, in $\mathcal{T}_\ell$ to get another univariate polynomial. The back substitution process is then repeated to all polynomials T_j in each $\mathcal{T}_\ell$. As a result, the varieties $\mathbb{V}(\mathcal{T}_\ell)$ of each $\mathcal{T}_\ell$ are defined as $\mathbb{V}(\mathcal{T}_\ell) = \{x :$

$T_1 = \cdots = T_j = 0\}$. By combining the varieties of all triangular bases $\mathcal{T}_1, \ldots, \mathcal{T}_t$, the varieties of an ideal I are then defined by the union set $\mathbb{V}(I) = \mathbb{V}(\mathcal{T}_1) \cup \cdots \cup \mathbb{V}(\mathcal{T}_t)$ of each triangular basis' varieties.

The main advantage in using the symbolic-numerical approach is that it characterizes all the zeros of a system of polynomial equations [31]. In other words, it provides global information about the structure (i.e. the number and the values) of the varieties of an ideal. These information are provided in the forms of the triangular bases $\mathcal{T}_1, \ldots, \mathcal{T}_t$. As a result, if one is interested in studying the behavior of a particular variety, one may choose to focus only on the triangular basis $\mathcal{T}_\ell$ associated with that variety.

Algorithm 5 summarizes the symbolic-numerical method for computing the varieties of a zero dimensional ideal based on a triangular decomposition [31]. This algorithm has been implemented in SINGULAR computer algebra [32]. The Buchberger's algorithm [14] can be used in step 1 to compute a Gröbner basis $\mathbb{G}(I)$ of a zero dimensional ideal I. Furthermore, the SINGULAR library `triang.LIB` contains a function `triangMH` that can be used in step 2 to compute the triangular bases $\mathcal{T}_\ell$. In particular, this library implemented triangular decomposition methods proposed in [97, 109]. Finally, numerical root finding method such as the function `roots` in MATLAB can be used in step 3. Section 6.3.4 illustrates the use of Algorithm 5 in robust stability analysis of the kinetic realization (6.18).

Remark 6.3.4. Note that the triangular decomposition method described in this section can only be applied for zero dimensional ideals (i.e. ideals whose variety contains only finitely many points). The extension of this method to positive dimensional ideals is currently an active research topic [96, 105] and so its integration to the method proposed in this thesis is left for future work.

Algorithm 5 Symbolic-numerical method for solving zero dimensional ideals [31]

Input: a list of polynomials $f_1, \ldots, f_n \in \mathbb{K}[x]$ generating a zero dimensional ideal in $\mathbb{K}[x]$

Output: the set of all complex solutions of $f_1, \ldots, f_n = 0$

1: Compute a reduced lexicographic Gröbner basis $\mathbb{G}(I)$ for $I = \langle f_1, \ldots, f_n \rangle$.
2: Starting from the Gröbner basis $\mathbb{G}(I)$, compute a triangular decomposition $\mathcal{T}_1, \ldots, \mathcal{T}_t$ for I using function `triangMH` in SINGULAR library `triang.LIB`
3: Step 3. For each i, successively use a numerical root finding solver to find the coordinate entries of the zeros of $\mathcal{T}_\ell$

6.3.4 Robust Asymptotic Stability Analysis of the Single Clone Model

This section describes an application of the robust stability analysis method from Section 6.3.3 for characterizing regime shifts in the kinetic realization (6.19)-(6.23). In this analysis, we are interested in characterizing the value of the dilution rate, δ where the system's robust asymptotic stability could no longer be assured. We show that the prediction from our analysis match the results from numerical simulations.

For the kinetic realization in (6.19)-(6.23), we first computed the extreme generators or the EFMs using EFMtool [149, 87]. We found 36 flux modes as shown in equation (D.1) in Appendix D.3. The convex cone $\mathcal{K}_v$ in (6.24) is given in equation (D.2) in Appendix D.3. The Jacobian matrix (6.26) of the realization was computed and is given by

$$
J(\lambda, h) = \begin{bmatrix} J_{1,1} & J_{1,2} & 0 & 0 \\ h_1\left(\sum_{i=6}^{8}\lambda_i + \sum_{j=10}^{12}\lambda_j\right) & J_{2,2} & 0 & J_{2,4} \\ -h_1\left(\lambda_1 - \lambda_4\right) & h_2\left(\sum_{i=1}^{4}\lambda_i\right) & 0 & 0 \\ J_{4,1} & J_{4,2} & h_3\left(\sum_{i=13}^{20}\lambda_i\right) & -h_4\left(\sum_{i=13}^{20}\lambda_i\right) \end{bmatrix},
$$

$$(6.35)$$

158

where $h_i = 1/x_i^*$, $i = 1, \ldots, 4$ and

$$J_{1,1} = -h_1 \left(\sum_{i=21}^{24} \lambda_i + \sum_{j=31}^{36} \lambda_j + \sum_{\ell=23,24,26,29,32,35} \lambda_\ell \right),$$

$$J_{1,2} = -h_2(\lambda_{25} - \lambda_{23} - \lambda_{21} - \lambda_{27} + 2\lambda_{28} + \lambda_{29} + \lambda_{31} - \lambda_{33} + 2\lambda_{34} + \lambda_{35}),$$

$$J_{2,2} = h_2 \left(\lambda_9 - \lambda_7 + \lambda_{10} + \lambda_{12} \right),$$

$$J_{2,4} = -h_4 \left(\lambda_5 + \lambda_6 + \lambda_9 + \lambda_{10} \right),$$

$$J_{4,1} = -h_1 \left(\lambda_{14} - \lambda_{16} - \lambda_{17} + \lambda_{18} \right),$$

$$J_{4,2} = h_2 \left(\lambda_{15} + \lambda_{16} + \lambda_{18} + \lambda_{19} \right).$$

A MATLAB script was written to automate the computation of the Jacobian's parameter dependent form $A(\mu)$ in (6.31). For the Jacobian matrix (6.35), the following 66 new parameters were identified

$$\mu_1 = h_1\lambda_1, \quad \mu_2 = h_1\lambda_{10}, \quad \mu_3 = h_1\lambda_{11}, \quad \mu_4 = h_1\lambda_{12}, \quad \mu_5 = h_1\lambda_{14}, \quad \mu_6 = h_1\lambda_{16},$$

$$\mu_7 = h_1\lambda_{17}, \quad \mu_8 = h_1\lambda_{18}, \quad \mu_9 = h_1\lambda_{21}, \quad \mu_{10} = h_1\lambda_{22}, \quad \mu_{11} = h_1\lambda_{23}, \quad \mu_{12} = h_1\lambda_{24},$$

$$\mu_{13} = h_1\lambda_{26}, \quad \mu_{14} = h_1\lambda_{29}, \quad \mu_{15} = h_1\lambda_{31}, \quad \mu_{16} = h_1\lambda_{32}, \quad \mu_{17} = h_1\lambda_{33}, \quad \mu_{18} = h_1\lambda_{34},$$

$$\mu_{19} = h_1\lambda_{35}, \quad \mu_{20} = h_1\lambda_{36}, \quad \mu_{21} = h_1\lambda_4, \quad \mu_{22} = h_1\lambda_6, \quad \mu_{23} = h_1\lambda_7, \quad \mu_{24} = h_1\lambda_8,$$

$$\mu_{25} = h_2\lambda_1, \quad \mu_{26} = h_2\lambda_{10}, \quad \mu_{27} = h_2\lambda_{12}, \quad \mu_{28} = h_2\lambda_{15}, \quad \mu_{29} = h_2\lambda_{16}, \quad \mu_{30} = h_2\lambda_{18},$$

$$\mu_{31} = h_2\lambda_{19}, \quad \mu_{32} = h_2\lambda_2, \quad \mu_{33} = h_2\lambda_{21}, \quad \mu_{34} = h_2\lambda_{23}, \quad \mu_{35} = h_2\lambda_{25}, \quad \mu_{36} = h_2\lambda_{27},$$

$$\mu_{37} = h_2\lambda_{28}, \quad \mu_{38} = h_2\lambda_{29}, \quad \mu_{39} = h_2\lambda_3, \quad \mu_{40} = h_2\lambda_{31}, \quad \mu_{41} = h_2\lambda_{33}, \quad \mu_{42} = h_2\lambda_{34},$$

$$\mu_{43} = h_2\lambda_{35}, \quad \mu_{44} = h_2\lambda_4, \quad \mu_{45} = h_2\lambda_7, \quad \mu_{46} = h_2\lambda_9, \quad \mu_{47} = h_3\lambda_{13}, \quad \mu_{48} = h_3\lambda_{14},$$

$$\mu_{49} = h_3\lambda_{15}, \quad \mu_{50} = h_3\lambda_{16}, \quad \mu_{51} = h_3\lambda_{17}, \quad \mu_{52} = h_3\lambda_{18}, \quad \mu_{53} = h_3\lambda_{19}, \quad \mu_{54} = h_3\lambda_{20},$$

$$\mu_{55} = h_4\lambda_{10}, \quad \mu_{56} = h_4\lambda_{13}, \quad \mu_{57} = h_4\lambda_{14}, \quad \mu_{58} = h_4\lambda_{15}, \quad \mu_{59} = h_4\lambda_{16}, \quad \mu_{60} = h_4\lambda_{17},$$

$$\mu_{61} = h_4\lambda_{18}, \quad \mu_{62} = h_4\lambda_{19}, \quad \mu_{63} = h_4\lambda_{20}, \quad \mu_{64} = h_4\lambda_5, \quad \mu_{65} = h_4\lambda_6, \quad \mu_{66} = h_4\lambda_9.$$

$$(6.36)$$

The system's parameter dependent matrix $A(\mu)$ is then given by

$$A(\mu) = \begin{bmatrix} A_{1,1} & A_{1,2} & 0 & 0 \\ \sum_{i=2}^{4} \mu_i + \sum_{j=22}^{24} \mu_j & \mu_{26} + \mu_{27} - \mu_{45} + \mu_{46} & 0 & A_{2,4} \\ \mu_{21} - \mu_1 & \mu_{25} + \mu_{32} + \mu_{39} + \mu_{44} & 0 & 0 \\ \mu_6 - \mu_5 + \mu_7 - \mu_8 & \mu_{28} + \mu_{29} + \mu_{30} + \mu_{31} & \sum_{i=47}^{54} \mu_i & -\sum_{i=56}^{63} \mu_i \end{bmatrix},$$

$$(6.37)$$

where

$$A_{1,1} = -\mu_9 - \mu_{10} - 2\mu_{11} - 2\mu_{12} - \mu_{13} - \mu_{14} - \mu_{15} - 2\mu_{16} - \mu_{17} - \mu_{18} - 2\mu_{19} - \mu_{20},$$

$$A_{1,2} = \mu_{33} + \mu_{34} - \mu_{35} + \mu_{36} - 2\mu_{37} - \mu_{38} - \mu_{40} + \mu_{41} - 2\mu_{42} - \mu_{43},$$

$$A_{2,4} = -\mu_{55} - \mu_{64} - \mu_{65} - \mu_{66}.$$

Given the matrix $A(\mu)$ in (6.37), 67 matrices $A_0, A_1, \ldots, A_{66}$ were computed and used as inputs to the LMI toolbox function `pdlstab`. In addition to system matrices $A_0, A_1, \ldots, A_{66}$, the function pdlstab also requires a set of vertices characterizing the polytopic set Γ of the uncertain parameter μ. In our case, we used a simplex constructed around a nominal flux parameter μ_0. The vector μ_0 was computed from a nominal system parameter vector, k_0 as explained below.

6.3.4.1 Nominal State Equilibria

For model (6.2), we chose the following values for the nominal parameters.

$$\delta = 1.3, \qquad N_{in} = 100, \qquad b_C = 7.151, \qquad K_C = 11.164, \qquad b_B = 2.729,$$

$$K_B = 9.995, \qquad \epsilon = 0.066, \qquad m = 0.003, \qquad \alpha = 0.806. \qquad (6.38)$$

These nominal parameters, except the dilution rate δ, were values obtained from model identification (cf. Table 6.1). The dilution rate $\delta = 1.3$ was chosen for the
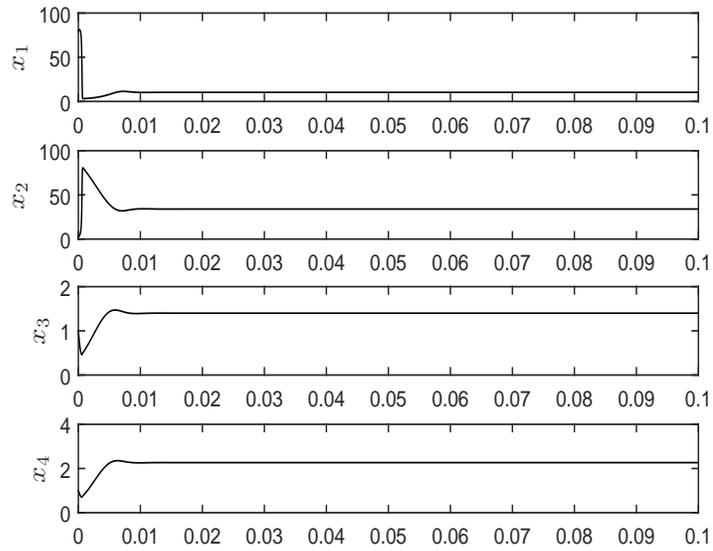
Figure 6.5. State trajectories of (6.18) for k_0 in (6.39).

nominal parameter because it ensures the nominal system has a stable coexistence equilibrium. With these parameters, the nominal set of parameters $k_0 = [k_1, \ldots, k_{14}]$ in the scaled model (6.18) is

$$k_0 = [1.3,\ 130,\ 7.151,\ 11.164,\ 2.729,\ 9.995,\ 41.348,$$

$$1.303,\ 2.109,\ 44.012,\ 5.851,\ 58.48,\ 6.921,\ 0.62]. \quad (6.39)$$

Figure 6.39 plots the state trajectories of system (6.18) for the given k_0.

Algorithm 5 was used to compute the nominal state equilibria for the chosen k_0.

First, note that the right hand side of the ODE (6.16) can be rewritten as

$$\dot{x}_1 = (K_B + x_2)g_1(x) = (K_B + x_2)\left[\delta(N_{in} - x_1)(K_C + x_1) - b_C x_1 x_2\right],$$

$$\dot{x}_2 = g_2(x) = b_C(K_B + x_2)x_1 x_2 - \frac{b_B}{\epsilon}(K_C + x_1)x_2 x_4 - \delta(K_C + x_1)(K_B + x_2)x_2,$$

$$\dot{x}_3 = (K_C + x_1)g_3(x) = (K_C + x_1)\left[b_B x_2 x_3 - (\delta + m + \alpha)(K_B + x_2)x_3\right],$$

$$\dot{x}_4 = (K_C + x_1)g_4(x) = (K_C + x_1)\left[b_B x_2 x_3 - (\delta + m)(K_B + x_2)x_4\right].$$
$$(6.40)$$

The system equilibria x^* are given by

$$x^* := \{x : (K_B + x_2)g_1(x) = 0, \ g_2(x) = 0, \ (K_C + x_1)g_3(x) = 0, \ (K_C + x_1)g_4(x) = 0\}.$$
$$(6.41)$$

Since we are only interested in nonnegative equilibria, it is sufficient to consider

$$x^* = \{x : \ g_i(x) = 0, \quad \text{for } i = 1, \ldots, 4\}. \tag{6.42}$$

This is because both the factors $(K_B + x_2)$ and $(K_C + x_1)$ in (6.41) define negative equilibria for x_2 and x_1, respectively. Thus, the input for Algorithm 5 was polynomials $g_i, \ i = 1, \ldots, 4$, that generate an ideal $I = \langle g_1, \ldots, g_4 \rangle$. The output of Algorithm 5 was the following set of three triangular bases

$$T_1 = \{x_4, \ x_3, \ 200x_2^2 + 1999x_2, \ (0.357x_1 x_2 - 5.774x_1 + 0.065x_1^2 - 72.566) \times 10^5\},$$

$$T_2 = \{(0.27x_4^2 - 2.53x_4 + 4.34) \times 10^{17}, \ (2.109x_3 - 1.303x_4) \times 10^3,$$

$$(0.124x_2 - 4.215) \times 10^6, \ (0.177x_1 + 4.358x_4 - 11.703) \times 10^8\},$$

$$T_3 = \{x_4, \ x_3, \ (0.029255x_2 - 2.852934) \times 10^6, \ (2.925x_1 - 7.256)) \times 10^8\}.$$
$$(6.43)$$

Thus, the system equilibria are defined by the varieties of these bases, i.e. $x^* := \{x : \ T_i = 0, \ i = 1, \ldots, 3\}$. We examined the varieties of each triangular basis by numerically solving for the zeros of their elements.

- The varieties $\mathbb{V}^1 = \mathbb{V}(T_1)$ of T_1 were given by

$$\begin{aligned}
\mathbb{V}_1^1 &= \{x_1 = 99.9953, \, x_2 = 0, \, x_3 = 0, \, x_4 = 0\}, \\
\mathbb{V}_2^1 &= \{x_1 = -11.1645, \, x_2 = 0, \, x_3 = 0, \, x_4 = 0\}, \\
\mathbb{V}_3^1 &= \{x_1 = 54.4417, \, x_2 = 9.995, \, x_3 = 0, \, x_4 = 0\}, \\
\mathbb{V}_4^1 &= \{x_1 = -20.5063, \, x_2 = 9.995, \, x_3 = 0, \, x_4 = 0\}.
\end{aligned} \tag{6.44}$$

Note that each of these varieties has either negative value of x_1 (corresponds to negative nitrogen concentration) or zero values of x_3 and x_4 (correspond to rotifers' extinction). Since we are only interested in those equilibria in which all species coexist in positive concentrations, the equilibria defined by $\mathbb{V}(T_1)$ were neglected in further analysis.

- The varieties $\mathbb{V}^2 = \mathbb{V}(T_2)$ of T_2 were given by

$$\begin{aligned}
\mathbb{V}_1^2 &= \{x_1 = -108.4769, \, x_2 = 33.9991, \, x_3 = 4.3855, \, x_4 = 7.0982\}, \\
\mathbb{V}_2^2 &= \{x_1 = 10.2915, \, x_2 = 33.9991, \, x_3 = 1.4002, \, x_4 = 2.2664\},
\end{aligned} \tag{6.45}$$

The first variety has a negative value of x_1 and so it was also neglected. The second variety contains all states with positive values. This means $\mathbb{V}_2^2$ defines an equilibrium where all species coexist and will be included for further analysis.

- The variety $\mathbb{V}^3 = \mathbb{V}(T_3)$ of T_3 was given by

$$\mathbb{V}_1^3 = \{x_1 = 2.4805, \, x_2 = 97.5195, \, x_3 = 0, \, x_4 = 0\}. \tag{6.46}$$

This variety was also neglected as it has x_3 and x_4 with zero values (correspond to the extinction of the rotifers population).

The above inspection suggests that we should consider the second element of T_2 in our analysis. In this case, the nominal equilibrim point was then given by

$$x^* = [x_1^*, \, x_2^*, \, x_3^*, \, x_4^*]^T = [10.2915, \, 33.9991, \, 1.4002, \, 2.2664]^T. \tag{6.47}$$

We compared the equilibrium computed above with that obtained from direct integration of ODE (6.18). We found that the results were the same (cf. Figure 6.5).

6.3.4.2 Nominal Flux Mode

The substitution of x^* in (6.47) and k_0 in (6.39) into the flux vector $v(x,k)$ in (6.21) gives a vector of nominal flux equilibrium

$$v(x^*, k_0) = 10^4 \times [0.0304, 0.0330, 2.0463, 6.9606, 1.5400, 0.1033, 3.2791, 0.0304, 0.0304$$

$$0.1337, 0.1121, 3.5570, 0.0330, 0.0330, 0.1450, 8.5072, 1.3372,$$

$$4.9344, 1.4506, 0.4681, 0.13761.6776, 0.4932, 0.1493]^T. \quad (6.48)$$

The nominal vector of convex parameters $\lambda^0 = [\lambda_1^0, \ldots, \lambda_{36}^0]$ was then obtained from the solution λ^0 of linear equations

$$E\lambda^0 = v(x^*, k_0), \quad (6.49)$$

where E is the matrix of flux modes in (D.1) and $v(x^*, k_0)$ is the nominal flux equilibrium in (6.48). The computed nominal convex parameters was

$$\lambda^0 = 10^4 \times [0, 0.0330, 0.0304, 0, 2.0463, 0, 0, 0, 1.2328, 3.5570, 1.6776, 0.4932, 0.1033$$

$$0, 0.0304, 0, 0, 0, 0.0330, 0.1121, 0.1376, 0, 0, 0, 1.4024, 0, 0,$$

$$0.7198, 0.4681, 0.1493, 4.9344, 0, 0, 1.4506, 0, 0]^T. \quad (6.50)$$

Using λ^0 in (6.50) and the reciprocals of the state equilibrium in (6.47), we computed the vector of nominal flux parameters $\mu_0 = [\mu_{1,0}, \ldots, \mu_{66,0}]^T$ using equation (6.36). Using μ_0, the nominal parameterized matrix $A(\mu_0)$ in (6.37) was then constructed

and is given by

$$A(\mu_0) = \begin{bmatrix} -0.6792 & -0.3237 & 0 & 0 \\ 0.5086 & 0.1409 & 0 & -3.0163 \\ 0 & 0.0019 & 0 & 0 \\ 0 & 0.0019 & 0.1991 & -0.1230 \end{bmatrix} \times 10^4. \tag{6.51}$$

The eigenvalues $\nu(A(\mu_0))$ of $A(\mu_0)$ were computed to be

$$\nu(A(\mu_0)) = 10^3 \times [-3.751, -0.512 - 0.896i, -0.512 + 0.896i, -1.982]. \tag{6.52}$$

All eigenvalues in (6.52) have negative real parts. This indicates that the nominal flux mode μ_0 guarantees the asymptotic stability of the nominal system.

6.3.4.3 Robust Stability to Variation in the Dilution Rate

Given the nominal matrix $A(\mu_0)$ in (6.51), we are interested in characterizing the range of dilution rate, δ where the system's states are guaranteed to be in a stable coexistence equilibrium. For this purpose, we identified a subset of the EFMs that depends on the parameter δ (or parameter k_1 in the scaled model (6.18)). In our analysis, this was done by first identified those λs in the convex cone $\mathcal{K}_v$ (D.2) that correspond to the last two rows of the flux vector $v(x, k)$ in equation (6.21). The EFMs that depend on k_1 were then given by those μ's in (6.36) that correspond to these λs. We identified the following seven EFMs: $\mu_4 = h_1\lambda_{12}$, $\mu_{17} = h_1\lambda_{33}$, $\mu_{20} = h_1\lambda_{36}$, $\mu_{24} = h_1\lambda_8$, $\mu_{27} = h_2\lambda_{12}$, $\mu_{36} = h_2\lambda_{27}$, $\mu_{41} = h_2\lambda_{33}$. Thus, the set of varying parameters in the MATLAB function `pdlstab` was given by $\tilde{\mu} = \{\mu_4, \mu_{17}, \mu_{20}, \mu_{24}, \mu_{27}, \mu_{36}, \mu_{41}\}$. All other EFMs were set to their nominal values in μ_0 and stored in matrix A_0 (cf. equation (6.30)).

In our analysis, a simplex characterizing polytopic set Γ of the uncertain param-

eters in $\tilde{\mu}$ was constructed around the nominal flux parameter μ_0. One vertex was obtained by scaling μ_0 back to the origin, $v_1 = \mu_0/\alpha$ where $\alpha > 1$. For the chosen varying EFMs, this origin is defined by $\mu_0 = \{479.2112, 0, 0, 0, 145.0706, 0, 0\}$. The other vertices were obtained by multiplying one component of μ_0 by α. A bisection search was used to find the largest α for which the function `pdlstab` assures the robust stability of the nominal parameter μ_0. From our analysis, we found that `pdlstab` guarantees the robust stability of μ_0 for $\alpha = 4.325$. The set of critical varying EFMs that corresponds to this α was $\tilde{\mu}^* = \{2072.6, 0, 0, 0, 627.4302, 0, 0\}$.

We then perturbed the parameter δ to find the critical value δ^* that corresponds to $\tilde{\mu}^*$. We found $\delta^* = 1.54965$ which corresponds to $\tilde{\mu}^* = \{2072.6, 0, 0, 0, 172.92, 0, 0\}$. In terms of the chosen varying EFMs, this indicates that the robust stability is more sensitive to variation in μ_4. From this result, we then concluded that the robust stability can be assured for the following range of dilution rate

$$1.3 = \delta_{apd}^L \leq \delta \leq \delta_{apd}^U = 1.5496. \tag{6.53}$$

We checked the result of our analysis with that from numerical simulations through direct integration of the ODE model in (6.16). In our simulations, the parameter values in (6.38) for which the system has a stable coexistence equilibrium were chosen as the nominal parameters. We varied parameter δ and used simulations to determine the upper δ_{sim}^U and lower δ_{sim}^L values where the system was no longer has a stable coexistence equilibrium. We found that stable coexistence equilibria were assured within the parameter range

$$1.07 = \delta_{sim}^L \leq \delta \leq \delta_{sim}^U = 1.66. \tag{6.54}$$

Figure 6.6 plots the state trajectories of system (6.18) for dilution rates outside this range. For $\delta = 1.06 < \delta_{sim}^L$ the system exhibits limit cycle whereas for $\delta = 1.67 > \delta_{sim}^U$

the system is in a steady state where rotifers are extinct. It can be seen that the results from the APD robust stability analysis lie within the range obtained from numerical simulations through direct integration of the system's ODE model. This suggests that the proposed method indeed provides a way of characterizing how robust a system may be to parametric variations and therefore can be used to estimate lower bounds on the distance to regime shifts.

The analyis results described in this chapter can be tested in a manner that is similar to the presented simulation. First, a nominal set up of the chemostat with a dilution rate $\delta = 1.3$ should be established. From the analysis, this nominal dilution rate is expected to result in a stable coexistence of both the *C. vulgaris* and *B. calyciflorus* population in the chemostat. Once the nominal condition is achieved, the dilution rate can be changed to $\delta^* = 1.5496$. Based on the APD robust stability analysis, we expect that the nominal stable equilibrium where both species coexist will still be maintained for this δ^*. As the the dilution rate is further increased up to $\delta > 1.67$, we expect that the *C. vulgaris* will survive in the chemostat while the *B. calyciflorus* will go extinct.

6.4   Final Remark

This chapter has presented an application of regime shifts analysis in a model of real life systems from ecology. We descibed the practical limitations of the D_2B analysis method when studied regime shifts in this model and proposed an alternative approach [98]. The approach essentially formulated the regime shifts analysis as robust stability analysis of affine parameter dependent systems [51]. The approach combined the effectiveness of linear matrix inequality (LMI) methods [12] and the symbolic-numerical algorithms for solving a system of polynomial equations [31]. We found that the analysis results from the newly proposed method provide lower bounds on the results obtained from numerical simulations of the model. This suggests that
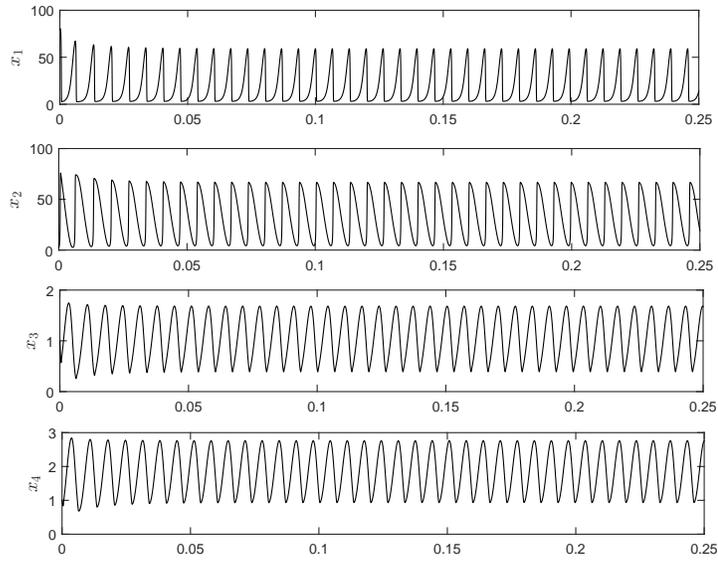
(a) Dilution rate $\delta = 1.06$.
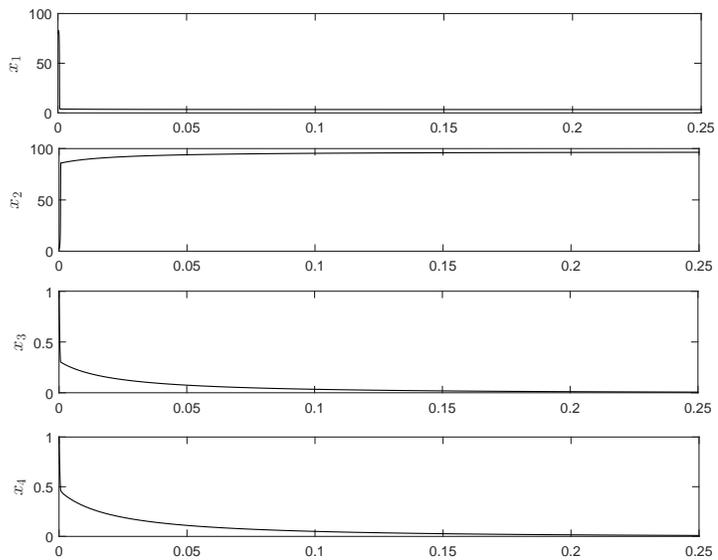
(b) Dilution rate $\delta = 1.67$.

Figure 6.6. State trajectories of (6.18) for different dilution rates.

the used method indeed provides a way of characterizing how robust a system may be to parametric variations and therefore can be used as a measure of the distance to regime shifts.

CHAPTER 7

CONCLUSION

Regime shifts refer to sudden and rapid changes in the structure or function of an ecosystem due to the presence of forces from external disturbances. These changes can occur suddenly, and at times without warning, potentially causing drastic or irreversible shifts in the ecosystems states. While regime shifts are well documented in the ecological literature, the underlying mechanisms governing such shifts are often unknown for certain. This thesis contributes to the study and understanding of the underlying mechanisms leading to regime shifts. Specifically, this thesis developed mathematical and computational methods to forecast the onset of regime shifts in biological and ecological systems.

The basic approach on the developed methods is the formulation of some quantities characterizing the likelihood that a system shifts from its current operating regime in the presence of external forces that perturb either its states or parameters. We identified two mechanisms by which regime shifts may occur and formulated some real-valued quantities that can be used as indicators of how close the system is to each type of regime shift. The first regime shifts mechanism was referred to as *bifurcation-induced regime shifts* and it occurs because variations in the system's parameters force the system's equilibria to undergo bifurcations. We used a quantity called the minimum *distance to bifurcation* as a measure of how close a system is to this type of regime shifts (cf. Chapter 4). We presented an SOS optimization method that simplifies the computation of this quantity's lower bound in the class of *nonnegative systems with kinetic realizations*. The second regime shifts mechanism

169

was referred to as *noise-induced regime shifts* and it occurs because the underlying system has multiple stable equilibria and external stochastic disturbances drive the system's state from the region of attraction (ROA) of one stable equilibrium to the ROA of an alternative stable equilibrium. We used probabilistic quantities called *mean first passage times* and *safety probability* to characterize the *expected time* and the *likelihood* for this type of regime shifts to occur (cf. Chapter 5). We presented SOS optimization methods that can be used to compute upper bounds for these quantities in systems that are modeled as jump diffusion processes. In both mechanisms, we demonstrated the effectiveness of the proposed methods to predict the occurrence of regime shifts in several theorethical models of dynamical systems.

The results from experiments on an ecological test bed that was used to evaluate the methods developed in this thesis were also reported. The test bed was a laboratory scale chemostat that cultures a microbial predator and prey system between green algae, *C. vulgaris*, and rotifer, *B. calyciflorus*. The experiment was conducted to identify a mathematical model for the *C. vulgaris* and *B. calyciflorus* interaction in the chemostat. The identified model was then analyzed to predict regime shifts that may occur in the system under parametric variations (e.g. change on the dilution rate of the chemostat). We showed how the methods developed in this thesis, combined with symbolic-numerical algorithm for solving polynomial equations, can be used to predict regime shifts in the identified model. In particular, we showed that the analysis results from the proposed method provide lower bounds on the distance to regime shifts. This suggests that the developed method can be used to measure and characterize the likelihood of a regime shift in models of real life system.

The results from our experimental evaluation suggest that the robust stability analysis based on the affine parameter dependent formulation is a more feasible approach to characterize distance to regime shifts in the models of real life systems. In this regard, we identified several directions in which the presently established ap-

proach can be extended. The first direction is the extension of the triangular decomposition method used in this thesis so that it can be applied to positive dimensional ideals. As mentioned in Chapter 6, such an extension is currently an active research topic among researchers in Algebraic Geometry [96, 105]. If such an extension can be achieved, then the currently developed approach can be used to analyze polynomial systems whose vector fields generate positive dimensional ideals. The second direction is to use different shapes of polytope (other than a simplex) when searching for the parameter region where the robust stability of the system is assured. We expect that the use of different polytope shape can helps refine the search of robustly stable parameter region. The last direction is to explore the graph properties of kinetic realizations and then examine their possible use to improve the robust stability analysis approch proposed in this thesis.

APPENDIX A

HANDELMAN RELAXATION OF POP PROBLEM

A.1   Handelman Polynomial

Consider again the set $\mathcal{S}$ in (2.9) and let us define the following sets

$$\mathcal{S}_F := \{x \in \mathbb{R}^n : F_i(x, k) = 0, \ k \in \mathbb{R}\}, \tag{A.1a}$$

$$\mathcal{S}_G := \{x \in \mathbb{R}^n : G_j(x, k) \geq 0, \ k \in \mathbb{R}\}, \tag{A.1b}$$

that is $\mathcal{S}_F$ and $\mathcal{S}_G$ denote the set of equalities and inequalities, respectively, which define the set $\mathcal{S}$. Assume further that each polynomial $G_j(x, k)$ in $\mathcal{S}_G$ are affine function in x. Then for a compact polytope $\mathcal{S}_G$ in (A.1b) and an N_G dimensional multi-index β, the βth *Handelman monomial* $H_\beta^G(x)$ that corresponds to $\mathcal{S}_G$ is defined as [66]

$$H_\beta^G(x) = \prod_{j=1}^{N_G} (G_j)^{\beta_j}. \tag{A.2}$$

One important property of the Handelman monomials is given by the following Handelmann representation theorem.

Theorem A.1.1 (Handelman representation [66])**.** *Let $\mathcal{S}_G \subset \mathbb{R}^n$ be a compact polytope as defined in (A.1b) where each $G_j : \mathbb{R}^n \to \mathbb{R}$ is an affine function for $j = 1, \ldots, N_G$. A polynomial $g : \mathbb{R}^n \to \mathbb{R}$ is nonnegative on $\mathcal{S}_G$ if and only if there exist nonnegative coefficients c_β such that $g(x)$ can be represented as*

$$g(x) = \sum_\beta c_\beta H_\beta^G(x). \tag{A.3}$$

For an integer $d \geq 1$, define the following set of Handelman polynomials

$$H_d(\mathcal{S}_G) = \left\{ \sum_{|\beta| \leq d} c_\beta H_\beta^G, \quad c_\beta \in \mathbb{R}_+ \right\}. \tag{A.4}$$

Given a function $g : \mathbb{R}^n \to \mathbb{R}$ with $g(x) \in \mathbb{R}[x]$, a sequence of lower bounds γ_d^* for the minimum $\gamma^* := \min\{g(x), \forall x \in \mathcal{S}_G\}$ of $g(x)$ on $\mathcal{S}_G$ can be computed by solving the following optimization problem.

$$\gamma_d^* = \max \ \gamma, \quad \text{such that} \ \ g(x) - \gamma \in H_d(\mathcal{S}_G).$$

Note that $\gamma_d^* \leq \gamma^*$ and γ_d^* converges asymptotically to γ^* as $d \to \infty$ [94, 30].

Theorem A.1.2 below combines the Handelman representation described above and the positivstellensatz [154, 137] to characterize the existence of solution to a system of equations of the form (A.1a) over a compact poytope $\mathcal{S}_G$ defined in (A.1b).

Theorem A.1.2 ([154]). *Consider the system of equations in (A.1a) and let $\mathcal{S}_G$ be a compact polytope as defined in Theorem A.1.1. Then the following two statements are equivalent.*

(i) *The system of equations in (A.1a) does not have a solution in $\mathcal{S}_G$.*

(ii) *There exist polynomials $P_i(x) \in \mathbb{R}[x]$, $i = 1, \ldots, N_F$ and nonnegative coefficients c_β such that the polynomial*

$$Z(x) = \sum_{i=1}^{N_F} P_i F_i - 1 \tag{A.5}$$

can be represented as

$$Z(x) = \sum_{|\beta|} c_\beta H_\beta^G(x). \tag{A.6}$$

For given polynomials F_i in (A.1a), one may see that the $Z(x)$'s representation in (A.5) is linear in the unknown coefficients of polynomials $P_i(x)$. Similarly, for given polynomials G_j in (A.1b) then $Z(x)$'s representation in (A.6) is also linear in the unknown Handelman polynomials' coefficients c_β. By equating the coefficients of the

monomials of similar degree in both (A.5)'s and (A.6)'s representations, the search for certifying polynomial $Z(x)$ may then be posed as a feasibility problem of a linear programming (LP) formulation (cf. [154, procedure/algorithm in pp. 1143]). By Theorem A.1.2, the existence of such a polynomial $Z(x)$ implies that the solution to a system of polynomial equations in (A.1a) does not exist.

A.2   Handelman Relaxation of POP

In order to use the above Handelman representation to compute the bound γ in RPOP (2.10), let us consider the set $\mathcal{S}_{\bar{G}}$ defined below

$$\mathcal{S}_{\bar{G}} := \{x \in \mathbb{R}^n : G_j(x, k) \geq 0, \ f(x, k) - \gamma > 0\}. \qquad \text{(A.7)}$$

Note that the set $\bar{\mathcal{S}}_G$ in (A.7) is obtained by combining the set $\mathcal{S}_G$ in (A.1b) and the constraint $f(x, k) - \gamma > 0$ in RPOP (2.10). Thus, for any nonnegative constant γ and an $(N_G + 1)$ dimensional multi-index $\bar{\beta}$, the following extension of the Handelman monomials in (A.2) can be defined

$$H_{\bar{\beta}}^{\bar{G}}(x) = \prod_{j=1}^{N_G}(G_j)^{\beta_j}(f(x, k) - \gamma)^{\beta_{N_G+1}}. \qquad \text{(A.8)}$$

As a result, Theorem A.1.2 can be used to compute the bound γ by simply replacing H_{β}^G in (A.6) with $H_{\bar{\beta}}^{\bar{G}}(x)$ in (A.8), i.e. polynomial $Z(x)$ in (A.6) is now given by

$$Z(x) = \sum_{|\beta|} c_{\bar{\beta}} H_{\bar{\beta}}^{\bar{G}}(x). \qquad \text{(A.9)}$$

Thus for any fixed γ, then $Z(x)$ in (A.9) still maintains linearity in the unknown coefficients $c_{\bar{\beta}}$. This suggests that the LP formulation discussed in the previous section can also be constructed by implementing a bisection procedure on γ. The optimal γ^* will then corresponds to the largest value of γ for which $Z(x)$'s representations that

174

satisfy both (A.5) and (A.9) exist.

A.3 Size of the LP formulation

Consider polynomials $F_i(x)$ and $G_j(x)$ in (A.1). Let d_i^F and d_j^G denote the maximum degree of each F_i and G_j, respectively, and define $D_F = \max(d_i^F)$ and $D_G = \max(d_j^G)$. In what follows, we will compute the size of the LP formulation that is required to search for polynomial certificate $Z(x)$ in Theorem A.1.2.

- *Equality constraints:* For polynomial $Z(x)$ in (A.5)

$$Z(x) = \sum_{i=1}^{N_F} P_i F_i - 1,$$

where for each $i = 1, \ldots, N_F$, note each polynomial $P_i \in \mathbb{R}[x]$ has a maximum degree d_i^P with multi-index representation of the form

$$P_i(x) = \sum_{|\alpha| \le d_i^P} c_\alpha x^\alpha. \tag{A.10}$$

It can be shown that the number of coefficients $N(P_i)$ in each polynomial P_i is

$$N(P_i) = \sum_{k=1}^{d_i^P} \frac{(k + (n-1))!}{k!(n-1)!}.$$

Thus, if there are N_F equality constraints of the form (A.1a) and by assuming that each polynomial $P_i(x)$ has a similar maximum degree of $D_P = d_i^P$ for all $i = 1, \ldots, N_F$, then the number of unknown coefficients $N_1(Z)$ of polynomial $Z(x)$ in (A.5) is

$$N_1(Z) = N_F N(P_i) = N_F \left(\sum_{k=1}^{D_P} \frac{(k + (n-1))!}{k!(n-1)!} \right).$$

Note that the degree D_Z of polynomial $Z(x)$ in this case is $D_Z = D_F + D_P$.

- *Inequality constraints:* Let $D_H = 1 + \lceil D_Z/2 \rceil$ be the degree of the Handelman representation in (A.6) where $\lceil \cdot \rceil$ denotes the ceiling function. The Handelman representation of polynomial $Z(x)$ in (A.6) is then given by

$$Z(x) = \sum_{|\beta| \le D_H} c_\beta H_\beta^G(x). \tag{A.11}$$

It can be shown that the number of unknown coefficients $N_2(Z)$ of the Handelman representation of $Z(x)$ defined in (A.6) is

$$N_2(Z) = \sum_{\ell=1}^{D_H} \frac{(\ell + (N_G - 1))!}{\ell!(N_G - 1)!}. \tag{A.12}$$

Thus, the number of decision variables N_H of the LP formulation for searching polynomial certificate $Z(x)$ which verifies Theorem A.1.2 is

$$N_H = N_1(Z) + N_2(Z) = N_F \left(\sum_{k=1}^{D_P} \frac{(k + (n - 1))!}{k!(n - 1)!} \right) + \sum_{\ell=1}^{D_H} \frac{(\ell + (N_G - 1))!}{\ell!(N_G - 1)!}. \tag{A.13}$$

Table A.1 illustrates the size N_H of the LP formulation when there is one equality constraint of degree 2 and 2^n inequality constraints of the form (A.1b) in which n is the number of unknown variables. When compared with the size of SDP's decision variables in Table A.2, one may see from Table A.1 that the size of the LP problem is larger than that of the SDP problem for a chosen degree of polynomial multipliers for the equality constraints.

As a comparison, the size N_{sos} of the SDP formulation (2.15) for computing an SOS relaxation of a polynomial function in n variables with degree d is given by

$$N_{sos} = \binom{n + d}{d}. \tag{A.14}$$

It can be seen that N_{sos} grows polynomially if either n or d is fixed. Table A.2 (cf. [119, Table 1]) illustrates the number of decision variables when there is only one equality constraint in the SDP formulation (2.15).

A.3.1 Comparison between SOS and Handelman Representations

As discussed in the previous sections, the use of SOS and Handelman relaxations for solving the POP (2.9) boils down to formulations of SDP and LP problems,

TABLE A.1

NUMBER OF DECISION VARIABLES IN LP

Number of unknowns (n)

		2	3	4	5	6	7	8	9	10
Degree of $P_i(x)$	1	59	124	225	370	569	824	1149	1550	2035
	2	62	130	235	385	588	852	1185	1595	2090
	3	136	350	750	1421	2464	3996	6150	9075	12936
	4	141	365	785	1491	2590	4206	6480	9570	13651
	5	273	848	2128	4620	9030	16296	27621	44506	68783
	6	280	876	2212	4830	9492	17220	29337	47509	73788

TABLE A.2

NUMBER OF DECISION VARIABLES IN SDP.

Variables (n)

		3	5	7	9
Degree of $r_i(x)$	1	4	6	8	10
	2	10	21	36	55
	3	20	56	120	220
	4	35	126	330	715
	5	56	252	792	2002
	6	84	462	1716	5005

respectively, that correspond to RPOP (2.10). Note that both SDP and LP belong to the class of convex optimization problems. Moreover, an SDP problems can be viewed as a generalization of an LP problem and there are several computational properties and methods in LP formulation that can be extended to SDP formulation. There are, however, some properties of LP problems that do not extend to SDP problems. Some of these are mentioned below [78].

- On one hand, a feasible solution to an LP problem is always guaranteed to achieve its optima. On the other hand, the solutions of SDP problem may or may not achieve their optima and so there may be a finite/ infinite duality gap between the solutions of its primal and dual formulations.

- While there exist finite algorithms (such as simplex) for solving an LP problem, there exists no finite algorithm for solving an SDP problem. In other words, SDP formulations do not have direct analog of the "basic feasibility problem" [11] found in LP formulation.

A major consequence of the above discrepencies can be seen in term of the gap on their solvers' maturity. Specifically, it is widely known that most of the currently available solvers for SDP problems are still under developed and are not as mature as solvers for LP problems [94]. As a result, while many solvers for LP problems have been proven to be capable of dealing with large scale problems [1, 2, 158], solvers for SDP problems have often been found to be capable of dealing with only medium scale problems (cf. [150, 94]).

Based on the above discussion, it is reasonable to expect that the method for solving the RPOP (2.10) based on LP formulation will have a better performance than that of the method based on the SDP formulation one. However, a result from numerical simulations described below, which were conducted to verify the hypothesis, suggests a negative answer. In this simulation, the objective is to compute the minimum of a quadratic polynomial function $p(x)$ in three variables ($n = 3$)

subject to one equality ($N_F = 1$) and six inequalities ($N_G = 6$) constraints.

$$\min \quad p(x) = (x_1 - 1)^2 + (x_2 - 1)^2 + (x_3 - 1)^2$$
$$\text{s.t.} \quad x_1 - (x_1 - x_2)x_3 = 0$$
$$0 \le x_i \le 2, \ i = 1, \dots, 3.$$

Both the SOS and Handelman relaxations were used to solve the above POP. The corresponding SDP and LP formulation were both solved using similar SDP/LP solver Mosek [111].

Figure A.1 illustrates the results obtained after solving both the SDP and LP relaxations of the above optimization problem. On one hand, this figure shows that, while a second order SOS relaxation (solved in 0.45 second) is sufficient to compute a lower bound that satisfies a prespecified accuracy, a tenth order Handelman relaxation (solved in 4.05 second) is required to compute a lower bound that satisfies the same level of accuracy. As suggested in [94], this result illustrates that the SOS relaxation method has a faster convergence to the optimal solution than that of the Handelman relaxation method. On the other hand, Figure A.1b illustrates that, although the size (i.e the number of decision variables) of the corresponding LP problem is larger than that of the corresponding SDP problem (for equal degree of relaxation), the time required to solve LP problem is relatively shorter than that to solve the SDP problem. Such a gap in computation time illustrates the limitation of solvers for SDP problems as compared to solvers for LP problems.

The preliminary finding from numerical simulations discussed above illustrates that, for relatively small size problems ($n \le 5$), the SDP formulation via SOS relaxation outperforms the LP formulation via Handelman relaxation. However, since the currently available solvers for SDP problems are limited to small or medium sizes problems, it is important to realize that similar numerical characteristics as illustrated in Figure A.1 cannot be guaranteed to hold for problems that consist of higher

(a) Comparison of bound errors.   (b) Comparison of computation times.
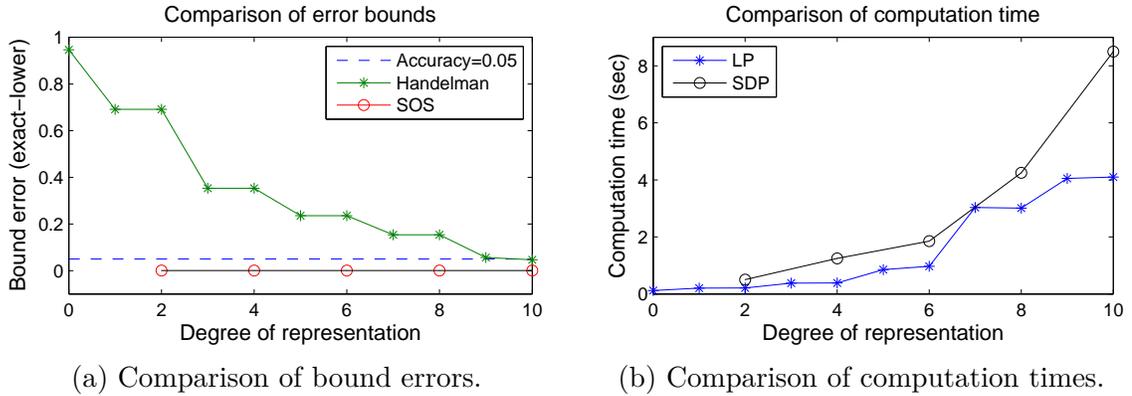
Figure A.1. Comparison between SOS and Handelman relaxations.

order polynomials and larger unknown variables. Equivalently, although solvers for LP problems are known to be capable of handling large scale problems, the combination of "slow convergence" (to the true/optimal solution) of its solutions and the exponential increase on the number of decision variables (relative to linear increase on the degree of relaxation) illustrated in Figure A.1b raises a concern about the conservativeness of the LP formulation. Future research that further investigate this trade off is therefore essential as it possibly allows one to make decision about which type of relaxation is more suitable for a given problem.

APPENDIX B

COMPUTER CODES FOR EXAMPLE 2.4

B.1 SINGULAR Code

Listing B.1: Computing Gröbner basis in (2.20 using SINGULAR[32].

```
> ring R=(0,P,Q,B,G),(V,x,y),lp;
> poly f1 = -V^2*G + V*(G*y+B*x) - P;
> poly f2 = -V^2*B - V*(G*x - B*y) - Q;
> poly f3 = x^2 + y^2 - 1;
> poly f4 = B^2 + G^2 - 2*B^2*V*y - 2*G^2*V*y;
> ideal I = (f1,f2,f3,f4);
> ideal G = groebner(I);
> G;
G[1]=(4P2B2-8PQBG+4PB2G+4PG3+4Q2G2+4QB3+4QBG2-B4-2B2G2-G4)
```

B.2 MATLAB Code

Listing B.2: Computing γ in (2.21) using SOSTOOLS[121].

```
pvar P Q gam  % declare variables
vars = [P;Q]; % variables of the ring
P0 = 0; Q0 = 0; % initial load parameters
B = .5/.26; G = .1/.26; % system parameter
J = (P-P0)^2 + (Q-Q0)^2; % distance to instability
```

```matlab
prog = sosprogram(vars);          % initialize SOSprogram
prog = sosdecvar(prog,gam);   % Declare the decision variable


% Use the Grobner basis from Listing B.1 as an equality constraint
Cons = B^4-4*B^3*Q+2*B^2*G^2-4*B^2*G*P-4*B^2*P^2-4*B*G^2*Q...
    +8*B*G*P*Q+G^4-4*G^3*P-4*G^2*Q^2;


% Define SOS multiplier (constant is sufficient in this case)
[prog,lam] = sospolyvar(prog,1);


% Define the SOS inequality
f = J + lam*Cons;
prog = sosineq(prog,(f-gam));


% Set objective function: maximize decision variable gam
prog = sossetobj(prog,-gam);


% Solve the SOS program
prog = sossolve(prog);


% Get and display the solutions
Lbound = sosgetsol(prog,gam);  display(Lbound);
```

APPENDIX C

TOOLKIT MANUAL

C-BRANDS: A MATLAB toolkit for Certificate-based Bifurcation and Robustness
Analyses of Nonnegative Dynamical Systems

User's Guide

C.1 Introduction

C-BRANDS (Certificate-based Bifurcation and Robustness Analyses for Nonnegative Dynamical Systems) is a MATLAB toolkit for local bifurcation and robustness analyses of nonnegative polynomial dynamical systems. It implements the method described in [146] by utilizing freely available softwares such as SeDuMi [139], SOSTOOLS [121], CellNetAnalyzer [87], and SINGULAR[62]. The current version of C-BRANDS can be used to study the robustness and the distance-to-bifurcation (from a nominal operating point) of nonnegative dynamical systems.

C-BRANDS is a free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version. C-BRANDS is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. You should have received a copy of the GNU General Public License along with this toolkit. If not, please refer to `http://www.gnu.org/licenses/`.

Throughout this user's manual, we use the `typewriter` typeface to denote MATLAB variables and functions, MATLAB commands that you should type, and results given by MATLAB. MATLAB commands that you should type will also be denoted by the symbol `>>` before the commands. For example,

```
>> x = sin(1)


x =

    0.8415
```

In this case, `>> x = sin(1)` is the command that you type, and `x = 0.8415` is the result given by MATLAB.

C.2 Getting Started

C.2.1 System Requirements and Installation

C-BRANDS is developed under Windows platform and takes advantage of MATLAB's symbolic engine MuPAD for symbolic object manipulation. Thus in order to use C-BRANDS, you need to make sure that your MATLAB version includes the Symbolic Math Toolbox. One way to verify this is by typing the following command in MATLAB's Command Window

```
>> ver
```

and then check if Symbolic Math Toolbox is in the list of your MATLAB's toolboxes. The current version of C-BRANDS has been tested under MATLAB R2013a (and higher) and is available for download as a .ZIP file `C-BRANDS.zip` at

`http://nd.edu/˜lemmon/projects/NSF-12-520/`

To install C-BRANDS, please follow the following step-by-step guidelines.

1. Download the .ZIP file `C-BRANDS.zip` from
   `http://nd.edu/~Iemmon/projects/NSF-12-520/`.
   Using any file archiver tool (such as 7-zip, cf. `http://www.7-zip.org/`), extract
   the `C-BRANDS.zip` file to your MATLAB's default directory. In Windows platform,
   this usually creates a folder `C:\Users\UserName\Documents\MATLAB\C-BRANDS`
   (assuming that `C:\Users\UserName\Documents\MATLAB` is the MATLAB's de-
   fault directory). After extracting the content of the .zip file, navigate to the
   C-BRANDS folder and make sure that it contain the following folders/file

   - `SOSTOOLS3.00`   (cf. `http://www.cds.caltech.edu/sostools/`)
   - `SeDuMi_1_3`   (cf. `http://sedumi.ie.lehigh.edu/`)
   - `BRANDS`
   - `Docs`
   - `Singular-3-1-7-Small.exe`   (cf. `http://www.singular.uni-kl.de/`)
   - `SingularTest.sh`

   `SOSTOOLS3.00` and `SeDuMi_1_3` are Matlab's third party toolboxes for sum of
   squares (SOS) and semidefinite programmings, respectively.
   `BRANDS` contain MATLAB files (mfiles) that implement C-BRANDS's functional-
   ity.
   `Docs` this user's guide and report [146].
   `Singular-3-1-7-Small.exe` is an executable installation file of SINGULAR com-
   puter algebra (cf. step 3).
   `SingularTest.sh` is a shell script file which will be used to test the connection
   between MATLAB and SINGULAR computer algebra (see step 5).

2. Download the .ZIP file of CellNetAnalyzer toolbox from `http://www2.mpi-magdeburg.mpg.de/projects/cna/download.html` and extract its content to the C-BRANDS's
   folder (CellNetAnalyzer is not included in C-BRANDS due to distribution copy-
   right).

3. Navigate to the C-BRANDS's folder. Double click the file `Singular-3-1-7-Small.exe`
   to run the installation of SINGULAR computer algebra in your computer. Follow
   the pop-up windows instruction to complete the installation (example views of
   this installation can be seen at `http://www.singular.uni-kl.de/index.php/singular-download/install-windows-single-file.html`). Note that the in-
   stallation of SINGULAR will includes the installation of Cygwin in your computer
   (cf. `www.cygwin.com`).

4. Run MATLAB from Cygwin and add C-BRANDS's folder to the MATLAB's path
   by following the steps below.

- Open Cygwin bash shell (terminal) from Windows desktop by navigating through
  `Start → All Programs → Singular CAS → Cygwin → Bash shell`

- In Cygwin shell, type `matlab`. This will open a MATLAB window with Current Folder at `C:\cygwin\home\UserName`

- In MATLAB's window, set the Current Folder to the MATLAB's folder, i.e. `C:\Users\UserName\Documents\MATLAB`.

- Add C-BRANDS's folder to the MATLAB's path, i.e. right click on the CBRAND's folder and choose
  `Add to Path → Selected Folders and Subfolders`

- Set MATLAB's Current Folder to C-BRANDS's folder, i.e.
  `C:\Users\UserName\Documents\MATLAB\C-BRANDS`

5. In MATLAB's Command Window, type and execute the following command

   `>> system('C:\cygwin\bin\bash singular "SingularTest.sh"')`

   You should receive the following output in MATLAB's Command Window which indicates that the SINGULAR computer algebra can be accessed sucessfully from MATLAB.

   ```
                       SINGULAR                                 /
    A Computer Algebra System for Polynomial Computations      /   version 3-1-6
                                                              0<
    by: W. Decker, G.-M. Greuel, G. Pfister, H. Schoenemann    \   Dec 2012
   FB Mathematik der Universitaet, D-67653 Kaiserslautern       \
   Auf Wiedersehen.

   ans =

        0
   ```

6. Run example files in folder `C-BRANDS/examples` to get familiar with C-BRANDS.

C.2.2 Polynomial Representation

Polynomials in C-BRANDS are represented as symbolic object according to the standard of MATLAB Symbolic Math Toolbox. To avoid an ambiguity in symbolic variables' declaration and ordering in MATLAB and Singular, the current version of C-BRANDS requires that polynomials are defined in polynomial ring $\mathbb{K}(k)[x]$ where

$\mathbb{K}$ is a field, $k = (k_1, \ldots, k_m)^T \in \mathbb{K}^m$ is a vector of unknown parameters and $x = (x_1, \ldots, x_n)^T \in \mathbb{R}^n$ is a vector of indeterminate variables in the ring.

To define a scalar polynomial or vector of polynomial functions, one should first declare the unknown parameters/variables and then apply standard algebraic operations that define the functions. For example, to define a vector of polynomial functions $f(k)[x]$ with unknown parameters k_i, $i = (1, \ldots, 4)$ and indeterminate variables $x = (x_1, x_2)^T$ of the form

$$f(x_1, x_2) = \begin{bmatrix} f_1(x_1, x_2) \\ f_2(x_1, x_2) \end{bmatrix} = \begin{bmatrix} k_1 x_1 - k_2 x_1 x_2 \\ k_3 x_1 x_2 - k_4 x_2 \end{bmatrix},$$

we first declare the unknown parameters k and the indeterminate variables x by typing

```
>> syms k1 k2 k3 k4 x1 x2;
```

and then construct polynomial $f(x_1, x_2)$ using the command

```
>> f = [k1*x1 - k2*x1*x2; k3*x1*x2 - k4*x2]

f =

    k1*x1 - k2*x1*x2

    k3*x1*x2 - k4*x2
```

Alternatively, one may use the vector representation of x and k as follows.

```
>> x = sym('x',[1,2]); k = sym('k',[1,4]);

>> f = [k(1)*x(1) - k(2)*x(1)*x(2); k(3)*x(1)*x(2) - k(4)*x(2)];
```

Polynomial representation `f` as given above can be manipulated in MATLAB using standard scalar or vectorial operators. For example, the Jacobian matrix $J = \frac{\partial f}{\partial x}$ of $f(k)[x]$ with respect to indeterminate variables x can be computed using the command

```
>> J = jacobian(f,x)

J =
```

```
[ k1 - k2*x2, -k2*x1]

[ k3*x2, k3*x1 - k4]
```

and the coefficients $a = [a_0, a_1, a_2]$ of the characteristic polynomial $p(s)$ of the form

$$p(s) = \det(sI - J) = a_0 + a_1 s + a_2 s^2$$

of the Jacobian matrix J can be computed using the command

```
>> a = charpoly(J)

a =

 [ 1, k4 - k1 + k2*x2 - k3*x1, k1*k3*x1 - k1*k4 + k2*k4*x2]
```

For other types of symbolic object manipulations and descriptions, please refer to the MATLAB Symbolic Math Toolbox documentation which can be accessed using the command

```
>> doc sym
```

C.3 Method and Implementation

C.3.1 Basic Theory

C-BRANDS can be used to study nonnegative polynomial dynamical systems modedeled as ordinary differential equations (ODE) of the form

$$\dot{x}(t) = f(x, k), \quad x(0) = x_0. \tag{C.1}$$

for $t \geq 0$ where $x \in \mathbb{R}^n$ is a vector of state variables, $k \in \mathbb{R}^m$ is a vector of unknown parameters, $f(x, k) \in \mathbb{R}(k)[x]$ is polynomial function in polynomial ring $\mathbb{R}(k)[x]$ (i.e. the set of all polynomials in indeterminate variables $x = (x_1, \ldots, x_n)$ with unknown real-valued coefficients $k = (k_1, \ldots, k_m)$). System (C.1) is said to be *nonnegative* if

188

and only if $x(t, x_0) \in \mathbb{R}_+^n$ for all $x_0 \in \mathbb{R}_+^n, t \geq 0$. A necessary and sufficient condition for (C.1) to be non-negative is that $f(x, k) \geq 0$ for all x in which $x = 0$ [65, 64].

The analysis of local bifurcation and robustness of ODE model (C.1) in C-BRANDS is carried out by studying the *kinetic realization* of (C.1) of the form

$$\dot{x}(t) = Nv(x, k) = N\text{diag}(k)x^Z, \quad x(0) = x_0$$
$$w^T x(t) = c,$$
(C.2)

where N is a matrix with integer coefficients, $v(x, k)$ is a vector of monomials describing the *flux* in the system, Z is a matrix of nonnegative integers whose ith column correspond to the x's multi-index of the ith monomial in $v(x, k)$ (i.e. the flux vector $v(x, k)$ satisfies a decomposition of the form $v(x, k) = \text{diag}(k)x^Z$). The equation $w^T x(t) = c$ is known as the conservation relation in the system

Remark C.3.1. Beside the ODE model in equation (C.1), C-BRANDS can also be used to analyze chemical reaction network (CRN) described by a set of r elementary reactions between $s \geq 1$ chemical *species* $X_1, X_2, \ldots, X_s$ described by the following CRN

$$\sum_{j=1}^{s} a_{ij}X_j \xrightarrow{k_i} \sum_{j=1}^{s} b_{ij}X_j, \quad \text{for } i = 1, 2, \ldots, r,$$
(C.3)
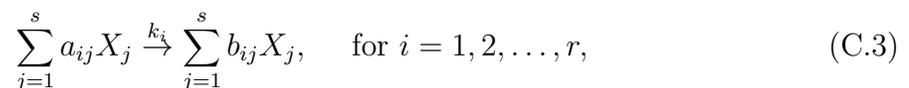
where parameter k_i's are the *rate constant* of the ith reaction between the *reactant species* $\sum_{j=1}^{s} a_{ij}X_j$ and the *product species* $\sum_{j=1}^{s} b_{ij}X_j$. By denoting the concentration of species X_j at time t as $x_j(t)$ and defining $x(t) = [x_1(t), \ldots, x_s(t)]^T$, a kinetic realization of the form (C.2) for CRN in (C.3) can be constructed (cf. [57, 58]).

As discussed in [57, 58], the main advantage of using the kinetic realization in (C.2) is that it allows one to obtain an expression of the state equilibria $x^*(k, \lambda) \in \mathbb{Q}(k, \lambda)$ as a rational function of the system parameter k and some convex parameter λ. The special structure in (C.2) means that the system's vector fields $f(x, k)$ may be

decomposed into a linear map (N) which is independent of x and k and a vector of nonlinear fluxes (v). This decomposition has two important consequences that were originally exploited in [57, 58]

First: it means that the *equilibrium fluxes* (i.e. those v^* such that $Nv^* = 0$) are non-negative vectors lying in the null space of N. In particular, any equilibrium flux must lie in a convex polyhedral cone

$$v^* \in K_v = \ker(N) \cap \mathbb{R}_+^m = \left\{ v \in \mathbb{R}_+^m : v = \Sigma_{i=1}^q \lambda_i E_i \right\}. \tag{C.4}$$

The cone, K_v, in equation (C.4) is finitely generated by a set of extreme rays, $E_i \in \mathbb{R}_+^m$ for $i = 1, 2, \ldots, q$. Such rays are routinely computed using tools such as CellNetAnalyzer [87]. Every equilibrium flux in K_v can therefore be parameterized with respect to these rays. In equation (C.4), the parameters $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_q)$ are called *convex parameters* [22] and so any equilibrium flux can be written as $v^*(\lambda)$ a linear function of the convex parameters.

Second: any flux, v, in the system must satisfy the equation

$$v_i = k_i x^{z_i}, \quad i = 1, 2, \ldots, m, \tag{C.5}$$

where z_i is the ith column of Z in (C.2). The equations in (C.5) are *binomials* in $\mathbb{R}(k)[x, v]$ and this system's zeros characterize both the equilibrium fluxes, v^*, and the state equilibria x^*. The ideal generated by these binomials is a *toric ideal* [7] for which efficient algorithms for computing a Gröbner basis are available [140] and have been implemented in computer algebra programs such as SINGULAR [62]. One can therefore solve for the equilibria of the system in terms of its equilibrium fluxes and system parameters.

The preceding two consequences of kinetic realization (C.2) can be summarized

as 1) any flux equilibrium can be expressed as a function, $v^*(\lambda)$, in terms of the convex parameters and 2) any state equilibrium can be expressed as a rational function, $x^*(v^*, k)$ of the equilibrium and the system parameters, k. Using the convex parameterization of v^*, one may then parameterize the equilibrium as $x^*(k, \lambda) \in \mathbb{Q}^n(k, \lambda)$. This algebraic equation characterizes all system equilibria as a function of the system and convex parameters and it provides a critical starting point for characterizing the bifurcation constraints in the distance-to-bifurcation problem.

The above equilibrium parameterization implies that the Jacobian matrix J of the system can also be parameterized in term of parameters k and λ as follows [57, 58].

$$J(\lambda, h) = N \text{diag}(v^*(\lambda)) Z^T \text{diag}(h), \tag{C.6}$$

where $h = 1/x^*(\lambda, k)$ and $v^*(\lambda)$ is a parameterization of the flux equilibrium in (C.4). Given the parameterization of Jacobian matrix in equation (C.6), one may then analyze the robustness and local bifurcation of system (C.1) using techniques from robust stability and local bifurcation analyses.

Consider the Jacobian matrix J in (C.6) and let $p(s) = |sI - J|$ be its characteristic polynomial defined as

$$q(s) = a_0 s^n + a_1 s^{n-1} + \cdots + a_{n-1} s + a_n, \tag{C.7}$$

where the coefficients $a_i(\lambda, h)$ are polynomial functions of the parameters (λ, h). For notational convenience, we denote these parameters as $\mu = (\lambda, h) \in \mathbb{R}^p$ where p denotes the number of parameters in μ. For $z = 1, \ldots, n$, let $\triangle_z$ denotes the zth Hurwitz determinant associated with $q(s)$. The following Routh-Hurwitz criteria provides a necessary and sufficient condition for robust stability of system (C.1).

Theorem C.3.2 (Routh-Hurwitz stability criteria)**.** *System (C.1) with Jacobian matrix J in (C.6) is robustly stable if and only if the Hurwitz determinant $\triangle_z$ of the*

characteristic polynomial $q(s)$ in (C.7) satisfy $\triangle_z > 0$ for all $z = 1, \ldots, n$

On the other hand, the necessary condition for system (C.1) to undergoes a local bifurcation (i.e. Hopf or saddle-node bifurcations) is given in the following theorem.

Theorem C.3.3 (Necessary local bifurcation condition). *System (C.1) with Jacobian matrix J in (C.6) will undergoes a Hopf or saddle-node bifurcation if there exists parameter μ such that $\mu \in \Omega(\mu)$ where $\Omega(\mu) = \Omega^H \cup \Omega^{SN}$ with $\Omega^H = \{ \mu \in \mathbb{R}^p_+ \mid \triangle_{n-1}(\mu) = 0 \}$ and $\Omega^{SN} = \{ \mu \in \mathbb{R}^p_+ \mid a_n(\mu) = 0 \}$ denotes the set of parameters at which a Hopf or a saddle-node bifurcation occurs.*

Using the condition in Theorem C.3.3, one may also formulate and solve the distance-to-bifurcation problem for system (C.1). In particular, consider the dynamical system (C.1) with a nominal parameter μ^0 for which the system has an asymptotically stable equilibrium $x^*(\mu^0)$. The minimum distance-to-bifurcation $\gamma = \inf_\mu |\mu^0 - \mu^*|$ is defined as the distance between the nominal parameter μ^0 and the closest critical parameter μ^* at which the corresponding system equilibrium $x^*(\mu^*)$ undergoes a bifurcation. Using the condition in Theorem C.3.3, a lower bound on γ for a particular local bifurcation can be computed by solving a sum of squares (SOS) optimization problem. For the case of Hopf bifurcation, a lower bound on β is given by the solution to the following SOS optimization problem.

$$\begin{aligned} \max \quad & \gamma, \\ \text{s.t.} \quad & V(\mu) - \gamma + r(\mu)\triangle_{n-1}(\mu) \quad \text{is SOS,} \end{aligned} \tag{C.8}$$

where $V(\mu), r(\mu)$ are some polynomials parameterization in indeterminates μ of the form

$$V(\mu) = \Sigma_\alpha \, c_\alpha \mu^\alpha, \qquad r(\mu) = \Sigma_\beta \, c_\beta \mu^\beta,$$

whose coefficients c_α and c_β, respectively, are decision variables to be determined

in the optimization. Similarly, a lower bound on γ for the case of a saddle-node bifurcation can be computed by solving the following SOS optimization problem.

$$
\begin{aligned}
&\max \quad \gamma, \\
&\text{s.t.} \quad V(\mu) - \gamma + r(\mu)a_n(\mu) \quad \text{is SOS.}
\end{aligned}
\tag{C.9}
$$

Notice that the SOS optimization problems as defined above can be solved using SOS programming tool such as SOSTOOLS [121] in conjunction with semidefinite programming solvers such as Sedumi [139].

C.3.2   What C-BRANDS Does

C-BRANDS implements the method described in the previous section by taking advantage of several third party toolboxes including SINGULAR, CellNetAnalyzer, SOSTOOLS and Sedumi. As illustrated in Figure C.1, the robustness/bifurcation analysis using C-BRANDS can be devided into four major steps, namely the construction of kinetic realization (C.2), the construction of robustness/bifurcation conditions, the computation of equilibrium parameterization $x^*(\lambda, k)$ and the computation of minimum distance to bifurcation (D_2B). Each step is described below.

C.3.2.1   Construction of Kinetic Realization

There are two types of system descriptions that can be analyzed in C-BRANDS: an ODE model of the form (C.1) or a set elementary reactions in CRN (C.3). In either case, C-BRANDS can be used to construct kinetic realizations of the form (C.2).

For the case in which the input is an ODE model, let us consider an example using the well-known Lotka-Volterra model of predator (x_1) and prey (x_1) interaction of
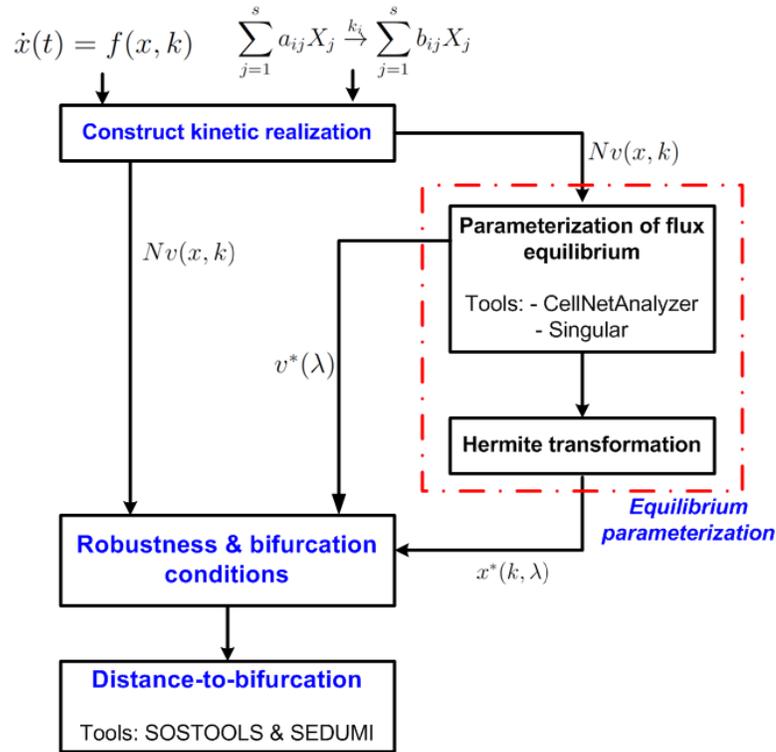
Figure C.1. Flow chart for using C-BRANDS.

the form

$$\dot{x}_1 = k_1 x_1 - k_2 x_1 x_2,$$

$$\dot{x}_2 = k_3 x_1 x_2 - k_4 x_2,$$

where k_i, $(i = 1, \ldots, 4)$ are some nonnegative parameters. To define this model in C-BRANDS, one first declare the set of indeterminate variables $x = (x_1, x_2)$ and the set of unknown parameters $k = (k_1, \ldots, k_4)$ and then define the polynomial function $f(x, k)$ according to the system model. This can be done using the following MATLAB commands.

```
>> x = sym('x',[1,2]); k = sym('k',[1,4]);
>> f = [k(1)*x(1) - k(2)*x(1)*x(2); k(3)*x(1)*x(2) - k(4)*x(2)]
```

```
f =
```

```
 k1*x1 - k2*x1*x2

 k3*x1*x2 - k4*x2
```

A kinetic realization of the form (C.2) for this system can be constructed using the function d2b␣ODE2realization as follows

```
>> NVKZ = d2b_ODE2realization(f,x,k)


NVKZ =


      v: [4x1 sym]

      k: [4x1 sym]

      Z: [2x4 double]

      N: [2x4 double]

    Inv: 'No conservation relation'
```

The output NVKZ of function d2b␣ODE2realization is a data structure that stores the values of matrices/vectors $N, v(x, k), k, Z$ that corresponds to kinetic realization (C.2). They can be accessed using standard MATLAB commands as follows.

```
>> NVKZ.N


ans =


     0     0    -1     1

    -1     1     0     0


>> NVKZ.v
```

```
ans =


    k4*x2

  k3*x1*x2

  k2*x1*x2

    k1*x1


>> NVKZ.k


ans =


  k4

  k3

  k2

  k1


>> NVKZ.Z


ans =


     0     1     1     1

     1     1     1     0
```

Thus a kinetic realization of the form (C.2) for this system is given by

$$
\dot{x} =
\begin{bmatrix}
0 & 0 & -1 & 1 \\
-1 & 1 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
k_4 x_2 \\
k_3 x_1 x_2 \\
k_2 x_1 x_2 \\
k_1 x_1
\end{bmatrix}
$$

$$
=
\begin{bmatrix}
0 & 0 & -1 & 1 \\
-1 & 1 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
k_4 & 0 & 0 & 0 \\
0 & k_3 & 0 & 0 \\
0 & 0 & k_2 & 0 \\
0 & 0 & 0 & k_1
\end{bmatrix}
\begin{bmatrix}
x_2 \\
x_1 x_2 \\
x_1 x_2 \\
x_1
\end{bmatrix}
$$

The last data `Inv` in `NVKZ` data structure contains the conservation relation $w^T x = c$ in model (C.2). The output of function `d2b_ODE2realization` as shown above indicates that the kinetic realization of the Lotka-Volterra model has no conservation relation.

For the case in which the input is a CRN, let us consider the following reaction network between species X, Y with concentration $[X]$ and $[Y]$, respectively.

$$
2X + Y \xrightarrow{k_1} 3X
$$

$$
\emptyset \underset{k_3}{\overset{k_4}{\rightleftarrows}} X \xrightarrow{k_2} Y
$$
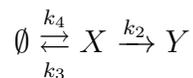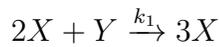
Let x_1, x_2 denote $[X], [Y]$, respectively. The kinetic realization of the form (C.2) for this CRN can be construced using the function `d2b_CRN2realization` as follows

```
>> x = sym('x',[1,2]); k = sym('k',[1,4]);
```

```
>> R = [2*x(1)+x(2),    3*x(1),      k(1);...
>>      x(1),           x(2),        k(2);
>>      x(1),           0,           k(3);
```

197

```
>>      0,              x(1),        k(4)];
```

```
>> NVKZ = d2b_CRN2realization(R)
```

```
NVKZ =


      N: [2x4 double]
      v: [4x1 sym]
      k: [4x1 sym]
      Z: [2x4 double]
    Inv: 'No conservation relation'
```

The above MATLAB script indicates that one first declare the set of indeterminate variables $x = (x_1, x_2)$ (i.e. vector of species concentrations) and the set of unknown parameters $k = (k_1, \ldots, k_4)$ (i.e. reaction constants) in the CRN. All r elementary reactions are then declared as a $3 \times r$ vector R in which the first and second columns of R contains the term on the left- (reactant complexes) and right- (product complexes) hand sides of each elementary reaction, respectively, whereas the third column of R contains the reaction constant of each elementary reaction. Similar to the case where systems described by an ODE model, the output of function d2b_CRN2realization is also a data structure NVKZ containing matrices/vectors $N, v(x, k), k, Z$ that correspond to the kinetic realization (C.2). Using the same method as described above to access the content of NVKZ data, it can be verified that a kinetic realization for the

above CRN is given by

$$\dot{x} = \begin{bmatrix} 1 & -1 & -1 & 1 \\ -1 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} k_1 x_1^2 x_2 \\ k_2 x_1 \\ k_3 x_1 \\ k_4 \end{bmatrix},$$

$$= \begin{bmatrix} 1 & -1 & -1 & 1 \\ -1 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} k_1 & 0 & 0 & 0 \\ 0 & k_2 & 0 & 0 \\ 0 & 0 & k_3 & 0 \\ 0 & 0 & 0 & k_4 \end{bmatrix} \begin{bmatrix} x_1^2 x_2 \\ x_1 \\ x_1 \\ 1 \end{bmatrix}.$$

It is easy to see that the ODE model for the above CRN is then given by

$$\dot{x}_1 = k_4 - k_2 x_1 - k_3 x_1 + k_1 x_1^2 x_2,$$

$$\dot{x}_2 = -k_1 x_2 x_1^2 + k_2 x_1.$$

C.3.2.2 Construction of Robustness/Bifurcation Conditions

Using the obtained kinetic realization described in the previous sections, then the robust stability or bifurcation conditions in Theorems C.3.2 and C.3.3 can be constructed from the coefficients and Hurwitz determinants of the characteristic polynomials (C.7) that corresponds to the parameterized Jacobian matrix in (C.6). These constructions can be done using C-BRANDS as illustrated below using the kinetic realization of the CRN discussed in the previous section.

First, the the parameterized Jacobian matrix J in (C.6) and the parameterized flux equilibrium $\mathcal{K}_v$ in (C.4) for the CRN model described in the previous section can be computed using the matlab function d2b_jacobian as follows

The outputs J and Kv of function d2b_jacobian above correspond to matrix J in (C.6) and vector $\mathcal{K}_v$ in (C.4), respectively.

The coefficients `a` and Hurwitz determinant `detH` of the Jacobian's characteristic polynomial $q(s)$ can be computed using function `d2b_hurwitz`. Vector $\mathcal{K}_v$ will later be used to compute the equilibrium parameterization $x^*(\lambda, k)$ (see section C.3.2.3) whereas matrix J will be used to compute characteristic polynomial $q(s)$ in (C.7) as shown below.

```
>> [detH,a] = d2b_hurwitz(J,NVKZ.Inv)


detH =


[                         h_1*lam_2 - h_1*lam_1 + h_2*lam_1, 1]
[ h_1*h_2*lam_1*lam_2*(h_1*lam_2 - h_1*lam_1 + h_2*lam_1), 1]


a =


[ 1, h_1*lam_2 - h_1*lam_1 + h_2*lam_1, h_1*h_2*lam_1*lam_2]
```

The Hurwitz determinant data `detH` in the output of function `d2b_hurwitz` shown above is interpreted as follows. The first column in `detH` is the nth Hurwitz determinant of J. Thus in our example we have that

$$\triangle_1 = h_1\lambda_2 - h_1\lambda_1 + h_2\lambda_1, \qquad \triangle_2 = h_1 h_2 \lambda_1 \lambda_2 (h_1\lambda_2 - h_1\lambda_1 + h_2\lambda_1).$$

The second column in `detH` contains logical indexes indicating whether the corresponding zth Hurwitz determinant always positive (index 0) or not (index 1) for any nonnegative values of λ and h. As indicated in `detH` data, both the first and the second Hurwitz determinants can take values less than or equal to zero for some choice of nonnegative parameters h and λ. Based on the condition stated in Theorem C.3.2, the CRN described above is robustly stable if and only if $\triangle_i > 0$, $(i = 1, 2)$.

The output data `a` of the function `d2b_jacobian` shown above suggests that the characteristic polynomial for the Jacobian matrix of our CRN model takes the form $q(s) = s^2 + a_1 s + a_2$ where

$$a_1 = h_1 \lambda_2 - h_1 \lambda_1 + h_2 \lambda_1, \qquad a_2 = h_1 h_2 \lambda_1 \lambda_2$$

Using the condition in Theorem C.3.3, one may concludes that this system cannot exhibits a saddle-node bifurcation since $a_n = a_2 \neq 0$ for any choice of positive parameters λ and h. This system, however, may exhibits a Hopf bifurcation if $\triangle_{n-1} = \triangle_1 = 0$ for some choice of nonnegative parameters λ and h.

C.3.2.3   Computation of Equilibria

The computation of equilibrium parameterization $x^*(\lambda, k)$ in C-BRANDS can be done using function `d2b_equi`. For the above CRN model, the following command can be used.

```
>> TG = d2b_toricGroebner(x,k,NVKZ.v);
>> [G,vw,xw] = d2b_equi(x,k,TG,Kv,NVKZ.Z)


G =


         lam_2 - k4
 k3*lam_1 - k2*lam_2



vw =


 w1
```

```
w2

w2

 1
```

```
xw =
```

```
      w2

 w1/w2^2
```

The output `TG` of `d2b_toricGroebner` above denotes a Gröbner basis of the binomial system (C.5). The output `G` of `d2b_equi`denotes the intersection between Gröbner basis `TG` and the flux equilibrium $\mathcal{K}_v$ in (C.4) (cf. `Kv` in section C.3.2.2). One may then rewrite the flux equilibrium v^* as

$$
v^* = \begin{bmatrix} k_2 k_4 / k_3 \\ k_2 k_4 / k_3 \\ k_4 \\ k_4 \end{bmatrix}.
$$

On the other hand, the outputs `vw` and `xw` on the other hand are variable transformation (obtained using Hermite transformation, see) that can be used to compute x^* from v^*. By equating `diag(NVKZ.k)*vw` and v^* as given above, it can be shown that

$$
\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} k_2 k_4 / k_1 k_3 \\ k_4 / k_3 \end{bmatrix}
$$

from which the equilibrium parameterization can be obtained using the output data `xw` as follows

$$x^* = \begin{bmatrix} k_4/k_1 \\ k_2 k_4/k_1 k_3 \end{bmatrix}.$$

C.3.2.4 Computation of distance-to-bifurcation

For our CRN model, a lower bound on the minimum distance γ to a Hopf bifurcation from a nominal parameter μ^0 can then be computed by solving the following SOS optimization (cf. (C.8)).

$$\begin{aligned} \max \quad & \gamma, \\ \text{s.t.} \quad & V(\mu) - \gamma + r(\mu)[h_1\lambda_2 - h_1\lambda_1 + h_2\lambda_1] \quad \text{is SOS.} \end{aligned}$$

Note that for any set of nominal parameters $\mu^0 = (k^0, h^0)$, the corresponding initial convex parameter λ^0 can be computed using equations x^* and the Gröbner basis `G`.

C.4 Applications

For detailed explanation on each of these examples, please refer to the example section in [146].

C.4.1 Robustness Analysis

C.4.1.1 Lotka-Volterra predator and prey model

Consider the Lotka-Volterra model for predator and prey interaction as follows.

$$\dot{x}_1 = k_1 x_1 - k_2 x_1 x_2,$$

$$\dot{x}_2 = k_3 x_1 x_2 - k_4 x_2,$$

where k_1, k_2, k_3, k_4 are some positive constants. Using MATLAB codes in Section C.5.1, it can be concluded that this system exhibits a robust oscillatory dynamics for any choice of nonnegative parameters k_i, $(i = 1, \ldots, 4)$.

C.4.1.2 A minimal lake model

Consider the following minimal lake model from [152].

$$\dot{x}_1 = k_1 x_1 x_2 - k_2 x_1,$$

$$\dot{x}_2 = -k_1 x_1 x_2 + x_3 \mu(x_4),$$

$$\dot{x}_3 = k_2 x_1 - x_3 \mu(x_4),$$

$$\dot{x}_4 = k_3(k_4 - x_4) + k_5 x_1 - k_6 x_3 \mu(x_4),$$

where x_1, x_2, x_3, x_4 denote the concentrations of autotroph, nutrient, detritus, and dissolved oxygen, respectively, k_i $(i = 1, \ldots, 6)$ are some positive constants, and $\mu(x_4) = x_4$ is a function describing the oxygen transformation (cf. [152] for details). Using MATLAB codes in Section C.5.2, it can be concluded that this system is robustly asymptotically stable for any choice of nonnegative parameters k_i, $(i = 1, \ldots, 6)$.

C.4.1.3 Feedback inhibition pathway model

Consider a chemical reaction network for the feedback inhibition pathway below.

$$\emptyset \xrightarrow{k_9} x_1 \xrightarrow{k_8} \emptyset$$

$$x_1 + x_6 \xrightarrow{k_1} x_2 + x_6$$

$$x_2 \xrightarrow{k_2} x_3 \xrightarrow{k_3} x_4 \xrightarrow{k_4} x_5 \xrightarrow{k_5} \emptyset$$

$$x_5 + x_6 \underset{k_7}{\overset{k_6}{\rightleftharpoons}} x_7$$
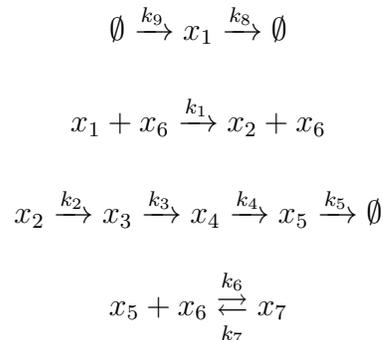
Using MATLAB codes in Section C.5.3, it can be concluded that this system is robustly asymptotically stable for any choice of nonnegative rate constants k_i, $(i = 1, \ldots, 9)$.

C.4.2 Distance-to-bifurcation problem

C.4.2.1 Hopf bifurcation

Consider the model of the Brusselator described in the previous section. Let the nominal operating point be given by initial parameter $k_i^0 = 1$ for $i = (1, \ldots, 4)$ and initial condition $x_1(0) = x_2(0) = 1$ for which the corresponding nominal equilibrium is $x_1^* = x_2^* = 1$. Using MATLAB codes in Section C.5.4, we found that a lower bound on the minimum distance a Hopf bifurcation for this system is $\gamma = 0.4$.

C.4.2.2 Saddle-node bifurcation

Consider a model of the peroxidase-oxidase system below

$$\dot{x}_1 = k_2 + k_3 x_1 - 2k_1 x_1^2 - k_4 x_1 x_2$$

$$\dot{x}_2 = k_5 - k_6 x_2 - k_4 x_1 x_2$$

Let us consider an initial condition $x^0 = [0.5, 1.5]$ and a nominal parameter $k^0 = [2, 3, 3, 2, 3, 1]$ which correspond to a nominal stable equilibrium point $x_j^* = 1$, $(j = 1, 2)$. Using MATLAB codes in Section C.5.5, we found a lower bound on the minimum distance a saddle-node bifurcation for this system is $\gamma = 1.5473$.

C.5 Matlab codes for the examples

To run these examples, MATLAB's Current Folder should be set to the C-BRANDS's folder.

C.5.1 Lotka-Volterra predator and prey model

```
clear all; close all; clc


x = sym('x',[1,2]);
k = sym('k',[1,4]);


f = [k(1)*x(1) - k(2)*x(1)*x(2);...
    k(3)*x(1)*x(2) - k(4)*x(2)];


% Construct the kinetic realization
NVKZ = d2b_ODE2realization(f,x,k);


% Jacobian matrix, characteristic polynomial, and Hurwitz determinant
[Jac,Kv,E] = d2b_jacobian(NVKZ.N,NVKZ.Z);
[detH,Cpoly] = d2b_hurwitz(Jac,NVKZ.Inv);


display(Cpoly);
```

C.5.2 Minimal lake model

```
clear all; close all; clc


x = sym('x',[1,4]);
k = sym('k',[1,6]);
```

```matlab
f = [k(1)*x(1)*x(2) - k(2)*x(1);...

    -k(1)*x(1)*x(2) + x(3)*x(4);...

    k(2)*x(1) - x(3)*x(4);...

    k(3)*(k(4)-x(4)) + k(5)*x(1) - k(6)*x(3)*x(4)];


% Construct the kinetic realization

NVKZ = d2b_ODE2realization(f,x,k);


% Jacobian matrix, characteristic polynomial, and Hurwitz determinant

[Jac,Kv,E] = d2b_jacobian(NVKZ.N,NVKZ.Z);


% Test robust local stability

dec = d2b_isasystable(Jac,NVKZ.Inv);
```

C.5.3 Feedback inhibition pathway

```matlab
clear all; close all; clc;


x = sym('x',[1,7]);

k = sym('k',[1,9]);


% ODE model

f = [k(9) - k(8)*x(1) - k(1)*x(1)*x(6);...

    k(1)*x(1)*x(6) - k(2)*x(2);...

    k(2)*x(2) - k(3)*x(3);...

    k(3)*x(3) - k(4)*x(4);...

    k(4)*x(4) - k(5)*x(5) - (k(6)*x(5)*x(6) - k(7)*x(7));...
```

```matlab
    -k(6)*x(5)*x(6) + k(7)*x(7);...

    k(6)*x(5)*x(6) - k(7)*x(7)];


% Kinetic realization

NVKZ = d2b_ODE2realization(f,x,k);


% Compute steady state Jacobian matrix

[Jss,Kv,E] = d2b_jacobian(NVKZ.N,NVKZ.Z);


% Test robust local stability

dec = d2b_isasystable(Jss,NVKZ.Inv);
```

C.5.4   Brusselator

```matlab
clear all; close all; clc;


% states and parameters

x = sym('x',[1,2]);

k = sym('k',[1,4]);


% Elementary reactions network

R = [2*x(1)+x(2),    3*x(1),      k(1);...

    x(1),           x(2),        k(2);

    x(1),           0,           k(3);

    0,              x(1),        k(4)];

tic

% Kinetic realization

[NVKZ,f] = d2b_CRN2realization(R);
```

```matlab
% Jacobian matrix, characteristic polynomial, and Hurwitz determinant
[Jss,Kv,E] = d2b_jacobian(NVKZ.N,NVKZ.Z);
[detH,Cpoly] = d2b_hurwitz(Jss,NVKZ.Inv);


% Construct local bifurcation condition
[HB,SNB] = d2b_Instability(detH,Cpoly);


% Compute Grobner basis of the toric ideal and Hermite transformation
TGB = d2b_toricGroebner(x,k,NVKZ.v);
[Gn,vw,xw] = d2b_equi(x,k,TGB,Kv,NVKZ.Z);


% Equilibrium parameterization
lam_vars = setdiff(symvar(Gn),k);
lam = solve(Gn,lam_vars(1),lam_vars(2));
Kv  = subs(Kv,lam_vars,[lam.lam_1, lam.lam_2]);
w_lam = diag(k)*vw - Kv; w = w_lam(w_lam~=0);
wvars = setdiff(symvar(w),k);
wsol = solve(w,wvars(1),wvars(2));
Equi = subs(xw,symvar(xw),[wsol.w1,wsol.w2]);


%% Minimum Distance to Hopf bifurcation
SOSvar = symvar(HB.Eq);
Obj = sum((SOSvar - ones(1,4)).^2);
Eq = SOSvar(1)*SOSvar(4) - SOSvar(1)*SOSvar(3) + SOSvar(2)*SOSvar(3);
Ineq = SOSvar(1)*SOSvar(4) - SOSvar(1)*SOSvar(3) + SOSvar(2)*SOSvar(3);
[d2b,var,optval] = findbound(Obj,[SOSvar, Ineq],Eq,2);
```

```
display(d2b); toc
```

C.5.5   The peroxidase-oxidase reaction network

```
clear all; close all; clc;


% states and parameters

x = sym('x',[1,2]);

k = sym('k',[1,6]);


% vector fields

f = [-2*k(1)*x(1)^2 + k(2) + k(3)*x(1) - k(4)*x(1)*x(2);...

    k(5) - k(6)*x(2) - k(4)*x(1)*x(2)];

tic

% Construct the kinetic realization

NVKZ = d2b_ODE2realization(f,x,k);


% Jacobian matrix, characteristic polynomial, and Hurwitz determinant

[Jac,Kv,E] = d2b_jacobian(NVKZ.N,NVKZ.Z);

[detH,Cpoly] = d2b_hurwitz(Jac,NVKZ.Inv);


% Construct the bifurcation conditions

[HB,SNB] = d2b_Instability(detH,Cpoly);


% Compute toric variety and variable transformation

TGB = d2b_toricGroebner(x,k,NVKZ.v);

[Gn,vw,xw] = d2b_equi(x,k,TGB,Kv,NVKZ.Z);

Gn(4) = [];
```

```
% SOS program to compute distance to saddle-node bifurcation

k0 = [2, 3, 3, 2, 3, 1];    % initial parameter

x0 = [1 1];                 % equilibrium corresponding to k0

G = subs(Gn,k,k0);          % equation to determine initial lambda

lam0 = ones(1,5);           % initial lambda corresponding to k0


SOSvar = symvar(SNB);

Object = sum((SOSvar(3:end) - lam0).^2);

Eq = subs(SNB,SOSvar(1:2),ones(1,2));

[d2b,var,optval] = findbound(Object,SOSvar(3:end),Eq,2);

display(d2b); toc
```

C.6 Functions Descriptions

The main functions used in C-BRANDS are described as follows.

- `NVKZ = d2b_ODE2realization(f,x,k)`
  Function `d2b_ODE2realization` constructs a kinetic realization of the form (C.2) from an ODE model $\dot{x} = f(x, k)$. The inputs of this function are symbolic vector `f` of the vector field $f(x, k)$, symbolic vector `x` of indeterminate variables x and symbolic vector `k` of unknown parameters k. The output of this function is a data structure `NVKZ` that stores the matrices/vectors $N, v(x, z), k, Z$ in kinetic realizations (C.2).

- `[NVKZ,f,CRNTmod] = d2b_CRN2realization(R)`
  Function `d2b_CRN2realization` constructs a kinetic realization (C.2) from a set of r elementary reactions of the form (C.3). The input `R` to this function is a $(r \times 3)$ matrix in which r denotes the number of elementary reactions. The first and second columns of `R` correspond to the reactant and product complexes, repectively, in each elementary reactions, whereas the third column of `R` is the corresponding reactant constant. There are three outputs of this function namely `NVKZ, f` and `CNRTmod`. `NVKZ` is a data structure containing matrices/vectors $N, v(x, k), k, Z$ and w that correspond to kinetic realization in (C.2). The output `f` is a symbolic vector of vector field $f(x, k)$ that defines an ODE model $\dot{x} = f(x, k)$ for the input CRN defiines in `R`. The output `CRNTmod` is a data structure containing matrices/vectors $Y, I_a, I_k, \psi(x)$ that define alternative realization of the form $\dot{x} = Y I_a I_k \psi(x)$ which is often used in the chemical reaction network theory (CRNT) literatures (cf. [44]).

- `J = d2b_jacobian(N,Z)`
  Function `d2b_jacobian` constructs the parameterized Jacobian matrix (C.6) of a kinetic realization of the form (C.2). The inputs `N` and `Z` to this function are matrices N and Z, respectively, in kinetic realizations (C.2). The output `J` is a symbolic object of Jacobian matrix (C.6).

- `[dH,a] = d2b_hurwitz(J,w)`
  Function `d2b_hurwitz` computes the coefficients of the characteristic polynomial and its corresponding zth Hurwitz determinant. The inputs `J` and `w` of this function are the parameterized Jacobian matrix J in (C.6) and the conservation matrix `w` in (C.2). The output of this function is a vector `a` containing the coefficients a_i of the characteristic polynomial $q(s)$ in (C.7) and a vector `dH` containing the zth Hurwitz determinant of $q(s)$.

- `G = d2b_toricGroebner(x,k,v)`
  Function `d2b_toricGroebner` computes a Gröbner basis for the toric ideal formed by the kinetic realizations' flux vector. The inputs to this function are symbolic vector `x` of indeterminate variables x, symbolic vector `k` of unknown parameters k, and symbolic vector `v` of the flux $v(x, k)$ in (C.2). The output of

this function is a Gröbener basis G of the toric ideal formed by binomial system in (C.5).

- `[Gn,vw,vx] = d2b_equi(x,k,G,Kv,Z)`
  Function `d2b_equi` constructs the parameterized flux equilibria and the corresponding Hermite transformation [141] that can be used to obtain the equilibrium parameterization $x^*(\lambda, k)$. The inputs for this function are symbolic vector `x` of indeterminate variables x, symbolic vector `k` of unknown parameters k, symbolic vector `G` of the Gröbner basis G_n, symbolic vector `Kv` of the flux equilibrium parameterization $\mathcal{K}_v$, and matrix Z in kinetic realizations (C.2).

- `[HB, SNB] = d2b_Instability(dH,a)`
  Function `d2b_Instability` constructs the conditions for the Hopf and saddle-node bifurcations to occurs in system (C.2). The inputs `dH` and `a` to this function are the Hurwitz determinant $\triangle_z$ and coefficients a_i, respectively, of the characteristic polynomial $q(s)$ in (C.7). The outputs `HB` and `SNB` of this function are the conditions for the occurrence of Hopf and saddle-node bifurcations, respectively, stated in Theorem C.3.3.

DATA OF THE CHEMOSTAT EXPERIMENT & ANALYSIS

D.1 BBM-Medium used in the experiment

BOLD'S BASAL MEDIUM (modified)		
STOCK	SOLUTION	100% (mL/L)
1. KH_2PO_4	8.75 gr/500 mL	10 mL
2. $CaCl_2 \cdot 2\,H_2O$	1.25 gr/500 mL	10 mL
3. $MgSO_4 \cdot 7\,H_2O$	3.75 gr/500 mL	10 mL
4. $NaNO_3$	12.5 gr/500 mL	4 mL
5. K_2HPO_4	3.75 gr/500 mL	10 mL
6. NaCl	1.25 gr/500 mL	10 mL
7. Na_2EDTA KOH	10 gr/L 6.2 gr/L	1 mL
8. $FeSO_4\,7\,H_2O$ H_2SO_4 (concentrated)	4.98 gr/L 1 mL	1 mL
9. Trace Metal Solution	See below*	1 mL
10. H_3BO_3	5.75 gr/ 500 mL	0.7 mL

Instructions for making 2L of 50%: Add 1 L of MilliQ water to a 2L volumetric flask. Add the amount of each stock indicated in the above table for 100%. Add stocks one at a time using the pre-labelled 10 mL serological pipettes. Be sure to keep track of which ones you have added. Add 175 μL of 1M NaOH and 10 gr NaCL

crystal. Fill the flask to the line with MilliQ water and put the flask on magnetic stirrer for 10 minutes. When complete, pour into a 2L media bottle and autoclave on liquid setting for 30 minutes.

Trace Metal Solution*	
Substance	gr/L
1. H_3BO_3	2.86 gr
2. $MnCl_2 \cdot 4\,H_{20}$	1.81 gr
3. $ZnSO_4 \cdot 7\,H_2O$	0.222 gr
4. $Na_2MoO_4 \cdot 2\,H_2O$	0.39 gr
5. $CuSO_4 \cdot 5\,H_2O$	0.079 gr
6. $Co(NO_3)_2 \cdot 6\,H_2O$	0.0494 gr

D.2 Chemostat Measurements

D.2.1 Raw data for $\delta = 0.95$

The data were obtained from 45 days of chemostat measurements. The data were averages of measurements from two chemostats.

- The *C. vulgaris* measurements for each chemostat were the average number of cells in two samples (each of which was 10 microliters).

- The *B. calyciflorus* measurements for each chemostat were the average number of individuals in four samples (each of which was 1 milliliters).

Day	1	2	3	4	5	6	7	8	9	10
C. vulgaris	5437500	4512500	4637500	4725000	4125000	4287500	3775000	2175000	2062500	3137500
B. calyciflorus	0.9375	0.75	0.8125	1.8125	2.25	4.0625	4.8125	7.8125	5.5625	4.875

Day	11	12	13	14	15	16	17	18	19	20
C. vulgaris	4275000	4187500	4425000	4712500	4025000	4512500	4525000	4762500	4662500	5387500
B. calyciflorus	2.1875	1.9375	0.8125	1.0625	0.9375	0.375	0.25	0.1875	0.25	0.5625

Day	21	22	23	24.0	25	26	27	28	29	30
C. vulgaris	5037500	3737500	4412500	4325000	4862500	4287500	3875000	3625000	2975000	1750000
B. calyciflorus	0.3125	0.25	0.6875	0.4375	1.75	1.8125	3.1875	4.375	7.5	6.9375

Day	31	32	33	34	35	36	37	38	39	40
C. vulgaris	2712500	3712500	3725000	4312500	4187500	3912500	4537500	4462500	4812500	4937500
B. calyciflorus	5.375	3.9375	1.75	1.1875	0.75	1.125	0.9375	0.125	0.3375	0.1875

Day	41	42	43	44	45
C. vulgaris	5137500	4862500	4075000	5812500	4662500
B. calyciflorus	0.3125	0.8125	0.1875	0.8625	0.4375

D.2.2 Raw data for $\delta = 0.1$

The following data were obtained from 45 days of chemostat measurements.

The data were averages of measurements from two chemostats.

- The *C. vulgaris* measurements for each chemostat were the average number of cells in two samples (each of which was 10 microliters).

- The *B. calyciflorus* measurements for each chemostat were the average number of individuals in four samples (each of which was 1 milliliters).

Day	1	2	3	4	5.0	6	7	8	9	10
C. vulgaris	1762500	1687500	1587500	1762000	1712500	1612500	1887500	1737500	1675000	1575000
B. calyciflorus	1.875	1.75	1.875	1.6875	1.75	1.9375	2.3125	1.8125	1.875	1.9375

Day	11	12	13.0	14	15	16	17	18	19	20
C. vulgaris	1512500	1462500	1675000	1587500	1831250	1593750	1362500	1975000	1587500	1762500
B. calyciflorus	2.0625	2.125	2.0625	2.1875	1.9375	2.1875	1.8125	1.75	1.6875	1.5625

Day	21	22	23	24	25	26	27	28	29	30
C. vulgaris	1575000	1662500	1675000	1487500	1462500	1587500	1567500	1767500	1412500	1725000
B. calyciflorus	1.9375	1.9375	2.0625	1.9375	2.1875	2.0625	2	1.625	1.6875	1.875

Day	31	32	33	34	35	36	37	38.0	39	40
C. vulgaris	1537500	1662500	1512500	1762500	1512500	1537500	1712500	1687500	1525000	1537500
B. calyciflorus	2.125	1.6875	1.75	1.875	1.75	2.0625	1.9375	1.875	2.0625	1.75

Day	41.0	42	43	44	45
C. vulgaris	1512500	1762500	1612500	1562500	1662500
B. calyciflorus	2.0625	1.6875	1.9375	2.0625	1.875

D.3 Extreme Generators

$$E = [E_1,\, E_2,\, \ldots,\, E_{36}]$$

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

The convex cone $\mathcal{K}_v = \sum_{i=1}^{36} \lambda_i E_i$ defined by the extreme generators in (D.1) is

$$
\mathcal{K}_v(\lambda) =
\begin{bmatrix}
\lambda_3 + \lambda_4 \\
\lambda_1 + \lambda_2 \\
\lambda_5 + \lambda_6 + \lambda_7 + \lambda_8 \\
\lambda_9 + \lambda_{10} + \lambda_{11} + \lambda_{12} \\
\lambda_{21} + \lambda_{25} + \lambda_{26} + \lambda_{27} \\
\lambda_{13} + \lambda_{14} \\
\lambda_5 + \lambda_9 \\
\lambda_1 + \lambda_3 \\
\lambda_{15} + \lambda_{18} \\
\lambda_{13} + \lambda_{15} + \lambda_{16} + \lambda_{17} \\
\lambda_{17} + \lambda_{20} \\
\lambda_6 + \lambda_{10} \\
\lambda_2 + \lambda_4 \\
\lambda_{16} + \lambda_{19} \\
\lambda_{14} + \lambda_{18} + \lambda_{19} + \lambda_{20} \\
\lambda_{25} + \lambda_{28} + \lambda_{31} + \lambda_{34} \\
\lambda_{22} + \lambda_{28} + \lambda_{29} + \lambda_{30} \\
\lambda_{23} + \lambda_{31} + \lambda_{32} + \lambda_{33} \\
\lambda_{24} + \lambda_{34} + \lambda_{35} + \lambda_{36} \\
\lambda_{26} + \lambda_{29} + \lambda_{32} + \lambda_{35} \\
\lambda_{21} + \lambda_{22} + \lambda_{23} + \lambda_{24} \\
\lambda_7 + \lambda_{11} \\
\lambda_8 + \lambda_{12} \\
\lambda_{27} + \lambda_{30} + \lambda_{33} + \lambda_{36}
\end{bmatrix}
\tag{D.2}
$$

where λ_i, $(i = 1, \ldots, 36)$ are the convex parameters.

BIBLIOGRAPHY

1. E. D. Andersen and K. D. Andersen. The mosek interior point optimizer for linear programming: an implementation of the homogeneous algorithm. In *High performance optimization*, pages 197–232. Springer, 2000.

2. E. D. Andersen, J. Gondzio, C. Mészáros, and X. Xu. *Implementation of interior point methods for large scale linear programming*. HEC/Université de Genève, 1996.

3. A. Ang and A. Timmermann. Regime changes and financial markets. Technical report, National Bureau of Economic Research, 2011.

4. M. Barange, G. Beaugrand, R. Harris, R. I. Perry, M. Scheffer, F. Werner, et al. Regime shifts in marine ecosystems: detection, prediction and management. *Trends in Ecology & Evolution*, 23(7):402–409, 2008.

5. D. R. Bellwood, T. P. Hughes, C. Folke, and M. Nyström. Confronting the coral reef crisis. *Nature*, 429(6994):827–833, 2004.

6. F. D. Biase and R. Urbanke. An algorithm to calculate the kernel of certain polynomial ring homomorphisms. *Experimental Mathematics*, 4(3):227–234, 1995.

7. A. Bigatti and L. Robbiano. Toric ideals. *Matemática Contemporânea*, 21:1–25, 2001.

8. R. Biggs, S. R. Carpenter, and W. A. Brock. Turning back from the brink: detecting an impending regime shift in time to avert it. *Proceedings of the National academy of Sciences*, 106(3):826–831, 2009.

9. G. Blekherman, P. A. Parrilo, and R. R. Thomas. *Semidefinite Optimization and Convex Algebraic Geometry*, volume 13. SIAM, 2013.

10. J. Bochnak, M. Coste, M.-F. Roy, et al. *Real Algebraic Geometry*, volume 95. Springer Berlin, 1998.

11. S. P. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge university press, 2004.

12. S. P. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*, volume 15. SIAM, 1994.

13. P. Bremaud. *Markov Chains: Gibbs Fields, Monte Carlo Simulation, and Queues*, volume 31. Springer, 1999.

14. B. Buchberger. Bruno buchberger's phd thesis 1965: An algorithm for finding the basis elements of the residue class ring of a zero dimensional polynomial ideal. *Journal of Symbolic Computation*, 41(3):475–511, 2006.

15. C. A. Canizares. Calculating optimal system parameters to maximize the distance to saddle-node bifurcations. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 45(3):225–237, 1998.

16. S. Carpenter and W. Brock. Rising variance: a leading indicator of ecological transition. *Ecology Letters*, 9(3):311–318, 2006.

17. S. Carpenter and W. Brock. Early warnings of unknown nonlinear shifts: a nonparametric approach. *Ecology*, 92(12):2196–2201, 2011.

18. S. R. Carpenter. Eutrophication of aquatic ecosystems: bistability and soil phosphorus. *Proceedings of the National Academy of Sciences of the United States of America*, 102(29):10002–10005, 2005.

19. S. R. Carpenter, N. F. Caraco, D. L. Correll, R. W. Howarth, A. N. Sharpley, and V. H. Smith. Nonpoint pollution of surface waters with phosphorus and nitrogen. *Ecological Applications*, 8(3):559–568, 1998.

20. S. R. Carpenter, D. Ludwig, and W. A. Brock. Management of eutrophication for lakes subject to potentially irreversible change. *Ecological Applications*, 9 (3):751–771, 1999.

21. A. Chakraborty, P. Seiler, and G. J. Balas. Nonlinear region of attraction analysis for flight control verification and validation. *Control Engineering Practice*, 19(4):335–345, 2011.

22. B. L. Clarke. *Stability of Complex Reaction Networks*. Wiley Online Library, 1980.

23. B. L. Clarke and W. Jiang. Method for deriving hopf and saddle-node bifurcation hypersurfaces and application to a model of the belousov–zhabotinskii system. *The Journal of Chemical Physics*, 99:4464, 1993.

24. R. Contamin and A. M. Ellison. Indicators of regime shifts in ecological systems: what do we need to know and when do we need to know it. *Ecological Applications*, 19(3):799–816, 2009.

25. D. A. Cox, J. Little, and D. O'Shea. *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*, volume 10. Springer Verlag, 2007.

26. G. Craciun, Y. Tang, and M. Feinberg. Understanding bistability in complex enzyme-driven reaction networks. *Proceedings of the National Academy of Sciences*, 103(23):8697–8702, 2006.

27. V. Dakos, S. R. Carpenter, W. A. Brock, A. M. Ellison, V. Guttal, A. R. Ives, S. Kefi, V. Livina, D. A. Seekell, E. H. Van Nes, et al. Methods for detecting early warnings of critical transitions in time series illustrated using simulated ecological data. *PLoS One*, 7(7):e41010, 2012.

28. G. M. Daskalov. Overfishing drives a trophic cascade in the black sea. *Marine Ecology Progress Series*, 225:53–63, 2002.

29. G. M. Daskalov, A. N. Grishin, S. Rodionov, and V. Mihneva. Trophic cascades triggered by overfishing reveal possible mechanisms of ecosystem regime shifts. *Proceedings of the National Academy of Sciences*, 104(25):10518–10523, 2007.

30. E. De Klerk and M. Laurent. Error bounds for some semidefinite programming approaches to polynomial minimization on the hypercube. *SIAM Journal on Optimization*, 20(6):3104–3120, 2010.

31. W. Decker and C. Lossen. *Computing in Algebraic Geometry*, volume 16 of *Algorithms and Computation in Mathematics*. Springer Verlag, 2006.

32. W. Decker, G.-M. Greuel, G. Pfister, and H. Schönemann. Singular 3-1-6 — A computer algebra system for polynomial computations. 2012. http://www.singular.uni-kl.de.

33. P. Demenocal, J. Ortiz, T. Guilderson, J. Adkins, M. Sarnthein, L. Baker, and M. Yarusinsky. Abrupt onset and termination of the african humid period:: rapid climate responses to gradual insolation forcing. *Quaternary Science Reviews*, 19(1):347–361, 2000.

34. A. Dhooge, W. Govaerts, and Y. A. Kuznetsov. Matcont: a matlab package for numerical bifurcation analysis of odes. *ACM Transactions on Mathematical Software (TOMS)*, 29(2):141–164, 2003.

35. I. Dobson. Computing a closest bifurcation instability in multidimensional parameter space. *Journal of Nonlinear Science*, 3(1):307–327, 1993.

36. I. Dobson. Distance to bifurcation in multidimensional parameter space: Margin sensitivity and closest bifurcations. In *Bifurcation Control*, pages 49–66. Springer, 2003.

37. K. L. Drury and D. M. Lodge. Using mean first passage times to quantify equilibrium resilience in perturbed intraguild predation systems. *Theoretical Ecology*, 2(1):41–51, 2009.

38. K. L. Drury and D. M. Lodge. Using mean first passage times to quantify equilibrium resilience in perturbed intraguild predation systems. *Theoretical Ecology*, 2(1):41–51, 2009.

39. T. W. Dubé. The structure of polynomial ideals and gröbner bases. *SIAM Journal on Computing*, 19(4):750–773, 1990.

40. H. W. Engl, C. Flamm, P. Kügler, J. Lu, S. Müller, and P. Schuster. Inverse problems in systems biology. *Inverse Problems*, 25(12):123014, 2009.

41. B. Ermentrout. *Simulating, analyzing, and animating dynamical systems: a guide to XPPAUT for researchers and students*, volume 14. SIAM, 2002.

42. V. Fairén and B. Hernández-Bermejo. Mass action law conjugate representation for general chemical mechanisms. *The Journal of Physical Chemistry*, 100(49): 19023–19028, 1996.

43. V. Fairén and B. Hernandez-Bermejo. Mass action law conjugate representation for general chemical mechanisms. *The Journal of Physical Chemistry*, 100(49): 19023–19028, 1996.

44. M. Feinberg. Lectures on chemical reaction networks. *Notes of lectures given at the Mathematics Research Center, University of Wisconsin*, 1979.

45. J. A. Foley, M. T. Coe, M. Scheffer, and G. Wang. Regime shifts in the sahara and sahel: interactions between ecological and climatic systems in northern africa. *Ecosystems*, 6(6):524–532, 2003.

46. J. A. Foley, N. Ramankutty, K. A. Brauman, E. S. Cassidy, J. S. Gerber, M. Johnston, N. D. Mueller, C. OŠConnell, D. K. Ray, P. C. West, et al. Solutions for a cultivated planet. *Nature*, 478(7369):337–342, 2011.

47. C. Folke. Resilience: The emergence of a perspective for social–ecological systems analyses. *Global Environmental Change*, 16(3):253–267, 2006.

48. C. Folke, S. Carpenter, B. Walker, M. Scheffer, T. Elmqvist, L. Gunderson, and C. Holling. Regime shifts, resilience, and biodiversity in ecosystem management. *Annual Review of Ecology, Evolution, and Systematics*, pages 557–581, 2004.

49. A. Friedman. *Stochastic Differential Equations and Applications. Vol. 1*. Academic Press, New York, 1975.

50. G. F. Fussmann, S. P. Ellner, K. W. Shertzer, and N. G. Hairston Jr. Crossing the hopf bifurcation in a live predator-prey system. *Science*, 290(5495):1358–1360, 2000.

51. P. Gahinet, P. Apkarian, and M. Chilali. Affine parameter-dependent lyapunov functions and real parametric uncertainty. *Automatic Control, IEEE Transactions on*, 41(3):436–442, 1996.

52. F. Gantmacher. *The Theory of Matrices.* Taylor & Francis, 1960.

53. C. W. Gardiner. *Handbook of Stochastic Methods: For Physics, Chemistry, and the Natural Sciences.* Springer, New York, 1985.

54. C. W. Gardiner and C. Gardiner. *Stochastic Methods: A Handbook for the Natural and Social Sciences*, volume 4. Springer Berlin, 2009.

55. M. Gasca and T. Sauer. On the history of multivariate polynomial interpolation. *Journal of Computational and Applied Mathematics*, 122(1):23–35, 2000.

56. K. Gatermann. Counting stable solutions of sparse polynomial systems in chemistry. 286:53, 2001.

57. K. Gatermann and B. Huber. A family of sparse polynomial systems arising in chemical reaction systems. *Journal of Symbolic Computation*, 33(3):275–305, 2002.

58. K. Gatermann, M. Eiswirth, and A. Sensse. Toric ideals and graph theory to analyze hopf bifurcations in mass action systems. *Journal of Symbolic Computation*, 40(6):1361–1382, 2005.

59. F. Geier, G. Fengos, F. Felizzi, and D. Iber. Analyzing and constraining signaling networks: parameter estimation for the user. In *Computational Modeling of Signaling Networks*, pages 23–39. Springer, 2012.

60. S. Ghosh, A. K. Pal, and I. Bose. Noise-induced regime shifts: A quantitative characterization. *The European Physical Journal E*, 36(10):1–10, 2013.

61. D. R. Grayson and M. E. Stillman. Macaulay2, a software system for research in algebraic geometry. Available at http://www.math.uiuc.edu/Macaulay2/, 2013.

62. G.-M. Greuel, G. Pfister, and H. Schönemann. Singular—a computer algebra system for polynomial computations. In *Symbolic Computation and Automated Reasoning*, pages 227–233. AK Peters, Ltd., 2001.

63. J. Guckenheimer and P. Holmes. *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields.* Springer-Verlag, 1983.

64. W. M. Haddad and V. Chellaboina. Stability and dissipativity theory for non-negative dynamical systems: a unified analysis framework for biological and physiological systems. *Nonlinear Analysis: Real World Applications*, 6(1):35–65, 2005.

65. W. M. Haddad, V. Chellaboina, and Q. Hui. *Nonnegative and Compartmental Dynamical Systems.* Princeton University Press, 2010.

66. D. Handelman. Representing polynomials by positive linear functions on compact convex polyhedra. *Pacific Journal of Mathematics*, 132(1):35–62, 1988.

67. V. Hárs and J. Tóth. On the inverse problem of reaction kinetics. In *Colloquia Mathematica Societatis János Bolyai,(Szeged, Hungary, 1979) Qualitative Theory of Differential Equations (M. Farkas ed.)*, volume 30, pages 363–379, 1981.

68. A. Hastings and T. Powell. Chaos in a three-species food chain. *Ecology*, pages 896–903, 1991.

69. D. Henrion and J.-B. Lasserre. Gloptipoly: Global optimization over polynomials with matlab and sedumi. *ACM Transactions on Mathematical Software (TOMS)*, 29(2):165–194, 2003.

70. D. Herbert, R. Elsworth, and R. Telling. The continuous culture of bacteria; a theoretical and experimental study. *Journal of General Microbiology*, 14(3): 601–622, 1956.

71. B. Hernández-Bermejo and V. Fairén. Nonpolynomial vector fields under the lotka-volterra normal form. *Physics Letters A*, 206(1):31–37, 1995.

72. C. S. Holling. Resilience and stability of ecological systems. *Annual Review of Ecology and Systematics*, pages 1–23, 1973.

73. C. S. Holling. Engineering versus ecological resilience. In P. Schulze, editor, *Engineering with Ecological Constraints*, pages 31–44. National Academy Press, Washington D.C., 1996.

74. W. Horsthemke. *Noise Induced Transitions*. Springer, 1984.

75. R. Hoshaw, J. Rosowski, and J. Stein. Handbook of phycological methods: Culture methods and growth measurements. *Cambridge University Press, Cambridge, England*, page 53, 1973.

76. A. R. Hughes, B. D. Inouye, M. T. Johnson, N. Underwood, and M. Vellend. Ecological consequences of genetic diversity. *Ecology letters*, 11(6):609–623, 2008.

77. T. P. Hughes, M. J. Rodrigues, D. R. Bellwood, D. Ceccarelli, O. Hoegh-Guldberg, L. McCook, N. Moltschaniwskyj, M. S. Pratchett, R. S. Steneck, and B. Willis. Phase shifts, herbivory, and the resilience of coral reefs to climate change. *Current Biology*, 17(4):360–365, 2007.

78. M. Iwen. Lecture notes: Introduction to semidefinite programming. *Michigan State University*, Fall 2007.

79. M. D. Johnston and D. Siegel. Linear conjugacy of chemical reaction networks. *Journal of Mathematical Chemistry*, 49(7):1263–1282, 2011.

80. M. D. Johnston, D. Siegel, and G. Szederkényi. Computing weakly reversible linearly conjugate chemical reaction networks with minimal deficiency. *Mathematical Biosciences*, 2012.

81. M. D. Johnston, D. Siegel, and G. Szederkényi. A linear programming approach to weak reversibility and linear conjugacy of chemical reaction networks. *Journal of Mathematical Chemistry*, 50(1):274–288, 2012.

82. D. D. Jones and C. J. Walters. Catastrophe theory and fisheries regulation. *Journal of the Fisheries Board of Canada*, 33(12):2829–2833, 1976.

83. E. H. Kerner. Universal formats for nonlinear ordinary differential systems. *Journal of Mathematical Physics*, 22(7):1366–1371, 1981.

84. H. K. Khalil and J. Grizzle. *Nonlinear Systems*, volume 3. Prentice hall New Jersey, 1996.

85. R. Khasminskii. *Stochastic Stability of Differential Equations*, volume 66. Springer Science & Business Media, 2011.

86. R. Kindermann, J. L. Snell, et al. *Markov Random Fields and their Applications*, volume 1.

87. S. Klamt, J. Saez-Rodriguez, and E. D. Gilles. Structural & functional analysis of cellular networks with cellnetanalyzer. *BMC Systems Biology*, 1(1):2, 2007.

88. G. Kremer and D. Thompson. A bifurcation-based procedure for designing and analysing robustly stable non-linear hydraulic servo systems. *Proceedings of the IMechE, Part I: Journal of Systems and Control Engineering*, 212(5):383–394, 1998.

89. D. P. Kroese, T. Taimre, and Z. I. Botev. *Handbook of Monte Carlo Methods*, volume 706. John Wiley & Sons, 2011.

90. H. Kushner. Finite time stochastic stability and the analysis of tracking systems. *Automatic Control, IEEE Transactions on*, 11(2):219–227, 1966.

91. H. Kushner. *Stochastic Stability and Control, volume 33 of Mathematics in Science and Engineering.* Academic Press, New York, 1967.

92. Y. A. Kuznetsov. *Elements of Applied Bifurcation Theory.* Springer-Verlag New York, Inc., 1998.

93. J. B. Lasserre. Global optimization with polynomials and the problem of moments. *SIAM Journal on Optimization*, 11(3):796–817, 2001.

94. J. B. Lasserre. Semidefinite programming vs. lp relaxations for polynomial programming. *Mathematics of Operations Research*, 27(2):347–360, 2002.

95. M. Laurent. Sums of squares, moment matrices and optimization over polynomials. In *Emerging applications of algebraic geometry*, pages 157–270. Springer, 2009.

96. D. Lazard. A new method for solving algebraic systems of positive dimension. *Discrete Applied Mathematics*, 33(1):147–160, 1991.

97. D. Lazard. Solving zero-dimensional algebraic systems. *Journal of Symbolic Computation*, 13(2):117–131, 1992.

98. M. D. Lemmon and T. A. Tamba. Using elementary flux mode to estimate the distance to regime shifts in kinetic systems. In *Proceedings of the IFAC Conference on Analysis and Design of Hybrid Systems*, Atlanta, GA, 2015.

99. R. C. Lewontin. The meaning of stability. In *Brookhaven Symposia in Biology*, volume 22, pages 13–24, 1968.

100. J. Löfberg. Yalmip : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004. URL `http://users.isy.liu.se/johanl/yalmip`.

101. J. Lu, H. W. Engl, P. Schuster, et al. Inverse bifurcation analysis: application to simple gene systems. *Algorithms Mol Biol*, 1(11), 2006.

102. D. Ludwig, D. D. Jones, and C. S. Holling. Qualitative analysis of insect outbreak systems: the spruce budworm and forest. *The Journal of Animal Ecology*, pages 315–332, 1978.

103. K. Martin and R. Lorenzo. *Computational Commutative Algebra 1.* Springer-Verlag, 2000.

104. R. M. May. Thresholds and breakpoints in ecosystems with a multiplicity of stable states. *Nature*, 269(5628):471–477, 1977.

105. M. M. Maza. On triangular decompositions of algebraic varieties. In *Proceedings of the MEGA-2000 Conference*, Bath, UK, 2000.

106. A. P. Mazzoleni and I. Dobson. Closest bifurcation analysis and robust stability design of flexible satellites. *Journal of Guidance, Control, and Dynamics*, 18 (2):333–339, 1995.

107. A. N. Michel. Stability analysis of stochastic composite systems. *Automatic Control, IEEE Transactions on*, 20(2):246–250, 1975.

108. A. N. Michel. Stability analysis of stochastic large-scale systems. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, 55(2):113–123, 1975.

109. H. M. Möller. On decomposing systems of polynomial equations with finitely many solutions. *Applicable Algebra in Engineering, Communication and Computing*, 4(4):217–230, 1993.

110. M. Mönnigmann and W. Marquardt. Normal vectors on manifolds of critical points for parametric robustness of equilibrium solutions of ode systems. *Journal of Nonlinear Science*, 12(2):85–112, 2002.

111. A. Mosek. The mosek optimization software. *Online at http://www. mosek. com*, 54, 2010.

112. I. Noy-Meir. Stability of grazing systems: an application of predator-prey graphs. *The Journal of Ecology*, pages 459–481, 1975.

113. M. Nyström, C. Folke, and F. Moberg. Coral reef disturbance and resilience in a human-dominated environment. *Trends in Ecology & Evolution*, 15(10): 413–417, 2000.

114. T. Oguz, J. W. Dippner, and Z. Kaymaz. Climatic regulation of the black sea hydro-meteorological and ecological properties at interannual-to-decadal time scales. *Journal of Marine Systems*, 60(3):235–254, 2006.

115. B. K. Øksendal and A. Sulem. *Applied Stochastic Control of Jump Diffusions*, volume 498. Springer.

116. M. Pal, A. K. Pal, S. Ghosh, and I. Bose. Early signatures of regime shifts in gene expression dynamics. *Physical Biology*, 10(3):036010, 2013.

117. P. A. Parrilo. *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*. PhD thesis, California Institute of Technology, Pasadena, CA, 2000.

118. P. A. Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Mathematical Programming*, 96(2):293–320, 2003.

119. P. A. Parrilo and B. Sturmfels. Minimizing polynomial functions. *Algorithmic and quantitative real algebraic geometry, DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 60:83–99, 2003.

120. S. Prajna, A. Papachristodoulou, and P. Parrilo. Introducing sostools: A general purpose sum of squares programming solver. In *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, volume 1, pages 741–746. IEEE, 2002.

121. S. Prajna, A. Papachristodoulou, P. Seiler, and P. A. Parrilo. *SOSTOOLS: Sum of squares optimization toolbox for MATLAB*, 2004.

122. S. Prajna, A. Jadbabaie, and G. J. Pappas. A framework for worst-case and stochastic safety verification using barrier certificates. *Automatic Control, IEEE Transactions on*, 52(8):1415–1428, 2007.

123. W. H. Press. *Numerical Recipes: The Art of Scientific Computing, 3rd Ed.)*. Cambridge university press, 2007.

124. P. E. Protter. *Stochastic Integration and Differential Equations*, volume 21. Springer, 2004.

125. S. Redner. *A Guide to First-Passage Processes*. Cambridge University Press, 2001.

126. J. Rocha, J. Yletyinen, R. Biggs, T. Blenckner, and G. Peterson. Marine regime shifts: drivers and impacts on ecosystems services. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 370(1659):20130273, 2015.

127. J. Rockström, W. Steffen, K. Noone, Å. Persson, F. S. Chapin, E. F. Lambin, T. M. Lenton, M. Scheffer, C. Folke, H. J. Schellnhuber, et al. A safe operating space for humanity. *Nature*, 461(7263):472–475, 2009.

128. L. C. G. Rogers and D. Williams. *Diffusions, Markov Processes and Martingales: Volume 2, Itô calculus*, volume 2. Cambridge University Press, 2000.

129. M. Scheffer. *Ecology of Shallow Lakes*. Springer, 2004.

130. M. Scheffer and S. R. Carpenter. Catastrophic regime shifts in ecosystems: linking theory to observation. *Trends in Ecology & Evolution*, 18(12):648–656, 2003.

131. M. Scheffer, S. Hosper, M. Meijer, B. Moss, and E. Jeppesen. Alternative equilibria in shallow lakes. *Trends in Ecology & Evolution*, 8(8):275–279, 1993.

132. M. Scheffer, S. Carpenter, J. A. Foley, C. Folke, B. Walker, et al. Catastrophic shifts in ecosystems. *Nature*, 413(6856):591–596, 2001.

133. C. H. Schilling, D. Letscher, and B. Ø. Palsson. Theory for the systemic definition of metabolic pathways and their use in interpreting metabolic function from a pathway-oriented perspective. *Journal of Theoretical Biology*, 203(3): 229–248, 2000.

134. P. Seiler. Sosopt: A toolbox for polynomial optimization. *arXiv:1308.1889*, 2013.

135. K. W. Shertzer, S. P. Ellner, G. F. Fussmann, and N. G. Hairston. Predator–prey cycles in an aquatic microcosm: testing hypotheses of mechanism. *Journal of Animal Ecology*, 71(5):802–815, 2002.

136. H. L. Smith. *The Theory of the Chemostat: Dynamics of Microbial Competition*, volume 13. Cambridge University Press, 1995.

137. G. Stengle. A nullstellensatz and a positivstellensatz in semialgebraic geometry. *Mathematische Annalen*, 207(2):87–97, 1974.

138. M. H. Stone. The generalized weierstrass approximation theorem. *Mathematics Magazine*, 21(5):237–254, 1948.

139. J. F. Sturm. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11(1-4):625–653, 1999.

140. B. Sturmfels. Grobner bases of toric varieties. *Tohoku Mathematical Journal, Second Series*, 43(2):249–261, 1991.

141. B. Sturmfels. *Gröbner Bases and Convex Polytopes*, volume 8. American Mathematical Society, 1996.

142. B. Sturmfels. *Solving Systems of Polynomial Equations*, volume 97. American Mathematical Society, 2002.

143. G. Szederkényi. Computing sparse and dense realizations of reaction kinetic systems. *Journal of Mathematical Chemistry*, 47(2):551–568, 2010.

144. G. Szederkényi. Computing reaction kinetic realizations of positive nonlinear systems using mixed integer programming. In *8th IFAC Symposium on Nonlinear Control Systems-NOLCOS*, pages 1–6, 2010.

145. G. Szederkényi and K. M. Hangos. Finding complex balanced and detailed balanced realizations of chemical reaction networks. *Journal of Mathematical Chemistry*, 49(6):1163–1179, 2011.

146. T. A. Tamba and M. D. Lemmon. Certificate-based method for bifurcation and robustness analyses of nonnegative dynamical systems. *Technical Report, University of Notre Dame*, 2014.

147. T. A. Tamba and M. D. Lemmon. *C-BRANDS: A MATLAB tool for Certificate-based bifurcation and Robustness analyses of nonnegative dynamical systems*, 2014.

148. P. Tankov. *Financial Modelling with Jump Processes*. CRC Press, 2004.

149. M. Terzer and J. Stelling. Large-scale computation of elementary flux modes with bit pattern trees. *Bioinformatics*, 24(19):2229–2235, 2008.

150. U. Topcu and A. Packard. Local stability analysis for uncertain nonlinear systems. *Automatic Control, IEEE Transactions on*, 54(5):1042–1047, 2009.

151. U. Topcu, A. Packard, and P. Seiler. Local stability analysis using simulations and sum-of-squares programming. *Automatica*, 44(10):2669–2675, 2008.

152. A. Voinov and Y. M. Svirezhev. A minimal model of eutrophication in freshwater ecosystems. *Ecological modelling*, 23(4):277–292, 1984.

153. A. I. Vol'pert. Differential equations on graphs. *Mathematics of the USSR-Sbornik*, 17(4):571, 1972.

154. S. Waldherr and F. Allgöwer. Robust stability and instability of biochemical networks with parametric uncertainty. *Automatica*, 47(6):1139–1146, 2011.

155. T.-C. Wang, S. Lall, and M. West. Polynomial level-set method for attractor estimation. *Journal of the Franklin Institute*, 349(9):2783–2798, 2012.

156. H. Wolkowicz, R. Saigal, and L. Vandenberghe. *Handbook of Semidefinite Programming: Theory, Algorithms and Applications*, volume 27. Springer Science & Business Media, 2012.

157. T. Yoshida, L. E. Jones, S. P. Ellner, G. F. Fussmann, and N. G. Hairston. Rapid evolution drives ecological dynamics in a predator–prey system. *Nature*, 424(6946):303–306, 2003.

158. Y. Zhang. Solving large-scale linear programs by interior-point methods under the matlab environment. *Optimization Methods and Software*, 10(1):1–31, 1998.