ANALYSIS AND PREDICTION OF GENE EXPRESSION PATTERNS BY DYNAMICAL SYSTEMS, AND BY A COMBINATORIAL ALGORITHM

MESUT TAŞTAN

AUGUST 2005

ANALYSIS AND PREDICTION OF GENE EXPRESSION PATTERNS BY DYNAMICAL SYSTEMS, AND BY A COMBINATORIAL ALGORITHM

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF APPLIED MATHEMATICS

OF

THE MIDDLE EAST TECHNICAL UNIVERSITY

BY

MESUT TAŞTAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

THE DEPARTMENT OF SCIENTIFIC COMPUTING

AUGUST 2005

Approval of the Graduate School of Applied Mathematics

Prof. Dr. Ersan AKYILDIZ
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. Bülent KARASÖZEN
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Prof. Dr. Gerhard Wilhelm WEBER
Supervisor

Examining Committee Members

Prof. Dr. Bülent Karasözen

Prof. Dr. Gerhard Wilhelm Weber

Assoc. Prof. Dr. Tanıl Ergenç

Assist. Prof. Erkan Mumcuoğlu

Assist. Prof. Hakan Öktem

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: Mesut Taştan

Signature:

Abstract

ANALYSIS AND PREDICTION OF GENE EXPRESSION PATTERNS BY DYNAMICAL SYSTEMS, AND BY A COMBINATORIAL ALGORITHM

Mesut Taştan
M.Sc., Department of Scientific Computing
Supervisor: Prof. Dr. Gerhard Wilhelm WEBER
Co-Advisor: Prof. Dr. Bülent KARASÖZEN

August 2005, 75 pages

Modeling and prediction of gene-expression patterns has an important place in computational biology and bioinformatics. The measure of gene expression is determined from the genomic analysis at the mRNA level by means of *microarray* technologies. Thus, mRNA analysis informs us not only about genetic viewpoints of an organism but also about the dynamic changes in environment of that organism. Different mathematical methods have been developed for analyzing experimental data. In this study, we discuss the modeling approaches and the reasons why we concentrate on models derived from *differential equations* and improve the pioneering works in this field by including affine terms on the right-hand side of the nonlinear differential equations and by using Runge-Kutta instead of Euler discretization, especially, with Heun's method. Herewith, for stability analysis we apply modified Brayton and Tong algorithm to time-discrete dynamics in an extended space.

Keywords: Computational Biology, Gene-Expression Data, Mathematical Modeling, Prediction, Dynamical System, Runge-Kutta Discretization, Stability.

Öz

GEN MOTİLERİNİN DİNAMİK SİSTEMLER VE KOMBİNATORİK BİR ALGORİTMA İLE ANALIZLERİN YAPILMASI VE GELECEK TAHMİNLERİ

Mesut Taştan
Yüksek Lisans, Bilimsel Hesaplama Bölümü
Tez Yöneticisi: Prof. Dr. Gerhard Wilhelm Weber
Eş Danışman: Prof. Dr. Bülent KARASÖZEN

Ağustos 2005, 75 sayfa

Gen motiflerinin modellenmesi ve buna dayali tahminler, hesaplamalı biyoloji ve biyoinformatik alanlarinda çok önemli bir yer tutmaktadırlar. Gen aktivite değişimleri, mRNA değerlerinin mikrodizin teknolojisi sayesinde ölçülmesi ile anlaşılır. Deney verilerinin analizi için değişik matematiksel modeller geliştirilmiştir. Bu tezde, modelleme yaklaşımları incelenmiş ve gen motiflerinin değişimleri bayağı diferansiyel denklemler yardımı ile modellenmiş olup bunun sebepleri açıklanmıştır. Bu konuda daha önce yapılmış, sürekli modellere doğrusal olmayan yer değiştirme terimleri eklenerek yeni bir model geliştirilmiş ve Euler kesintileme metodu yerine Runga-Kutta metodu kullanılmıştır. Bunların yanında zaman kesintili modelin kararlılık analizi, genişletilmiş uzay içerisinde Brayton-Tong Algoritmasının değiştirilmiş hali ile yapılmıştır.

Anahtar Kelimeler: Biyolojik Hesaplama, Gen Düzenleme Verisi, Matematiksel Modelleme, Dinamik Sistemler, Runga-Kutta Kesintilemesi, Kararlılık.

To my family

Acknowledgements

My MS thesis might not have gotten finished without some support and encouragement of some key people. First and foremost, I would like to thank my advisor, Prof. Gerhard Wilhelm Weber. My friends and I greatly appreciate his close friendship and open personality, and the impulse he gave me. He showed me what it takes to lead a successful, vibrant, and fun research group. I feel truly lucky and privileged to have been his student.

I would like to thank to Prof. Bülent Karasözen for providing me with excellent reviews. Our institute has a big chances of having a sharp-minded member Assist. Prof. Hakan Öktem. I always benefitted from his answers to my questions and from his lectures. I am also indebted to Prof. Tanıl Ergenç for his important remarks and valuable discussions.

No doubt this study would have looked very different if Elena Bechmann, another student of Prof. Weber, had not provide me with her thesis about the algorithmic part. Thus, I thank to Elena and Prof. Pickl for this essential collaboration.

The environment of Institute of Applied Mathematics (IAM) itself has an important place in this study. Many thanks to Süreyya, Barış, Yeliz, Oktay, Ayşegül and especially to Derya for careful proofreading, comments and corrections. I am proud to call these people my friends.

For his great transportation performance and close friendship, I am grateful to my friend İlker Kılıç. A special thank needs to be given to Umut Orguner for the part in writing the algorithms.

Last - but not least - my thanks to Ömür Yılmaz for being in my life and especially for her patience in these couple of months. I acknowledge, appreciate, and return the love and support of my family.

Table of Contents

CHAPTER

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Introduction

Genome research is one of the most exciting scientific disciplines of the 21st century. The genome of an organism is perhaps the most important topic of biology but yet, in general, it is the least understood [59].

An organism is shaped by its genome by mapping the primary structure of nucleic acid molecules like *DNA* and *RNA* to a primary structure of protein molecule. During such a mapping, proteins are constructed in the process called *gene expression* which constitutes the essence of a cell. The answer of the question which and how genes are expressed, under which conditions and where, can give clues for understanding the functions of genes. The regulation of gene expression is specified by the interactions between DNA, RNA, proteins and small molecules [30]. This complex of interactions is called a *gene regulatory network*.

While the study of expression patterns is vital for understanding the physiological features, the characterization of most known genes is incomplete [53]. Thus, computational methods like networking and microarrays for identifying and understanding the dynamics behind the gene expression are found to be most efficient ways. Moreover, they engage the interest in gene regulatory systems and in bioinformatics. An advantage of these technologies is that they allow the profiling of thousands of genes in a single experiment.

1.2 DNA Experiments

It is assumed those huge amount of genes and their products (i.e., RNA and proteins) create the *mystery of life*. However, traditional methods in molecular biology generally work on a ”one gene in one experiment” basis, which means that the throughput is very limited and the ”whole picture” of a gene function is hard to obtain. Recently, a new technology, called a *DNA microarray*, has attracted a tremendous interest among biologists. This technology promises to monitor the whole genome on a single chip so that researchers can have a better picture of the interactions among thousands of genes simultaneously.

Microarray experiments provide us with a huge amount of data to be analyzed. Recently, many databases of experimental results are available for researchers but the evaluation of these results for representing the complex patterns of genes interactions merges with a big scientific challenge with high industrial pay-offs [30].

GENE \ time	0	9.5	11.5	13.5	15.5	18.5	20.5
’YHR007C’	0.224	0.367	0.312	0.014	-0.003	-1.357	-0.811
’YAL051W’	0.002	0.634	0.31	0.441	0.458	-0.136	0.275
’YAL054C’	-1.07	-0.51	-0.22	-0.012	-0.215	1.741	4.239
’YAL056W’	0.09	0.884	0.165	0.199	0.034	0.148	0.935
’PRS316’	-0.046	0.635	0.194	0.291	0.271	0.488	0.533
’KAN-MX’	0.162	0.159	0.609	0.481	0.447	1.541	1.449
’E. COLI 10’	-0.013	0.88	-0.009	0.144	-0.001	0.14	0.192
’E. COLI 33’	-0.405	0.853	-0.259	-0.124	-1.181	0.095	0.027

Table 1.1: A portion of row gene expression data for *Yeast* cells from Stanford microarray database [52].

Base-pairing (i.e., A-T and G-C for DNA; A-U and G-C for RNA) or *hybridization* is the underlining principle of DNA microarrays. An array, an orderly arrangement of samples, enables a medium for matching known and unknown DNA samples based on base-pairing rules and automating the process of identifying the unknowns. These experiments use microplates or standard blotting membranes, and can be created by hand or make use of robotics to deposit the sample. In general, arrays are described as *macroarrays* or *microarrays*, the difference lies in the size of the sample spots.

Main applications of DNA microarray technologies can be listed as gene discovery, disease diagnosis, drug discovery and toxicological researches.

1.3 Mathematical Strategies on Experimental Data

Since the 1950s, a variety of mathematical identifications have been proposed. In 1952, *Turing* has firstly introduced the idea of a mathematical model for biological systems [12, 57, 63]. In according to this approach, the change of the state of the cell is equal to the sum of all acting forces on that cell. This basic idea is one of the foundations for regulatory systems, but the development of experiments at molecular levels requires extended and computer supported models.

Easily accessible data through databases make modeling techniques popular. Based on these experimental data it is aimed to make reliable future predictions and simulations and to find the correlation between genes. For example, from the data above we try to analyze and compare the approximations with real data (cf. Figure 1.1).

The modeling approaches which we try to summarize and explain in this study are *Bayesian networks*, *Boolean networks*, models derived from *ordinary or piece-wise linear differential equations* and *hybrid system* modeling. All these methods have both advantages and disadvantages [20, 30] concerning goodness of data fit, computation time, capturing dynamics well, stability and other qualitative or quantitative aspects.

1.4 About This Thesis

Our work is an interdisciplinary specialization of scientific areas including molecular biology, discrete mathematics, computational technologies, dynamical systems, optimization theory, statistical analysis and interpretation of biological data.

We begin with briefly presenting the biological background for genes and gene expression, then we summarize the experimental design (cf. Chapter 2). We do this because of the need to point out our study's importance and possible applications in medicine, computational biology, bioinformatics, medicine and pharmacy.

In Chapter 3, we discuss the modeling approaches and the reasons why we concentrate on models derived from *differential equations*. Firstly, their more detailed representation of regulatory interactions can provide a more accurate understanding of the physical systems. Secondly, there is a large body of dynamical systems theory that can be used to analyze such models. Thirdly, concerning that biological systems evolve in continuous time, we prefer to use the systems of differential equations.

The method of data analysis should be selected with careful consideration for the experimental setup and the underlying physics (if they are known). We are particularly interested in how gene expression patterns change over time. In this module, we look at two different kinds of models, namely, as time-continuous models and time-discrete models which are examples of dynamical systems. Our work improves the time-continuous model introduced and developed in the pioneering works in this field [8, 13, 15, 47, 63] by additively including an affine shift term to the systems of nonlinear differential equations. Furthermore, we also improve the time-discrete model by using different numerical methods (cf. Chapter 4). These procedures apply in an extended dimension $2n$ instead of n.

When a *dynamical* model is introduced, then the questions concerning stability, parameter estimation, parameter sensitivity analysis and other qualitative or quantitative aspects arise. Parameter estimation requires statistical learning methods and optimization [25]. Such a kind of work, up to some extent, is done in [16, 63].

Stability is an important analysis criterion and item for systems of differential equations. Staying of a system near an equilibrium can be interpreted as both a controllability and coming of a disease to a rest, but also a lack of flexibility to adopt to new environmental condition. A system is *(Lyapunov) stable* if all states will remain bounded for all time, for any finite initial condition. Since gene expression patterns lie in a bounded regions, in Chapter 5, we perform the stability analysis for goodness of fit test of our generalized model. This will be done in the extended space, and with the help of the algorithmic method introduced by *Brayton and Tong* [5]. The idea of applying this combinatorial algorithm to gene dynamics was firstly mentioned by *Gebert, Lätsch, Pickl, Weber and Wünschiers* in the paper [15] which exploits the time-discrete dynamics by matrix multiplication and the extremal points of polyhedral regions. Moreover, it relates the geometry of polytopes with the theory of dynamical systems. To explain and illusrate the method better, similar to the one studied in [4], we explain the algorithm step by step and present an example in Chapter 6.

Our model provides an inference of a wide range of gene networks, and in terms of stability and predictions. We believe that given reliable data and high capability of computational efforts, our study can serve for identifying and anticipating gene expression patterns for the future. Herewith, we aim at a mathematical contribution to biological and medical progress, and to health care.

Chapter 2

Biological Background

2.1 Life Molecules

All living systems from single-celled to multicellular plants and animals store, replicate and transmit the information to next offsprings. *Genetics* is the study and the science of heredity which includes the information passing from generation, in fact parent, to generation, the progeny. Without a knowledge of genetics, a true understanding of biology cannot be obtained or appreciated [3]. *Cell*, the minimal unit of the life, contains many macromolecules that organize and coordinate all of the events. This wide range of molecules can be written as water, *Deoxyribonucleic acid* (DNA), proteins, polysaccharide, and small organic compounds like acids and sugars. In the following subsections, we give a rough summary about the elementary molecules of genetics.

2.1.1 DNA and Genes: The Heredity Materials

Some macromolecules have other vital properties as controlling and governing of all activities. Here, the DNA molecule is the chief of this complex structure because it stores the information about the structure of macromolecules and helps for production of them with respect to the cell's needs. A DNA molecule forms the "blueprint" of an organism by encoding the information [37]. Several lines of DNA contain genetic material of living organisms within the units known as nucleotides and in each such molecules, the sequence of four different bases: *adenine* (A), *guanine* (G), *thymine* (T) and *cytosine* (C) specify the type and synthesis timing of proteins. These base pairs stand as A-T and G-C in the double helix structure deduced by J. D. Watson and F. H. C. Crick in 1953 (see Figure 2.1).

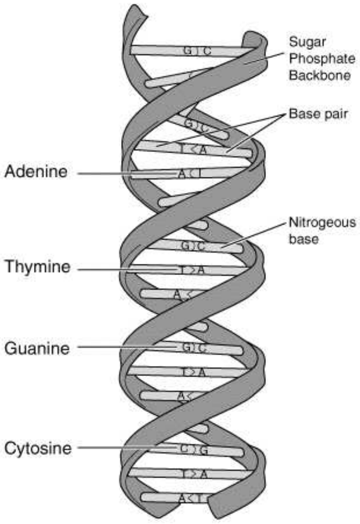

Figure 2.1: DNA structure and bases (taken from National Human Genome Research Institute, Division of Intramural Research) [43].

An important character of the DNA model is that the two strands are held together by noncovalent *hydrogen bonds*, weak electrostatic cord between two atoms that can be easily broken and reformed. The number of bonds varies in base-pairs so that there are two between A and T whereas there are three bonds between G and C. The meaningful parts, units, of the DNA sequence are *genes* that control the identifiable hereditary traits of an organism. In fact, a gene can be defined as a segment of DNA determining a functional RNA. Here, we say "meaningful parts" because only some regions of DNA and genes are encoding segments (Figure 2.2). In humans, the proportion of protein-coding region is less than 5% [23].

The total set of genes involved in an individual or in a cell is called its *genome*. What genomics tries to define is the *genotype*, the genetic structure of the organisms. The *phenotype*, however, tries to make the list of features expressed under a particular set of environmental factors. The phenotypic features (appearance) may or may not directly reflect the genotype (the present

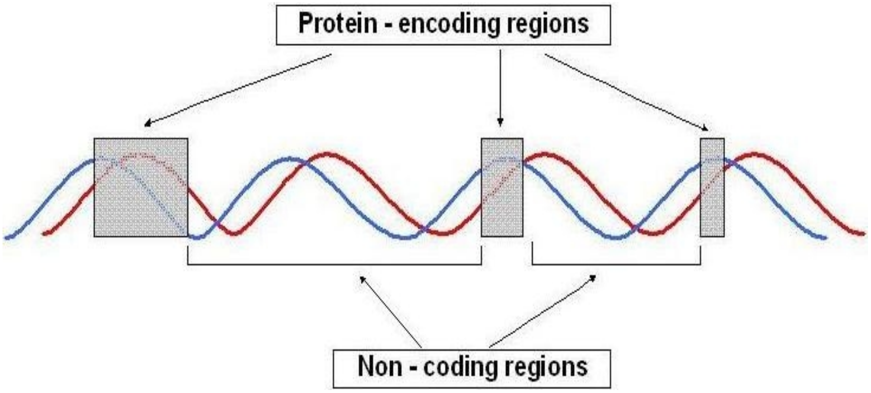

Figure 2.2: Coding Regions for DNAs and genes [23].

genes). Genetic material controls both phenotypic and genotypic characters. According to the definition done by *Mendel* a century ago [38], a gene is a unit element that satisfies two basic natures, namely:

- a "particulate factor" that must be capable of storing genetic information and that passes unchanged from parents to progeny,
- an object that may exist in different, alternative forms as alleles which may produce the differences.

The DNA molecules are contained in *chromosomes* composed of two kinds of large organic molecules called *proteins* and *nucleic acids*. Chromosomes also involve regulatory elements and other intervening nucleotide sequences.

2.2 Central Dogma of Biology

There is an information flow from DNA with *Ribonucleic acid* (RNA) to proteins. The intermediary element RNA has similar structure to DNA molecules. The differences stay, firstly, in sugar that instead of deoxyribose, RNA, involves *ribose* and, secondly, in base pairs such that instead of thymine (T) RNA contains *uracil* (U) and, thirdly, in strand property usually RNA is single-stranded. This route of transfer mentioned above defines *central dogma* of biology which involves the transfer of genetic information from DNA to RNA called *transcription* and the passing of information from RNA to proteins called *translation* (Figure 2.3). The idea which is coming from the *central dogma* and stating that

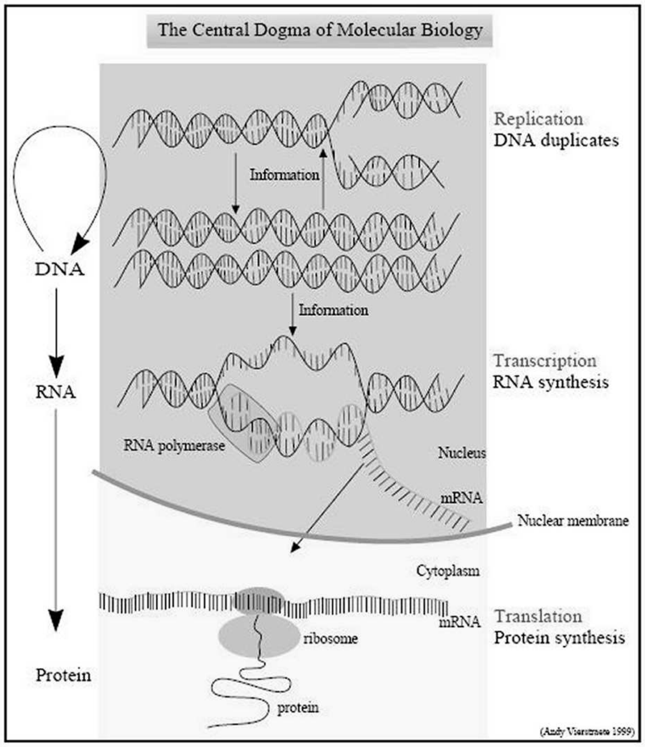

Figure 2.3: Central Dogma of Biology [61].

one gene is responsible for one protein molecule has recently been changed. We are faced with a more complex dynamics than it was imagined. One gene has may interactions with many particles in a cell and can lead to the production of more protein molecules. However, with all this complexity, the map of relationships between genes still seems to be constructible because of limited interaction of genes stemming from their limited topological structures [49].

2.2.1 Transcription

In most cases, proteins are synthesized in the cytoplasm, but the organisms like eukaryotes have their DNA and chromosomes in the nuclei of cells. RNA, therefore, is used as a template for synthesis. This indirect effect of DNA over RNA constitutes the main concept of the central dogma [24]. One strand of DNA called the *sense* or *noncoding* strand is the template for encoding information. This process begins with binding of RNA polymerase to *promotors*, which are the special regions of the noncoding strand near a gene where transcription starts. RNA polymerase causes hydrogen bounds to break and the helix strands to unwind. Then, for an opened single-pair, the complementary base-pairs are again tied by hydrogen bonds and construct the *messenger* RNA (*mRNA*) (for details please see [19]). After RNA polymerase reaches the termination region on the strand, the process ends and the DNA turns to the original form. The molecule mRNA is one the three types of RNA taking part in the protein synthesis. In most molecules, a high proportion of the RNA nucleotides are used for encoding. For this reason, in experiments usually mRNA levels are measured to determine future predictions.

2.2.2 Translation

Translation is the second process in protein synthesis in which mRNA carries the genetic information from the chromosomes to the ribosomes, a cell structure containing *ribosomal* RNA (*rRNA*) and protein base pairs. Similar to transcription, translation has also three parts, namely, *initiation, base-paring* and *termination.* Nucleotide sequences are transferred into triplets and each of those triplets is linked to a specific amino acid. Each three mRNA nucleotides are called as *codons*, and the corresponding ones in tRNA are called its *anticodons.* In ribosomes, the codon AUG which specifies the amino acid methionine initiates the polypeptide synthesis and, through hydrogen bonds, the base pairs on mRNA are matched to the ones on *transfer* RNA (*tRNA*). Here, tRNA is a small single-stranded RNA molecule to the length 74-95 nucleotids [23]. Each tRNA carries only one amino acid. As we see in Figure 2.4, when tRNA finds its places in the mRNA line, the attached amino acid becomes the new terminal end of the growing polypeptide chain. The codons UAA, UAG and UGA terminate the translation process. Transcription and translation processes result in proteins which are the most functional life molecules because of catalyzing a wide variety of chemical reactions and also serving as building blocks for cellular

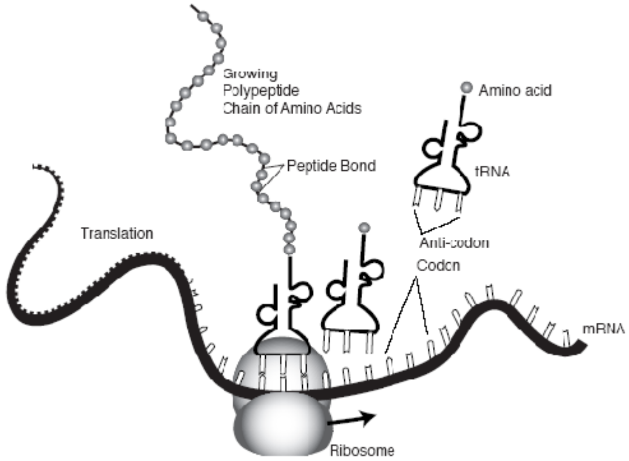

Figure 2.4: Translation in the cytoplasm: tRNA carries amino acids which are added to the growing peptide chain in the ribosome (taken from National Human Genome Research Institute, Division of Intramural Research) [43].

structures like muscles, skin, enzymes, etc.. The tissue, metabolic state of the cell, specifies the amount of protein that a cell expresses.

2.3 Gene Expression and Microarray Experiments

Gene (or *protein*) *expression* is the process by which gene information is converted for producing cell structures and cell functions. In the previous sections, we roughly summarized the beginning main events transcription and translation, taking part in this process. After transcription and translation, steps like *folding*, *post-translational modification* and *targeting* occur before protein product but we leave these details to [34]. Since mRNA is an exact copy of the DNA coding regions, mRNA analysis can be well used to explore the process in coding regions of DNA. More importantly, the measure of gene expression can be determined from the genomic analysis at the mRNA level [48]. Both genomic and environmental factors affect the gene expression levels. For example, the environmental factors including stress, light, temperature and other signals cause some changes in hormones and in enzymatic reactions that influence the gene expression level. That is why mRNA analysis informs us not only about genetic viewpoints of an organism but also about the *dynamic changes* in environment of that organism. For most genes, protein levels are defined by steady state mRNA levels [62]. Thus, quantitative expressions at mRNA level provide important clues about the underlying dynamics. Peculiar changes in monitoring mRNA levels generally refer to drug treatment, shocks, disease or metabolic states. Such kinds of perturbation are also concerned as *genomic instability* which we include in this study.

2.4 Microarray Technology

Microarray technology is an array-based technology that monitors thousands of different RNA molecules simultaneously revealing their expression patterns and perturbed subsequent cellular pathways. One of the most frequently used microarray applications [7] is to compare gene expression levels of the same cell type like healthy cell and diseased cell under two different conditions. Such application can give vital information on the reasons of diseases. Recently, expression analysis is the main large-scale application of microarrays and it is followed by DNA variation on a genome-wide scale [6]. Both of these applications share similar requirements, but they differ in some crucial aspects that have resulted in two different types of microarrays. In microarray technology, gene expression profiling or gathering the data is mainly done by either *cDNA* arrays or *oligonucleotide* arrays.

2.4.1 Oligonucleotide Arrays and Implementation

Small parts of DNA molecules up to 25 nucleotides called as *oligonucleotides* that can be created by reverse transcription of expressed mRNA levels in a cell type. Basically, oligonucleotide arrays are composed of glass or silicon surfaces having hundreds to thousands of immobilized oligonucleotide array of detection units (*probes*) that permit the *hybridization*. We can summarize the implementation as follows: In oligonucleotide chips, a flourescently labeled nucleic acid sample is injected to the probe where hybridization occurs between sample and probe. Then, laser detection is applied at the interface of array surface through the back of the glass material. A lens gathers the fluorescence emission to pass it to a sensitive detector with the help of series of optical filters. After scanning by the laser, a two-dimensional fluorescence image of hybridization intensity is created.

2.4.2 cDNA Arrays and Implementation

Chemically or by *polymerase chain reactions* (*PCR*) synthesized form of the coding part of the DNA sequence complementary to its corresponding mRNA transcript is defined as *complementary* DNA (*cDNA*). The cDNA chips include a relatively large amount of nucleic acids, usually more than 100 nucleotides. They are generally used for the measurement of RNA expression levels and cheaper than the oligonucleotide chips. The implementation process can be described as:

1. samples are arrayed on the slides, air-dried and are immobilized by ultraviolet irradiation,
2. labeled cDNA samples are hybridized with probes after little mousturing,
3. heating and washing the arrays, unbounded cDNAs and solutions are eliminated,
4. after image processing techniques, gene expression levels are obtained.

In Figure 2.5, we can see a generalized scheme illustrating DNA microarray experiments.

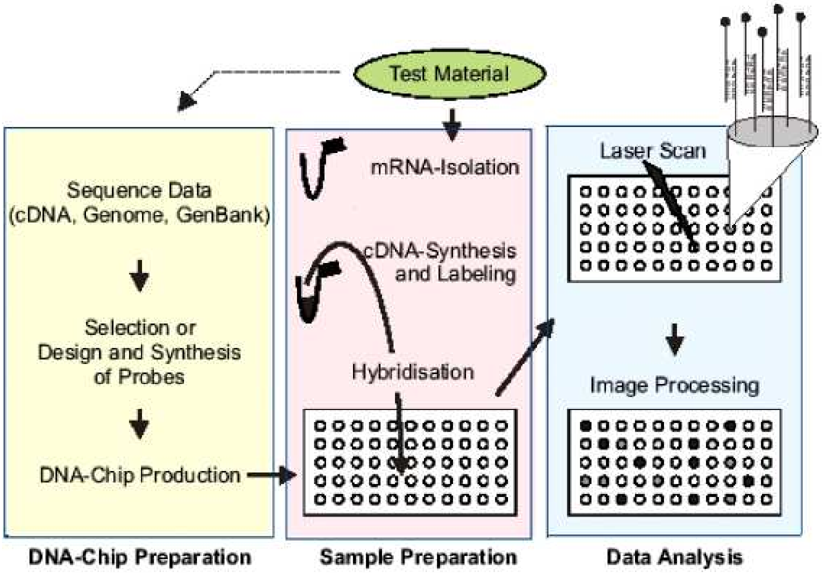

Figure 2.5: Design and implementation of DNA microarray experiments [15].

2.5 Evaluating the Expression Data

Microarray experiments can quickly monitor the expression values for large numbers of genes. The goal researchers have in mind is ultimately to clarify the precise connections of the *genetic network*: mathematically speaking, a graph consisting of vertices as genes and of edges connecting genes (please refer to Section 3.2). For each gene it is aimed to find which genes are how much influenced by this gene. Different mathematical methods have been developed for construction and analyzing such networks. Thus, our study may support these techniques to add new insights by means of mathematical modeling, dynamical systems and algorithms.

Chapter 3

Gene Regulatory Networks

3.1 Overview

Although recent development of microarray technology added new insights to the study of genetic regulatory systems, it still contains many challenges like the limited number of experimental data, noise, and quality of training data. Such obstacles make it difficult to understand the underlying dynamics of the regulatory network.

Besides the experimental design, the existence of lots of positive and negative feedbacks in genetic regulatory systems lets the intuitive understanding be hard. However, in the light of combined biochemical and genetics studies and user-friendly computer tools, a variety of mathematical models have been constructed to describe gene interactions and to make predictions in a systematic way [30].

Below in Figure 3.1, we express basic ideas and notions behind the modeling aspects in *mathematical biology*, *computational biology* and *bioinformatics*.

3.2 Modeling Approaches

3.2.1 Modeling by Graphs

Gene regulatory networks are generally modeled by considering them as a *directed graphs*.

Definition 3.1. *A directed graph* is a tuple $G = (V, E)$, where V is the set of

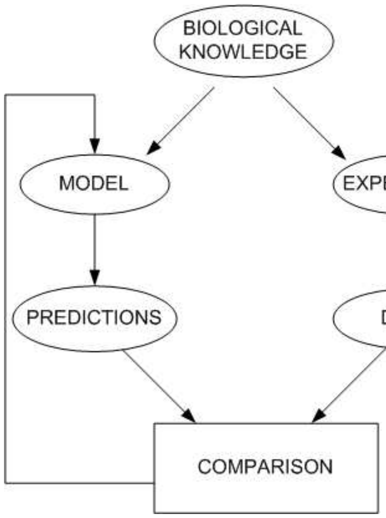

Figure 3.1: General modeling approach.

vertices (genes) and E is the set of *edges* (relationships). It is also possible to define the edges as directed in such that the tuple of vertices (v_i, v_j), sometimes in short: (i, j), with the head (v_i) and the tail (v_j) of the edge. These edges can carry some weight or they can be labeled. For example, the activation or inhibition of i^{th} gene by j^{th} gene can be represented as $(i, j, +)$ or $(i, j, -)$, respectively.

A simple directed graph representation of a regulatory network of three genes is shown in Figure 3.2.

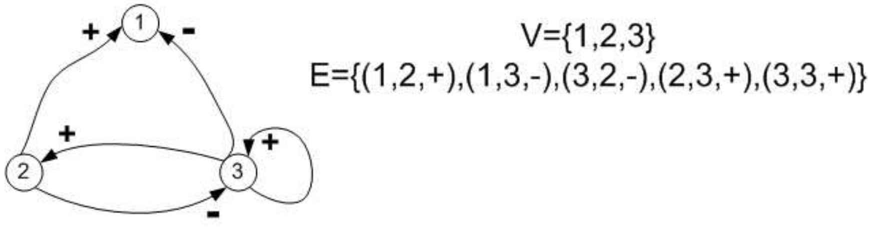

Figure 3.2: A graph representation of a simple regulatory network [30].

Biological features can be modeled by or learned from the operations on the regulatory networks. These operations such as pathways between two predefined genes, cycles, connectivity, etc., give us some clues about missing regulatory relations, connectivity, structures (components and clusters) and about redundancy in the network.

3.2.2 Bayesian Networks

Formally, Bayesian networks are described in terms of probabilities and conditional independence statements such that a relation can be noted between this characterization and the notion of direct causal influence [27, 51].

They are promising tools in network modeling because, firstly, statistical learning and simulation techniques of Bayesian networks from experiments and computational algorithms are well developed and have been used successfully in many applications [25]. Secondly, some *noise* naturally involved in measurements makes the Bayesian network approach attractive, because in those networks noise is taken into account [30]. Thirdly, they are useful for deducing events composed of locally interacting components, i.e., one can imagine or decompose the whole gene network in to small ones.

In [14], a genetic regulatory system is represented by a directed acyclic graph $G = (V, E)$ in which for each gene i, represented by the i^{th} vertex, there is an associated random variable X_i, and for each X_i a conditional distribution $p(X_i|parents(X_i))$ is defined by *Friedman, Linial, Nachman and Pe'er* [14]. Here, $parents(X_i)$ indicates the elements that regulate the i^{th} gene in a direct way. Such parents together with the graph define the unique joint probability distribution $p(X)$ by means of *Markov assumption* [30] in the form of

$$p(X_1, X_2, ..., X_n) = \prod_{i=1}^{n} p(X_i|parents(X_i)).$$

Having the experimental data and related set of independent values for $X = \{X_1, X_2, ..., X_n\}$, one can make a characterization of a network by Bayesian learning techniques, which, in essence, is based on a matching score to determine the network G with given training data D. Evaluating each network with respect to the training data needs a *scoring function.* A commonly used scoring function [14] is

$$Score(G : D) \ := \ \log P(G|D) = \log P(D|G) + \log P(G) + C,$$

where

$$P(G|D) \ := \ \int P(D|G,x)P(x|G)dx.$$

The choice of the priors $P(G)$ and $P(x|G)$ defines the score, but finding a network G which maximizes the score turns to be an *optimization problem* (see [14, 27]). Such an optimization problem is hard to solve because there is no known polynomial time algorithm to find the global maximum. Thus, it is called an *NP-hard* (nondeterministic polynomial time) problem. Hence, *Friedman et al.* [14] used a heuristic search method but reaching to a network cannot be guaranteed since we only have a bounded finite number of expression data among the thousands of genes [30]. This backbone arises the dimensionality problem for the method.

However, in [14] a heuristic search algorithm for deducing the network has been proposed to overcome such a discrepancy by focusing on the properties common to high-scoring networks. For this, the authors used two particular tools, namely, *Markov relations* and *order relations* between pairs of variables X and Y. (For details and examples please refer to [14, 50, 51].)

Since there is an incomplete information in gene networks, using a Bayesian approach seems to be reasonable. However, dynamical aspects of gene regularity networks are lost in this method. That is why, to some extent, *Murphy and Mian* [42] developed *dynamical Bayesian networks.*

3.2.3 Boolean Networks

In reality, many biological processes evolving in time exhibit an S-shaped, or sigmoidal curve, but in *Boolean networks* this is converted to the *on-off idea*. Thus, we have some discontinuity instead (see Figure 3.3):

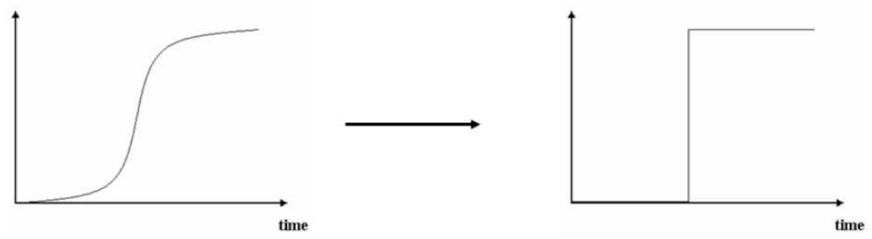

Figure 3.3: From reality to Boolean networks.

Boolean networks (BN) are systems of variables with a possible state of activity as *on* or *off*. Hence, they deal with (1) the *presence* or (0) the *absence* of genes. Moreover, activity of each element (gene) can depend on prior activity of some other gene according to a *Boolean* switching function.

The Boolean network model became popular after studies done by *Kauffman* (1969) for the modeling of *metabolism* of living organisms. It is mathematically defined as:

$G(V, F)$, where V is the set of nodes (genes) and F is a list of Boolean functions. If the vector $\hat{x}$ represents the state of n regulatory elements, then each $\hat{x}_i$ has only a value 0 or 1, and the total number of states is 2^n. At time $t+1$, the state of the gene $\hat{x}_i$ is determined by a Boolean function f_i which uses the k states of n elements at previous time t. Here, k defines the *number of connectivity* which varies from gene to gene. As an illustration, for $n = 3$ and $k = 2$, we give the following simple model:

$$\begin{aligned}
\hat{x}_1(t+1) &= \hat{x}_2(t) \text{ and } \hat{x}_3(t), \\
\hat{x}_2(t+1) &= \hat{x}_1(t) \text{ or } \hat{x}_3(t), \\
\hat{x}_3(t+1) &= \hat{x}_2(t) \text{ nor } \hat{x}_3(t).
\end{aligned}$$

Such an interaction is also represented in the form of a wiring diagram as shown in Figure 3.4:

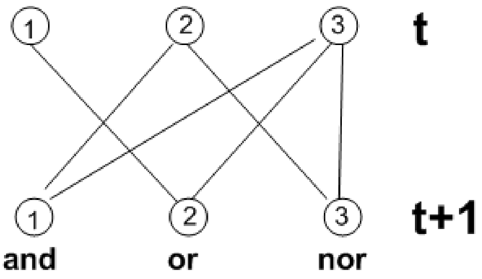

Figure 3.4: Wiring diagram of the Boolean network.

All three variables affect each other by predefined Boolean functions. For example, the next state of x_3 depends on the present states of x_2 and x_3. In Kauffman's example [33], the 8 states are displayed in Table 3.1.

	t			$t+1$	
x_1	x_2	x_3	x_1	x_2	x_3
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	1
1	0	0	0	1	1
1	0	1	0	1	1
1	1	0	0	1	1
1	1	1	1	1	1

Table 3.1: The iteration table [33].

As it can be seen in the iteration table, the state vector 000 at t moves to and remains in the state 000 at time $t+1$. So, we figured out that this initial state is connected with itself. A sequence of states which are connected with transitions are called the *trajectories* of the systems. Since the possible number of states is finite, all initial states of a trajectory eventually arrive at a steady state or state cycle, also known as *attractor*. In Figure 3.5, we see the state cycles for Kauffman's model.

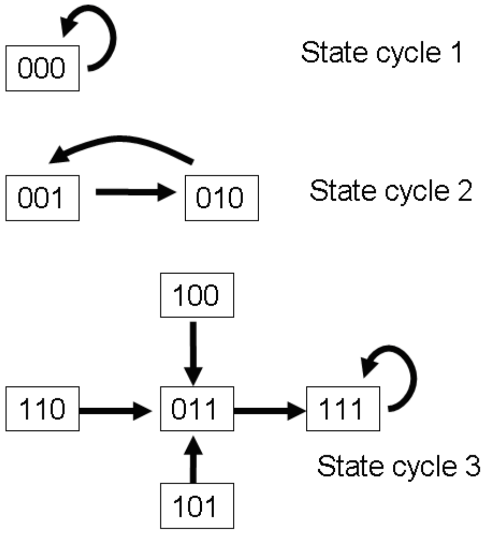

Figure 3.5: Limit cycles of Kauffman's example [33].

Drawing out such diagrams for small networks is possible by hand, but larger networks require computer programs. *Wuensche* [60] has developed such a computer program called *DDLab* to find attractors and their basins of attraction.

The basic logic behind the Boolean networks is to generate random networks with local properties that hold for all members of the system [33]. By defining the attractors, basins of attraction and trajectories, it is possible to investigate the simplifications of the local properties for the global dynamics of the network. Thus, the transition tables allowed algorithms like REVEAL to be constructed [39]. Basically, it searches the logical rules with respect to a given finite set of data. By assuming that the number of regulatory genes is bounded, *Akutsu, Miyano and Kuhara* [2] proposed a simpler algorithm such that it looks whether a unique Boolean exists or not.

Kauffman [33] has calculated that for small values of the number of connectivity (k) and certain choices of regulatory functions, the system demonstrates a highly-ordered dynamics. In such cases, the expected number of attractors was empirically found to be about $\sqrt{n}$ and the length of cycles is also proportional to $\sqrt{n}$ where n is the number of regulatory elements, genes. As k increases, the dynamics gets a more and more complex structure such that some *chaotic* behaviors can be observed [33].

Thomas and Kauffman [56] showed that replacing S-shaped functions by Boolean step functions, the qualitative behaviour of a gene network does not change but adding a particular dynamics to the system is very hard. In addition, although *on-off* BN idealization is better for nonlinear systems, in reality, instead of simultaneous activation there is a time delay between switching states. Thus, *Thomas* [56] has expanded the notion introduced by *Kauffman* by adding *multiple time delays.* Further extension considered by *Öktem, Pearson and Egiazarian* [45] by passing from time discrete to time continuous models.

3.2.4 Dynamical System Models

Differential equations are one of the most widely used modeling formalisms in mathematical biology because they have the capability of capturing behaviors like oscillations, cyclical patterns, multistationarity, switch-like, etc. [17].

The concentrations of gene products, such as RNAs and proteins, are continuous values, which are more accurate and can provide a detailed understanding of the nonlinear dynamical behavior exhibited by biological systems. A differential relation between n variables of gene networks generally is represented in the form

$$\frac{dx_j}{dt} = f_j(x) \quad (j = 1, 2, ..., n),$$

where $x = (x_1, x_2, ..., x_n)^T$ is the vector of positive concentrations of proteins, mRNAs, or small components, and $f_j : \mathbb{R}^n \to \mathbb{R}$ are nonlinear functions ($j = 1, 2, ..., n$).

A first differential equation or dynamic system model consisting of mRNA and protein concentrations was proposed by *Chen and Aihara* [8] in the form of $\dot{E} = ME$, where M is a constant matrix and E is the expression level of individual genes. Later on, *De Hoon and Imoto* [29] used this linear model on mRNA data of *Bacillus subtilis* to estimate M by the maximum likelihood estimation method. In 2001, *Sakamoto and Iba* [47] proposed a more flexible model

$$\dot{E}_j = F_j(E_1, E_2, ..., E_n) \quad (j = 1, 2, ..., n)$$

with F_j being functions of $E = (E_1, E_2, ..., E_n)^T$ determined by genetic programming and least-squares methods.

The models described above were refined by *Gebert, Lätsch, Pickl, Weber and Wünschiers* [15] with many ideas. They regard the model $\dot{E} = M(E)E$ with its deterministic matrix multiplicative form. Here, the matrix M, not usually a constant matrix, depends on E. In the same study, to modify the optimization problem which will be mentioned in Chapter 4, the solution space is restricted by assuming that the number of regulating factors for each gene is bounded.

Based on the idea in [16], *Yılmaz* [63] modeled the gene expression patterns by

$$\dot{E} = F(E),$$

where the right-hand side $F(E)$, $F = (F_1, F_2, ..., F_n)^T$, component-wisely consists of a sum of quadratic functions:

$$F_j(E) = f_{j,1}(E_1) + ... + f_{j,n}(E_n) \quad (j = 1, 2, ..., n).$$

(For details please we refer to Chapter 4.) In our study, we develop and generalize the model $\dot{E} = M(E)E$ with an affine term $C(E)$, namely,

$$\dot{E} = M(E)E + C(E). \tag{3.2.1}$$

The basic example is given by $C(E) \equiv E$, corresponding to the absolute (scalar) term from *Yılmaz* [63], quadratic ansatz. However, also *nonconstant* functions $C(E)$ are possible. The right-hand side $M(E)E + C(E)$ may be regarded as the result of factorizing a vector-valued function $F(E)$ withrespect to E. Herewith, it may represent to an extent that right-hand side $F(E)$ is close to multiplicative matrix form $M(E)E$ in n dimension.

By writing $C(E) = \check{M}(E)\check{E}$ where $\check{M}(E)$ is a $n \times n$ diagonal matrix, we eventually find that $\check{E}$ is constant. In fact, we shall consider the initial value problem with $E(t_0) = E_0$ and $\check{E}(t_0) = (1, 1, ..., 1)^T$. Thus, we reformulate (3.2.1) as

$$\begin{pmatrix} \dot{E} \\ \dot{\check{E}} \end{pmatrix} = \left(\begin{array}{cc} M(E) & \check{M}(E) \\ 0 & 0 \end{array} \right) \begin{pmatrix} E \\ \check{E} \end{pmatrix}.$$

Canonically defining

$$\mathbb{E} := \begin{pmatrix} E \\ \check{E} \end{pmatrix} \quad \text{and} \quad \mathbb{M}(\mathbb{E}) := \left(\begin{array}{cc} M(E) & \check{M}(E) \\ 0 & 0 \end{array} \right),$$

we have reintroduced the following general form

$$\dot{\mathbb{E}} = \mathbb{M}(\mathbb{E})\mathbb{E},$$

which is characterized by a matrix multiplication.

Since one of our main concerns is stability of the system (3.2.1), we investigate stability of the previous new system in terms of *polytopes* [15] in the extended space [55] (see Chapters 4-5).

3.2.5 Hybrid System Models

There also exist *hybrid models* [8, 11, 20] which combine discrete and continuous system models and illustrate both switch-like and smooth variations of genetic networks. Such a kind of behavior arises in both man-made systems like traffic flows and living systems in nature.

In the hybrid scheme, the concentration of proteins (or gene expressions) are supposed to obey the switching linear differential equations

$$\dot{y}_j = -y_j + F_j(\tilde{y}) \ \ (j = 1, 2, ..., n), \qquad \tilde{y} = (\tilde{y}_1, \tilde{y}_2, ..., \tilde{y}_n)^T,$$

where

$$\tilde{y}_j = \begin{cases} 0 & \text{if } y_j < 0, \\ 1 & \text{if } y_j > 0. \end{cases}$$

Since, there exist time delays in the gene expression process, *Chen et al.* [8] have improved the time-continuous models by time delays. In general, simulation these continuous models generally take much more time than the Boolean models need [8]. In terms of *delay differential equations* (DDE), they proposed the following model:

$$\begin{aligned} \dot{m}(t) &= -K_m m(t) + c(p(t,\tau_p)), \\ \dot{p}(t) &= -K_p p(t) + d(m(t,\tau_m)). \end{aligned}$$

Here, $m = (m_1, m_2, ..., m_n)^T \in \mathbb{R}^p$ and $p = (p_1, p_2, ..., p_n)^T \in \mathbb{R}^n$ stand for concentrations of mRNAs and proteins, respectively, $K_m = diag(k_{m1}, k_{m2}, ..., k_{mn}) \in \mathbb{R}^{n\times n}$ and $K_p = diag(k_{p1}, k_{p2}, ..., k_{pn}) \in \mathbb{R}^{n\times n}$ are positive real diagonal matrices representing the degradation. Furthermore, $\tau_m = (\tau_{m1}, \tau_{m2}, ..., \tau_{mn})^T \in \mathbb{R}^n$ and $\tau_p = (\tau_{p1}, \tau_{p2}, ..., \tau_{pn})^T \in \mathbb{R}^n$ are positive real vectors indicating the time delays with $m(t,\tau_m) = (m_1(t-\tau_{m1}), (m_2(t-\tau_{m2}), ..., m_n(t-\tau_{mn}))^T$ and $p(t,\tau_p) = (p_1(t-\tau_{1p}), p_2(t-\tau_{2p}), ..., p_n(t-\tau_{pn}))^T$. The vectors $c(p) = (c_1(p), c_2(p), ..., c_n(p))^T$ and $d(m) = (d_1(m_1), d_2(m_2), ..., d_n(m_n))^T$ are generally nonlinear terms corresponding to switching-like phenomena in the form of sigmoid functions, e.g., $\tanh(\frac{x_i}{\varepsilon})$ or $\frac{x_i^k}{x_i^k+\varepsilon}$ with a positive constant ε and hill coefficient k.

3.2.6 Models with Piecewise-Linear Differential Equations

The S-shaped, or sigmoid, change in concentration is approximated step functions in Boolean networks. An alternative approximation can be made by piecewise-linear differential equations (see Figure 3.6).

It is known that rates of degradation of some mRNA molecules and the maximal

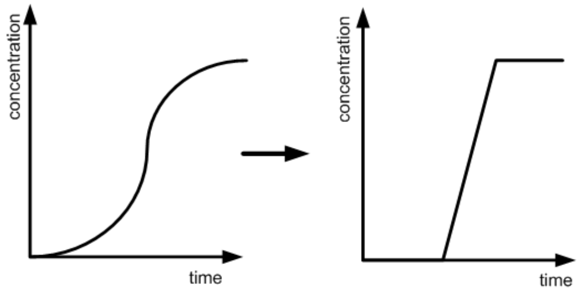

Figure 3.6: Approximation with piecewise-linear equations.

possible number of regulating genes for one gene can be included into the model as known parameters as well. A simple formalism of this approach stated in [20, 40, 56] is

$$\frac{dx_j}{dt} = g_j(x) - \gamma_j x_j \quad (j = 1, 2, ..., n),$$

where x_j denotes the cellular concentration of gene i and γ_i is the degradation rate of x_j.

Gebert, Radde and Weber [17] focused on the model represented by Figure 3.7 which is mathematically formulated as

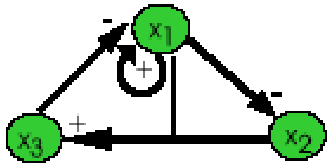

Figure 3.7: Example of a model derived from PLDE [17].

$$\begin{aligned}
\dot{x}_1 &= k_{1,1} \cdot h^{+}(x_1, \theta_{1,1}, m_{1,1}) + k_{1,3} \cdot h^{-}(x_3, \theta_{1,3}, m_{1,3}) - \gamma_1 x_1, \\
\dot{x}_2 &= k_{1,1} \cdot h^{-}(x_1, \theta_{2,1}, m_{2,1}) - \gamma_1 x_1, \\
\dot{x}_3 &= k_{3,1,2} \cdot h^{+}(x_1, \theta_{3,1}, m_{3,1}) \cdot h^{+}(x_2, \theta_{3,2}, m_{3,2}) - \gamma_3 x_3.
\end{aligned}$$

Using one variable for the mRNA concentration with assuming that the proteins are approximated by this variable can lead to bad approximations, but the fraction of post-transcriptional regulations in procaryotes is less than those in eucaryotes. Thus, the model approaches such as the above ones can provide a qualitative mathematical analysis only for procaryotes.

3.3 Our Extended Modeling

Most of the common modeling and simulation techniques for gene regulatory networks are deterministic. *Hoon* [29] has compared about ten different modeling formalisms and categorized them as *coarse*, *average* and *fine*. According to these comparison results, the models built up from partial and nonlinear differential equations are found to be qualitatively fine.

We also claim that modeling by a dynamical system is better prepared to capture the underlying structure of gene networks. In [1, 16, 63], the constant form, C, the linear form, $ME + C$, and the nonlinear (especially, the quadratic) form, $M(E)E + C$, of the time-continuous models are already compared in terms of fitting the experimental data, and it is numerically shown [63] that nonlinear models do a much better data fitting. Furthermore, the allowance of nonlinearity in modeling guarantees a better prediction of the future expression states by means of the corresponding time-discrete dynamics [63].

In this work, we extend the nonlinear continuous model $\dot{E} = M(E)E$ by adding an affine shift term to the right hand side of the equation, i.e.,

$$\dot{E} = M(E)E + C(E),$$

where $C(E)$ stands for the additive affine shift term. Differently from $M(E)E$, the second term (*shift*) $C(E)$ does not need to reveal E as a factor, e.g., exp or cos. In case where $M(E)$ and $C(E)$ are polynomial, component-wise understood, $M(E)E$ may have a higher degree than $C(E)$. The important and basic case where $C(E)$ is constant, i.e., $C(E) \equiv C$, was in modeling studied by *Yılmaz* [63].

Using such an additive shift $C(E)$ has several advantages in both biological sense and mathematical analysis. We will deepen and integrate these details of our model further in the coming chapters.

Chapter 4

The Extended Model

4.1 The State of the Art

Experiments involving dynamic measurements typically require a careful definition of the physical quantities to be measured and the instrumental means. One is often interested in testing hypotheses or making inferences on the basis of temporal patterns in time-series data. When the observed dynamics are relatively simple, such as sinusoidal periodicities, traditional analytical tools such as Fourier transforms are easily used to characterize the patterns. More complex dynamics like gene expression patterns can require more sophisticated approaches because of bifurcations, chaotic oscillations and unknown regulating factors.

Let the n-column vector $E = E(t)$ represent gene expression patterns at different times t. We denote the given finite set of experimental results as $\bar{E}_0, \bar{E}_1, ...,$ $\bar{E}_{l-1}$, where each $\bar{E}_m \in \mathbb{R}^n$ corresponds to the gene profile taken at time $\bar{t}_m$. Furthermore, we denote the time-discrete approximations introduced subsequently by $E_0, E_1, ..., E_{l-1}, ...$

Gebert et al. [16] developed a *time-continuous* model by taking account of that interaction between variables is nonlinear but the number of associated regulating influences is bounded. This flexible model was formulated in the following multiplicative form of nonlinear ordinary differential equations:

$$(\mathcal{CE}) \quad \dot{E} = M(E)E.$$

This system of ordinary differential equations $(\mathcal{CE})$ is autonomous; i.e., the right-hand side depends on the state vector E, but not on the time t. This implies that nontrivial trajectories (i.e., ones which are not stationary) do not cross themselves. The matrix $M(E)$ is component-wise defined by a family of any class of functions including unknown parameters. For example, for a (2×1)-vector $E = (E_1, E_2)^T$, the matrix $M(E)$ could be

$$M_{\substack{a_1,a_2,a_3,a_4\\ a_5,a_6,a_7,a_8}} := \begin{pmatrix} a_1 E_1^2 + a_2 E_1 E_2 & a_3 E_2 \cos(E_1) + a_4 \\ a_5 \cos(E_2) + a_6 E_1 & a_7 E_1^2 + a_8 E_2 \end{pmatrix}.$$

We note that the polynomial, trigonometric, but otherwise also exponential, etc., entries represent the growth or other kinds of changes in the concentrations. In this example, there are eight parameters in total.

Now, two different stages of problem come into consideration concerning the parametrized entries of the matrices $M(E)$. Firstly, the optimization problem of discrete (least-squares) approximation which can be written as

$$\underset{\alpha}{\text{minimize}} \quad \sum_{k=0}^{l-1} ||M_\alpha(\bar{E}_k)\bar{E}_k - \dot{\bar{E}}_k||^2.$$

Here, the least-squares methods of linear and nonlinear regression are used to estimate the vector α of a first part of the parameters to fit the set of given experimental data and to characterize the statistical properties of estimates. Secondly, we investigate which components of remaining parameter vector β produce a stable, which ones an unstable influence on the dynamics. For a closer presentation of this two-stage problem from parametric optimization, we refer to [16, 31].

4.2 Model with Quadratic Polynomials

An extension to $(\mathcal{CE})$ was considered by *Yılmaz et al.* [63] by proposing the nonlinear form

$$\dot{E} = F(E),$$

where $F = (F_1, F_2, ..., F_n)^T$ is a tuple of functions depending on $E \in \mathbb{R}^n$. More specifically, for representing the influence of gene i to gene j, the authors considered the quadratic (also: constant, linear) functions $f_{j,i}(x) = a_{j,i}x^2 + b_{j,i}x + c_{j,i}$ where $x = E_i$ denotes the concentration of $gene_i$ and $a_{j,i}, b_{j,i}, c_{j,i} \in \mathbb{R}$. In [63], the least-squares approximation errors of constant, linear and nonlinear, in fact quadratic, models are compared. Here, by saying error we mean the least-squares difference between the experimental data $\bar{E}$ and the model approximation E. In according to this comparison and the prediction by a corresponding time-discrete dynamics, the quadratic model yields better results.

4.3 Generalized Model

The model extended by *Yılmaz et al.* [63] allows nonlinear interactions and uses an affine constant term of shift. However, the recursive iteration idea mentioned in [15] looks to be lost by this shift term. Here, we generalize $(\mathcal{CE})$ by the following *affine* addition:

$$(\mathcal{ACE}) \qquad \dot{E} = M(E)E + C(E),$$

allowing even a non-constant, state-depending shift. We defend and underline that the additional column vector $C(E)$ can help us for accounting the environmental changes in the sense of perturbations and capturing the dynamics better. Our approach in overcoming the more complex form of $(\mathcal{ACE})$ algorithmically is that $C(E)$ can be written as

$$C(E) = \check{M}(E)\check{E},$$

where

$$\check{M}(E) := diag(C^T(E)) = \begin{pmatrix} C_1(E) & & & & 0 \\ & C_2(E) & & & \\ & & \cdot & & \\ & & & \cdot & \\ & & & & \cdot \\ 0 & & & & C_n(E) \end{pmatrix} \quad \text{and} \quad \check{E} := \begin{pmatrix} \check{E}_1 \\ \check{E}_2 \\ \cdot \\ \cdot \\ \cdot \\ \check{E}_n \end{pmatrix}.$$

In fact, we shall see by means of the corresponding initial value $\check{E}(t_0) = e$ ($e := (1, 1, ..., 1)^T$) that the (time-depending) variable $\check{E}$ is constant and identically $\check{E} \equiv e$. In this sense of initial conditions, $(\mathcal{ACE})$ is equivalent to

$$\dot{E} = M(E)E + \check{M}(E)\check{E}.$$

Let us define the vector and the matrix of *canonical* form:

$$\mathbb{E} := \binom{E}{\check{E}} \quad \text{and} \quad \mathbb{M}(\mathbb{E}) := \begin{pmatrix} M(E) & \check{M}(E) \\ 0 & 0 \end{pmatrix},$$

so that we end up with the following form of the extended initial value problem:

$$(\mathcal{CE})_{ext} \quad \dot{\mathbb{E}} = \mathbb{M}(\mathbb{E})\mathbb{E} \qquad \text{and} \qquad \mathbb{E}_0 := \mathbb{E}(t_0) = \begin{pmatrix} E(t_0) \\ \check{E}(t_0) \end{pmatrix} = \begin{pmatrix} E_0 \\ 1 \\ \cdot \\ \cdot \\ \cdot \\ 1 \end{pmatrix}.$$

By the previous reformulation, in our present research, we benefit from both *(i)* the affine shift term structure for better modeling gene expression patterns and for better future predictions, and *(ii)* the time-continuous matrix multiplication approach by means of dimensional extension into dimension $2n$. The latter multiplicative form will become an *iterative* multiplicative one in the corresponding *time-discrete* dynamics which we are going to introduce.

4.3.1 Time Discretization

Discretization concerns the process of transferring continuous models and equations into discrete counterparts. The *numerical* solution simulating the behavior of a system governed by a system of ordinary differential equations, (ODEs), starting with the initial value E_0 at t_0 gives an approximation to the solution at a discrete set of points. For a particular state, next states are generated iteratively. In other words, we follow trajectories with approximate solution values which are generated step by step in increments moving across a time interval in which the solution is sought.

Stability and precision of the simulation results are two main concerns of corresponding numerical solution of differential equation [10]. By discretizing the continuous process, we are able to compare the approximative results $E_0, E_1, ...,$ $E_{l-1}, ...$ with the given set of experimental results $\bar{E}_0, \bar{E}_1, ..., \bar{E}_{l-1}$.

Dubois and Kalisz [10] have already compared the discretization methods of Euler's and Runga-Kutta in the case of a simple differential equation system: the harmonic oscillator. In this study, it is shown that Euler algorithm is unstable compared to exact solution. Here, we start with Euler's method mainly for conceptual reasons and, for simplicity, of the explanation in the nonextended space $\mathbb{R}^n$ firstly.

4.3.1.1 Euler's Method

Let a general system of ODEs: $\dot{E} = f(t, E)$ be given, where E is a continuous and p times differentiable variable on an interval $I = [a, b]$. If $t, t + h \in I$, then we write the *Taylor series* as

$$E(t+h) = E(t) + hE^{'}(t) + \frac{h^2}{2!}E^{''}(t) + ... + \frac{h^{p-1}}{(p-1)!}E^{(p-1)}(t) + \frac{h^p}{p!}E^{(p)}(\gamma)$$

with an appropriate number $\gamma \in (t, t+h)$ if $h > 0$, and $\gamma \in (t+h, t)$ if $h < 0$ ($|h|$ being small enough). *Euler's method* results from canceling second and higher order terms in Taylor series to approximate the solution

$$E_{k+1} = E_k + h_k f(t_k, E_k) \quad (k \in \mathbb{N}_0).$$

Now, let us come back to $2n$ dimensions. We first apply Euler's method, to discretize the time-continuous process $\mathbb{M}(\mathbb{E})\mathbb{E}$ as follows: For all $k \in \mathbb{N}_0$,

$$\begin{aligned} \frac{\mathbb{E}_{k+1} - \mathbb{E}_k}{h_k} &= \mathbb{M}(\mathbb{E}_k)\mathbb{E}_k && (4.3.1) \\ \Leftrightarrow \quad \mathbb{E}_{k+1} &= (\mathbb{I} + h_k\mathbb{M}(\mathbb{E}_k))\mathbb{E}_k, && (4.3.2) \end{aligned}$$

where h_k is the step-size, i.e., $h_k = t_{k+1} - t_k$, between the discrete times t_k with $t_k < t_{k+1} \quad (k \in \mathbb{N}_0)$.

In our extended model, (4.3.2) looks like:

$$\begin{pmatrix} E_{k+1} \\ \check{E}_{k+1} \end{pmatrix} = (I + h_k \begin{pmatrix} M(E_k) & \check{M}(E_k) \\ 0 & 0 \end{pmatrix}) \begin{pmatrix} E_k \\ \check{E}_k \end{pmatrix}.$$

Let us define

$$\mathbb{M}_k := (I + h_k \begin{pmatrix} M(E_k) & \check{M}(E_k) \\ 0 & 0 \end{pmatrix})$$

so that we obtain the discrete time-varying difference equation and dynamics

$$(\mathcal{DE}) \quad \mathbb{E}_{k+1} = \mathbb{M}_k \mathbb{E}_k \qquad (k \in \mathbb{N}_0).$$

Thus, we iteratively generate, in fact approximate, the next state from the previous one. Setting $\mathbb{E}_0 = (\bar{E}_0^T, 1, 1, ..., 1)^T$, the k^{th} approximation is calculated as

$$(\mathcal{DE}) \quad \mathbb{E}_k = \mathbb{M}_{k-1}(\mathbb{M}_{k-2}...(\mathbb{M}_1(\mathbb{M}_0\mathbb{E}_0))) \;\; (k \in \mathbb{N}_0).$$

Having a multiplicative formula for predictions has a great computational and analytical advantage. However, the matrix multiplications in front of the given initial state $\mathbb{E}_0$ force us to consider the question of stability and boundedness of the solution.

4.3.1.2 Runga-Kutta Method

While solving ODEs numerically, we are faced with two kinds of errors, namely, the rounding error as a result of finite precision of floating-point arithmetic and, secondly, the truncation error associated with the method used. For example, in Euler's method, the truncation error is by far larger because the curve $E(t)$ is approximated by a straight-line between the end points t_k and t_{k+1} of time intervals. In addition, Euler's method evaluates derivatives at the beginning of the subinterval, i.e., at t_k, which makes the method asymmetric with respect to the beginning and the end of the interval. Hence, more symmetric integration methods like *Runge-Kutta method* (RK), which takes into account the midpoint of the interval, can be applied for the system $(\mathcal{CE})_{ext}$. Moreover, Runge-Kutte methods are similar in motivation to the Taylor series approach, but do not require the computation of higher derivatives. Using Runge-Kutta methods has another advantage: it is more sensitive to infinitesimal numerical changes [26].

RK methods use only the information at time t_k, which makes them self-starting at the beginning of integration, and also makes methods easy to program, which accounts in part for their popularity [26].

The idea of applying RK methods for modeling of gene expression patterns has firstly been introduced by *Ergenç and Weber* [13]. Here, we illustrate the application of a different RK method, called *Heun's method.* Heun's method is more illustrative and the simplest case of the Runge-Kutte approach. In our extended space, it is formulated as:

$$\mathbb{E}_{k+1} = \mathbb{E}_k + \frac{h_k}{2}(k_1 + k_2) \quad (k \in \mathbb{N}_0), \tag{4.3.3}$$

where

$$\begin{aligned} k_1 &= \mathbb{M}(\mathbb{E}_k)\mathbb{E}_k \ \text{ is the term } \textit{predictor}\text{, and} \\ k_2 &= \mathbb{M}(\mathbb{E}_k + h_k k_1)(\mathbb{E}_k + h_k k_1) \text{ corresponds to the term } \textit{corrector}. \end{aligned}$$

More explicitly, instead of (4.3.3) we write

$$\begin{aligned} & \mathbb{E}_{k+1} &=& \ \mathbb{E}_k + \frac{h_k}{2}\mathbb{M}(\mathbb{E}_k)\mathbb{E}_k + \frac{h_k}{2}\mathbb{M}(\mathbb{E}_k + h_k\mathbb{M}(\mathbb{E}_k)\mathbb{E}_k)(\mathbb{E}_k + h_k\mathbb{M}(\mathbb{E}_k)\mathbb{E}_k) \\ \Leftrightarrow \quad & \mathbb{E}_{k+1} &=& \ (\mathbb{I} + \frac{h_k}{2}\mathbb{M}(\mathbb{E}_k) + \frac{h_k}{2}\mathbb{M}(\mathbb{E}_k + h_k\mathbb{M}(\mathbb{E}_k))(I + h_k\mathbb{M}(\mathbb{E}_k)))\mathbb{E}_k. \end{aligned}$$

Defining

$$\begin{aligned} \mathbb{I} &:= (2n) \times (2n) \text{ unit matrix in } \mathbb{R}^{2n}, \qquad \text{and} \\ \mathbb{M}_k &:= \mathbb{I} + \frac{h_k}{2}\mathbb{M}(\mathbb{E}_k) + \frac{h_k}{2}\mathbb{M}(\mathbb{E}_k + h_k\mathbb{M}(\mathbb{E}_k)\mathbb{E}_k)(I + h_k\mathbb{M}(\mathbb{E}_k)), \end{aligned}$$

we get the following time-discrete equation

$$(\mathcal{DE})^2_{ext} \qquad \mathbb{E}_{k+1} = \mathbb{M}_k\mathbb{E}_k.$$

In the case of constant matrix $\mathbb{M}$, i.e., if $M(E)$ and $C(E)$ are constant, the time-discrete system would contain a quadratic (in $\mathbb{M}$) term of matrices [13]. That is why, we use the notation $(\mathcal{DE})^2_{ext}$.

We note that Runge-Kutta discretization of our model equation generates a nonlinear discrete equation with respect to parameters. In our case, the term

$$\mathbb{M}(\mathbb{E}_k + h_k\mathbb{M}(\mathbb{E}_k)\mathbb{E}_k)(I + h_k\mathbb{M}(\mathbb{E}_k))$$

shows the parametrical nonlinearity. If we use *implicit* Runge-Kutta methods [13], it may not be possible to get the discrete equation $(\mathcal{DE})^2_{ext}$.

The dynamics of the time-discrete model $(\mathcal{DE})^2_{ext}$ is strongly related with the matrix multiplications. Thus, we investigate the questions concerning how products of matrices $\mathbb{M}_k$ look like, what is the product structure and what does the block structure say about boundedness or unboundedness of the products of finitely many matrices.

4.4 Algebra of Matrix Products

Remember that the matrix in the time-continuous model has the *canonical* form

$$\mathbb{M}(\mathbb{E}) = \begin{pmatrix} M(E) & \check{M}(E) \\ 0 & 0 \end{pmatrix}.$$

These matrices help us for defining relation between genes and understanding the structure of gene networks.

The product of two matrices having this block form is again a matrix in the same structure, because for any $X, Y \in \mathbb{R}^n$ it holds:

$$\begin{aligned} \begin{pmatrix} M(X) & \check{M}(X) \\ 0 & 0 \end{pmatrix}\begin{pmatrix} M(Y) & \check{M}(Y) \\ 0 & 0 \end{pmatrix} &= \begin{pmatrix} M(X)M(Y) & M(X)\check{M}(Y) \\ 0 & 0 \end{pmatrix} \\ &:= \begin{pmatrix} \widetilde{M}(X,Y) & \widetilde{\check{M}}(X,Y) \\ 0 & 0 \end{pmatrix}. \end{aligned}$$

Matrix multiplication is not needed the case for the time-continuous model, but we try to understand whether our matrices $\mathbb{M}_k$ and their products in the time-discrete iterative system have some "canonical" block form or not. After some simplifications, by definition of $\mathbb{M}_k$, where we insert certain arguments and step-lengths h_k, we find that

$$\mathbb{M}_k = \mathbb{I} + \frac{h_k}{2}\begin{pmatrix} M(E_k) & \check{M}(E_k) \\ 0 & 0 \end{pmatrix} + \frac{h_k}{2}\begin{pmatrix} A & \widetilde{A} \\ 0 & 0 \end{pmatrix} + \frac{h_k^2}{2}\begin{pmatrix} B & \widetilde{B} \\ 0 & 0 \end{pmatrix},$$

where $\mathbb{I} = I_{2n}$ (unit matrix of type $(2n) \times (2n)$) and

$$\begin{aligned} A &:= M\left(E_k + h_k\left(M(E_k)E_k + \check{M}(E_k)\check{E}_k\right)\right), \\ \widetilde{A} &:= \check{M}\left(E_k + h_k\left(M(E_k)E_k + \check{M}(E_k)\check{E}_k\right)\right), \\ B &:= M\left(E_k + h_k\left(M(E_k)E_k + \check{M}(E_k)\check{E}_k\right)\right) M(E_k) \quad \text{and} \\ \widetilde{B} &:= M\left(E_k + h_k\left(M(E_k)E_k + \check{M}(E_k)\check{E}_k\right)\right) \check{M}(E_k). \end{aligned}$$

We conclude that $\mathbb{M}_k$ has its final *canonical* block form

$$\begin{pmatrix} \widehat{M(E_k)} & \widehat{\check{M}(E_k)} \\ 0 & I_n \end{pmatrix}.$$

Here, one of our main questions concerns *iterative multiplication* of matrices having the same form with model $\mathbb{M}_k$. In the next chapter, for our stability analysis we have to study these matrices $\mathbb{M}_k$, namely, the introduction (selection) and their iterative multiplication, in detail. What a form has the product of two and, by induction, finitely many matrices $\mathbb{M}_k$? By using $\hat{A}, \hat{B}, \hat{C}, \hat{D}$ to represent the corresponding block matrices, we calculate:

$$\begin{pmatrix} \widehat{A} & \widehat{B} \\ 0 & I_n \end{pmatrix}\begin{pmatrix} \widehat{C} & \widehat{D} \\ 0 & I_n \end{pmatrix} = \begin{pmatrix} \widehat{A}\widehat{C} & \widehat{A}\widehat{D} + \widehat{B} \\ 0 & I_n \end{pmatrix} =: \begin{pmatrix} \widehat{K} & \widehat{L} \\ 0 & I_n \end{pmatrix}.$$

We observe that any finite product of matrices in the extended space preserves the same structure as a single matrix $\mathbb{M}_k$. This same appearance as $\mathbb{M}_k$ enables us for doing a similar, n-dimensional stability analysis performed for $(\mathcal{CE})$ in [15].

In fact, multiplying any canonical matrix $\mathbb{M}_k$ by a vector $(E^T, e^T)^T$ reproduces a vector $(\widetilde{E}^T, e^T)^T$ of the same type. For this reason, there is no restriction if we focus our attention on the first n coordinates of the vectors and on the first n rows of our matrices.

Stability analysis of the time-continuous dynamical systems is closely related with the time-discrete case concerning the definition, eigenvalues, matrix norm and suitable Lyapunov functions. An algorithmic method which studies stability and introduces Lyapunov functions in the time-discrete case has first been introduced by *Brayton and Tong* [5], then modified and implemented by *Pickl* [15]. Very analogously, we apply the same algorithm to $(\mathcal{DE})^2_{ext}$ in the extended space in Chapter 5, keeping the focus on the first n coordinates in mind.

Chapter 5

Stability Analysis

5.1 Introduction

In the biological context, especially in population dynamics, a stability concept is well-defined [36]. From the point of view of genetics, it corresponds to the resistance of chemical compounds (e.g., proteins and nucleic acids) towards conformational changes and is associated the thermodynamics [15]. Cancer, for example, can be viewed as a genomic instability stemming from mutations and variance in chromosome numbers.

From the point view of mathematics, stability is a condition on the behavior of dynamical systems under initial perturbations around equilibrium points. This can be thought as a characterization of environmental changes given to the system, of disease or of the treatment of the cell by some medical or radiation. Since gene expression values lie in a bounded region, stable solutions can refer to a better goodness of data fit (see Figure 5.1).

The traditional logic behind the stability notion stating that the only good systems are those ones with all of their qualitative features not changing by perturbations might be reformulated [21]. For example, in terms of biology, stability could also be interpreted negatively, namely as a lack of flexibility or of the readiness to adopt to changes of the environment.

Since there are no general rules for writing down analytic formulas for the solutions of systems of nonlinear differential equations, the analysis of such systems is accomplished by two approaches [41]:

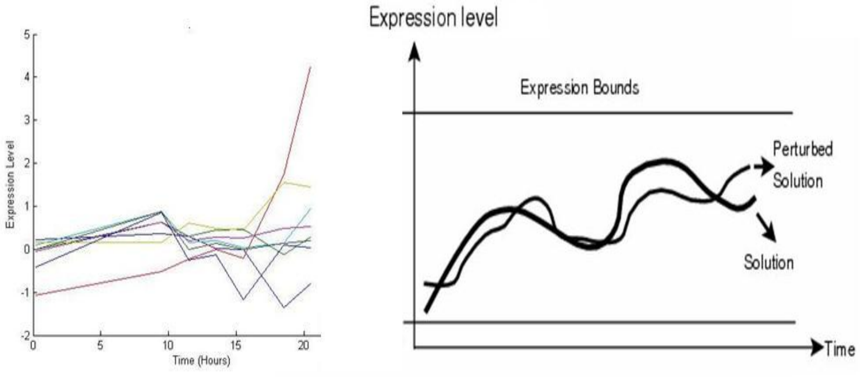

Figure 5.1: Real data and model stability.

1. A quantitative approach including numerical solutions with the help of sensitive computers.
2. A qualitative approach without seeking specific solutions but concerned with the behavior of families of solutions of given differential equations.

These approaches complete each other in applications. The fundamental principle of the qualitative approach includes the *stability* of an equilibrium point which is one of the main concerns in dynamical system theory.

Here we start with mathematical definition of stability of a time-continuous system:

Definition 5.1. A point $E^* \in \mathbb{R}^n$ is called an *equilibrium point of system* (S) $\dot{E} = f(t, E)$ where $(t, E) \in \mathbb{R} \times \mathbb{R}^n$ if $f(t, E^*) = 0$ for all $t \in \mathbb{R}$. An equilibrium E^* of (S) is called *stable* (in the Lypunov sense) if for every $\varepsilon > 0$ there exists a $\delta = \delta(\varepsilon) > 0$ such that at time $t = t_0$ it satisfies $||E(t) - E^*|| \leq \delta$ and for all $t > t_0$ it holds $||E(t) - E^*|| < \varepsilon$.

Roughly speaking, the solutions initiated from small neighborhood of the equilibrium never escapes and remains in a bounded region.

Another widely useful stability aspect for nonlinear systems is *BIBO stability* [44], meaning that any bounded input to the system ends up with a bounded deviation of the solutions.

A common method for demonstration of stability is to find a Lyapunov function for that system.

Definition 5.2. Let $D \subseteq \mathbb{R}^n$ be a open subset of real Euclidean space. A *Lyapunov function* is a real valued function $L : D \to \mathbb{R}$ that is continuously differentiable with the following properties:

1. Positive definite, i.e., $L(E) > 0$ for all $E \neq E^*$ and $L(E^*) = 0$,
2. Along the solution of (S), $\dot{L} < 0$ for all $E \neq E^*$.

The existence of a Lyapunov function for which the derivative of function L along the solution of the system $\dot{E} = f(t, E)$,

$$\dot{L} = \sum_{j=1}^{n} \frac{\partial L}{\partial E_j} \dot{E}_j$$

is smaller or equal to zero, *negative semi-definite*, on some region containing the origin, guarantees the stability of the zero solution of $\dot{E} = f(t, E)$ [28]. While the existence of a Lyapunov function for which derivative of L along solutions of $\dot{E} = f(t, E)$ is strictly less then zero, *negative definite*, on some region containing the origin guarantees the asymptotical stability of the zero solution [22, 32].

In order to introduce the idea of a Lyapunov function a bit closer, we give an example in the time-continuous case.

Example 5.3. ***Simple Pendulum***

Assume that for a simple pendulum we have the following governing equations

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= -k \sin(x_1), \end{aligned}$$

where $k > 0$. Basically the equilibrium point is $X^* = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$. Let us choose our Lyapunov function as

$$L(x_1, x_2) := \frac{1}{2}x_2^2 + k(2 - \cos(x_1)),$$

which is positive definite. Along the solution, the derivative of L is

$$\dot{L} = x_2\dot{x}_2 + k\sin(x_1)\dot{x}_1 = -x_2 k\sin(x_1) + k\sin(x_1)x_2 = 0,$$

which is negative semi-definite. Thus, we say that the solution $X^* = 0$ is stable.

However, the problem of finding a suitable Lyapunov function arises because there is no general rule for establishing such functions [5]. Therefore, *Brayton and Tong* described a promising iterative tool for determining stability of non-linear systems of differential equations in terms of a given finite set of matrices namely, $\mathcal{M} = \{\mathbb{M}_0, \mathbb{M}_1, ..., \mathbb{M}_{l-1}\}$, derived from discretization.

5.2 Stability of a Single Matrix

We begin with the stability of a single matrix M, i.e., in case $\mathcal{M} = \{M\}$.

Definition 5.4. (***stability of a matrix***) A matrix M is called *stable* if there exists a number $K \in \mathbb{R}$, $K > 0$, such that $||M^j|| \leq K$ for all $j \in \mathbb{N}$.

For any matrix $M \in \mathbb{C}^{n\times n}$, there exists an invertible matrix $P \in \mathbb{C}^{n\times n}$ such that [35]

$$P^{-1}MP = J = diag(J_1, J_2, ..., J_n),$$

where each block matrix J_k is of the form

$$J_k = \begin{pmatrix} \lambda_k & 1 & . & . & . & 0 & 0 \\ 0 & \lambda_k & . & . & . & 0 & 0 \\ . & . & . & & & . & . \\ . & . & & . & & . & . \\ . & . & & & . & . & . \\ 0 & 0 & . & . & . & \lambda_k & 1 \\ 0 & 0 & . & . & . & 0 & \lambda_k \end{pmatrix}.$$

This matrix J is called the *Jordan form* of the matrix M.

Using Jordan form and the matrix norm $\|.\|$ which is introduced by Euclidean vector norm $\|.\|_2$, we obtain:

$$\begin{aligned} M^j &= PJ\overbrace{P^{-1}P}^{I_n}J\overbrace{P^{-1}P}^{I_n}...PJP^{-1} = PJ^jP^{-1} && (5.2.1) \\ \Rightarrow \quad \|M^j\| &= \|PJ^jP^{-1}\| \leq \|P\|\|J^j\|\|P^{-1}\| \leq \|J\|^j. && (5.2.2) \end{aligned}$$

Lemma 5.5. *If $J \in \mathbb{C}^{n\times n}$ is a block matrix of type $s \times s$ whose eigenvalues $\lambda_j(\mathbb{M})$ have magnitude $|\lambda_j(\mathbb{M})|$, then*

$$\|J^j\|_\infty = \sum_{k=0}^{s-1} \binom{j}{k} |\lambda(J)|^{j-k}.$$

Proof. Please see [4]. □

From the lemma above, an equivalent characterization for stability of a matrix can be stated as follows

Corollary 5.6. *[4, 5] M is stable $\Leftrightarrow |\lambda(M)| \leq 1$ and in case of $|\lambda_j(M)| = 1$ for some eigenvalue $\lambda_j(M)$, then the algebraic multiplicity and geometric multiplicity coincide.*

5.3 Stability of a Set of Matrices

Stability of a set of matrices implies that the system of differential equations is stable. Such a set is derived by means of discretizing the function representing the continuous system (see Chapter 4) or by means of results coming from early experiments.

In this section, we will present a definition for the stability of a set of matrices and will produce the connections between a stable set and a norm. Based on this norm, statements can be encountered about the stable or unstable behavior of the set of matrices.

For the stability of a set of matrices it *not* sufficient that the single matrices in this set are stable [4, 5].

Let $\mathcal{M} = \{\mathbb{M}_0, \mathbb{M}_1, ..., \mathbb{M}_{l-1}\}$ be a set of given real matrices. We will consider the larger multiplicative semigroup $\mathcal{M}'$ containing all finite products of matrices produced from $\mathcal{M}$. In other words,

$$\mathcal{M}' = \{\prod_{s=1}^{k} \mathbb{M}_s^{l_s} : \ \mathbb{M}_s \in \mathcal{M}, \quad l_s \in \mathbb{N} \ (s \in \{1, 2, ..., k\}),$$

$$\mathbb{M}_s \neq \mathbb{M}_{s+1} \ \ \forall s \leq k-1, \quad k \in \mathbb{N}, \ \sum_{s=1}^{k} l_s = p, \quad p \in \mathbb{N}\}.$$

Now, we introduce a concept important for us: stability of a set of matrices. We begin with the formal definition:

Definition 5.7. (***stability of a set of matrices***) A set $\mathcal{M}$ of complex $(2n \times 2n)$-matrices is stable if for every neighborhood of the origin $\mathbb{U} \subseteq \mathbb{C}^{2n}$ there exists another neighborhood of the origin $\mathbb{V}$ such that $\mathbb{M}V \subseteq U$ for each $\mathbb{M} \in \mathcal{M}'$.

Figure 5.2 gives a small geometric interpretation of this definition.

Since our dynamical analysis bases on the linear algebra of matrices, especially, on the spectral study of eigenvalues, we have to locate our study over the

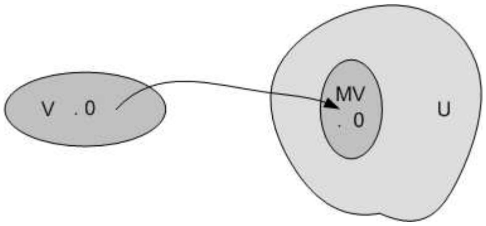

Figure 5.2: Stability of a set of $(n \times n)$-matrices [4].

complex numbers rather than the reals.

The following conclusion for stability of a set of matrices is formulated and proved by *Brayton et al.* [5]:

Lemma 5.8. *[5] A set of matrices $\mathcal{M}$ is stable if and only if there exists a bounded neighborhood of the origin $\mathbb{B} \subset \mathbb{C}^{2n}$ such that for each $\mathbb{M} \in \mathcal{M}'$ it holds*

$$\mathbb{M}\mathbb{B} \subseteq \mathbb{B}.$$

*Furthermore, $\mathbb{B}$ can be chosen to be convex and balanced, i.e, for any pair of vectors $u, v \in \mathbb{B}$, the vector $(1-\lambda)u + \lambda v$ is in B for all $\lambda \in [0,1]$ (**convexity**), and if $z \in \mathbb{B}$, then we have $ze^{i\theta} \in \mathbb{B}$ for all $\theta \in \mathbb{R}$ (**balancedness**).*

Proof. Please see [4, 5]. □

In [5], the authors developed a criterion that enables a relation between a Lypunov function and the stability of a set of matrices. A matrix norm plays the role of this function. This connection is stated by the following lemma:

Lemma 5.9. *[5] If the set of matrices $\mathcal{M}$ is stable, then there exists a norm, $\|.\|$, such that*

$$\|\mathbb{M}\mathbb{E}\| \leq \|\mathbb{E}\| \quad \forall\ \mathbb{M} \in \mathcal{M}, \qquad \mathbb{E} \in \mathbb{C}^{2n}.$$

Many studies have been made for understanding the special structure of the Lyapunov functions, but *Brayton et al.* [5] introduces an algorithmic method where the structure of Lyapunov functions is related with polytopes in a *constructive* way. Thus, the norm mentioned above will be obtained by means of the following construction principle.

Theorem 5.10. ***(Brayton and Tong [5])*** *Let $\mathcal{M} = \{\mathbb{M}_0, \mathbb{M}_1, ..., \mathbb{M}_{m-1}\}$ be a finite set of m $(n \times n)$-matrices. Let $\mathbb{B}_0 \subset \mathbb{C}^{2n}$ be a bounded neighborhood of the origin. If for all $k \in \mathbb{N}$, we define the sets in a iterative way*

$$\mathbb{B}_k := \mathcal{H}\left(\bigcup_{j=0}^{\infty} \mathbb{M}_{k'}^{j} \mathbb{B}_{k-1}\right), \quad \textit{where} \quad k' \equiv k-1 \pmod m,$$

then, $\mathcal{M}$ is stable $\Longleftrightarrow$ $\mathbb{B}^*$ is bounded, where

$$\mathbb{B}^* := \bigcup_{j=0}^{\infty} \mathbb{B}_j.$$

Proof. Please see Appendix A. □

Here, for any set S, the set $\mathcal{H}(S)$ denotes the *convex hull*. Formally, this is the smallest convex polyhedron which contains all the points of S [46]. Geometrically, it can be viewed as shown in Figure 5.3.

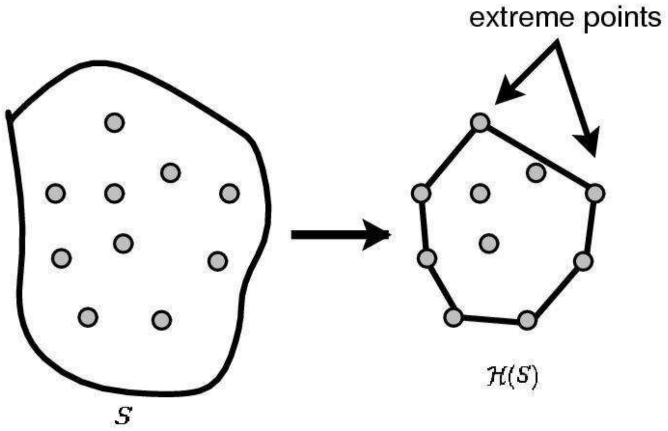

Figure 5.3: Convex hull of a set S.

Given a neighborhood of origin $\mathbb{B}_0$, two iterations of successive formula give us

$$\begin{aligned}\mathbb{B}_1 &= \mathcal{H}\left(\cup_{n_1=0}^{\infty}\mathbb{M}_0^{n_1}\mathbb{B}_0\right) = \mathcal{H}\left((\mathbb{M}_0^0\mathbb{B}_0) \cup (\mathbb{M}_0^1\mathbb{B}_0) \cup ... \cup (\mathbb{M}_0^{n_1}\mathbb{B}_0)...\right), \quad \text{and} \\ \mathbb{B}_2 &= \mathcal{H}\left((\mathbb{M}_1^0 \cup_{n_1=1}^{\infty} \mathbb{M}_0^{n_1}\mathbb{B}_0) \cup (\mathbb{M}_1^1 \cup_{n_1=1}^{\infty} \mathbb{M}_0^{n_1}\mathbb{B}_0) \cup ... \cup (\mathbb{M}_1^{n_2} \cup_{n_1=1}^{\infty} \mathbb{M}_0^{n_1}\mathbb{B}_0)...\right).\end{aligned}$$

We note that, in these iterations, computation of $\mathbb{M}_{k'}^{j}\mathbb{B}_{k-1}$ may results in some structureless sets. This drawback is overcomed in [5, 15], by regarding the convex neighborhoods $\mathbb{B}_{k-1}$ as polyhedral regions. Moreover, such regions can be represented by their finite number of extremal points. An *extreme point* of a convex set S in a real vector space is a point in S which does not lie in the open line segment joining any two different points of S. Intuitively, an extreme point is a "corner" of S (see Figure 5.3). From now on, we let the neighborhoods $\mathbb{B}_k$ be *polyhedral* regions. Those have finite number of extreme points. This finiteness will make the Theorem 5.6 constructive. Thus, we shall be allowed to focus on their extremal points for construction. We remark that our polyhedral neighborhoods $\mathbb{B}_k$ of $O = O_{2n}$ are closures of open sets; and with the boundedness of $\mathbb{B}_0$ and, by Theorem 5.10, these sets are bounded and, herewith, *polytopes* in $\mathbb{C}^{2n}$, in the case of stability.

5.4 Discrete Steps for Construction

Let $\mathcal{E}(\mathbb{B}_k)$ be the set of extreme points for $\mathbb{B}_k$ defined in Theorem 5.6. Then, it has the following property:

$$\mathcal{E}(\mathbb{B}_k) \subseteq \mathcal{E}\left(\bigcup_{j=0}^{\infty}\mathbb{M}_{k'}^{j}\mathbb{B}_{k-1}\right).$$

If $z \in \mathcal{E}(\mathbb{B}_k)$, then from the previous inclusion one can find a $j \in \mathbb{N}$ such that

$$z \in \mathcal{E}(\mathbb{M}_{k'}^{j}\mathbb{B}_{k-1}).$$

This implies that $z = \mathbb{M}_{k'}^{j} u$ with a suitable $u \in \mathcal{E}(\mathbb{B}_{k-1})$. Here, we can write the conclusion as follows.

Lemma 5.11. *[5] If z is an extremal point of $\mathbb{B}_k$ $(k > 0)$, then there exists a $j \in \mathbb{N}_0$ with*

$$z = \mathbb{M}_{k'}^{j} u,$$

where u is an extremal point of $\mathbb{B}_k$.

Thus, extremal points of the set $\mathbb{B}_{k-1}$ generate the extremal points of the set $\mathbb{B}_k$ (see Figure 5.4).

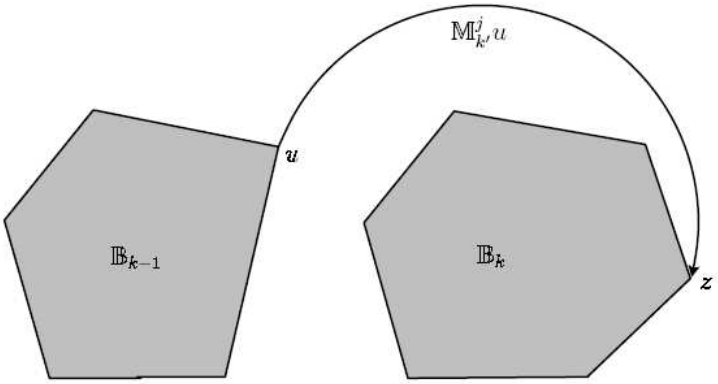

Figure 5.4: Generating new extreme points [4].

However, the nodes of the form $z = \mathbb{M}_{k'}^{j} u$ may not always be new extreme points. Thus, the following ***stopping*** criterion is applied to process:

Lemma 5.12. *Let u_i be an extremal point of $\mathbb{B}_{k-1}$ and $z_i = \mathbb{M}_{k'}^{j} u_i$ for some corresponding $j \in \mathbb{N}_0$. Then, the new polyhedral region $\mathbb{B}_k = \mathcal{H}\{z_1, z_2, ..., z_r\}$ is completely constructed, if and only if*

$$\mathbb{M}_{k'} z_i \in \mathcal{H}\{z_1, z_2, ..., z_r\} \quad (i \in \{1, 2, ..., r\}).$$

Proof. See Appendix B. □

This construction procedure is shown in Figure 5.5.

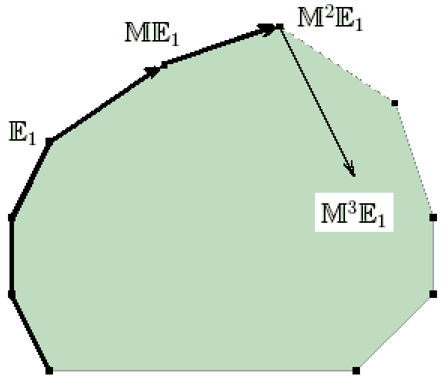

Figure 5.5: Illustration of the stopping criterion [15].

Note that in the general Theorem 5.6 of *Brayton and Tong* [5], an infinite number of regions would have to be formed iteratively but, in terms of computer applications, we need it to be finite. Lemma 5.13 guarantees us that no infinite union must be needed, but after some finite k iterations we are finished.

Lemma 5.13. *Let* $\mathbb{M}$ *be a matrix whose eigenvalues* $\lambda_j(\mathbb{M})$ *have magnitude* $|\lambda_j(\mathbb{M})| < 1$*, then, there exists a* $k \in \mathbb{N}_0$ *such that*

$$\bigcup_{j=0}^{\infty} \mathbb{M}^j \mathbb{B} = \bigcup_{j=0}^{k} \mathbb{M}^j \mathbb{B}$$

for any bounded neighborhood $\mathbb{B}$ *of the origin in* $\mathbb{R}^{2n}$*.*

Proof. Using equation (5.2.1) we write

$$\begin{aligned} \mathbb{M}^k \mathbb{B} &= PJ\overbrace{P^{-1}P}^{I_n}J\overbrace{P^{-1}P}^{I_n}...PJP^{-1}\mathbb{B} = PJ^iP^{-1}\mathbb{B}, \\ \Rightarrow \quad \|\mathbb{M}^k\| &= \|PJ^kP^{-1}\| \leq \|P\|\|J^k\|\|P^{-1}\| \leq \|J\|^k. \end{aligned}$$

Since by Lemma 5.5 and Corallary 5.6 we have

$$\begin{aligned} \|J^k\|_\infty &= \sum_{j=0}^{s-1} \binom{k}{j} |\lambda(J)|^{k-j}, \quad \text{and} \\ |\lambda_j(\mathbb{M})| &< 1, \end{aligned}$$

we write

$$\|J^k\| \to 0 \quad \text{as} \quad k \to \infty.$$

Thus, letting $v \in P^{-1}\mathbb{B}$, we can find a $k \in \mathbb{N}_0$ such that

$$\|J^k v\| < d := \min\{\|w\| : w \notin P^{-1}\mathbb{B}\}.$$

For such a k, $J^k P^{-1}\mathbb{B} \subseteq P^{-1}\mathbb{B}$ and, hence,

$$\mathbb{M}^j\mathbb{B} = P\underbrace{J^j P^{-1}\mathbb{B}}_{\subseteq P^{-1}\mathbb{B}} \subseteq PP^{-1}\mathbb{B} = \mathbb{B} \ \ \forall j \geq k.$$

This concludes the proof. □

However, it remains yet the question open when this k is reached and with which sequences. In order to answer this question, we need additional statements that presents boundedness and unboundedness criterions for the algorithm. The following lemmas indicated in [4, 5, 15] deal with boundedness of the set $\mathbb{B}^*$.

Lemma 5.14. *[5] Let*

$$\mathbb{B}_k := \mathcal{H}\left(\bigcup_{j=0}^{\infty} \mathbb{M}_{k'}^j \mathbb{B}_{k-1}\right), \quad \textit{where} \quad k' = (k-1) \quad \textit{modulo} \quad m, \quad \textit{and} \quad \mathbb{B}^* = \bigcup_{j=0}^{\infty} \mathbb{B}_j.$$

If there exists a $k \in \mathbb{N}_0$ such that

$$\partial\mathbb{B}_0 \cap \partial\mathbb{B}_k = \emptyset,$$

then $\mathbb{B}^*$ is unbounded which implies the set $\mathcal{M}$ is unstable.

Proof. See Appendix B. □

Thus, whenever we find that faces of the polyhedral regions have no common points, we stop and conclude that $\mathbb{B}^*$ is unbounded (cf. Figure 5.6).

Lemma 5.15. *We make the same assumptions as in Lemma 5.14. If for some $k \geq m$ it holds $\mathbb{B}_k = \mathbb{B}_{k-m}$, then the set $\mathbb{B}^*$ is bounded.*

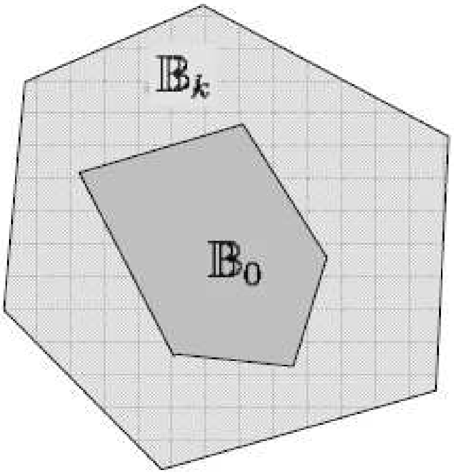

Figure 5.6: Illustration of the unboundness criterion [4].

Proof. Let us write $\mathbb{B}^*$ as

$$\mathbb{B}^* = \bigcup_{j=0}^{\infty} \mathbb{B}_j = \left(\bigcup_{j=0}^{k-m} \mathbb{B}_j \right) \cup \left(\bigcup_{j=k-m+1}^{\infty} \mathbb{B}_j \right). \tag{5.4.3}$$

We know that $\mathbb{B}_0 \subseteq \mathbb{B}_1 \subseteq ... \subseteq \mathbb{B}_{k-m} \subseteq ... \subseteq \mathbb{B}_k$. Since $\mathbb{B}_k = \mathbb{B}_{k-m}$, we have

$$\mathbb{B}_{k-m+1} = \mathbb{B}_{k-m+2} = ... = \mathbb{B}_k. \tag{5.4.4}$$

Thus, (5.4.3) is equivalent to

$$\begin{aligned} \mathbb{B}^* = \mathbb{B}_k \cup \left(\bigcup_{j=k-m+1}^{\infty} \mathbb{B}_j \right) &= \mathbb{B}_k \cup \left(\overbrace{\bigcup_{j=k-m+1}^{k} \mathbb{B}_j}^{=\mathbb{B}_k} \right) \cup \left(\bigcup_{j=k+1}^{\infty} \mathbb{B}_j \right) \\ &= \mathbb{B}_k \cup \left(\bigcup_{j=k+1}^{\infty} \mathbb{B}_j \right). \end{aligned}$$

If we can show for $j \geq k+1$, $\mathbb{B}_j = \mathbb{B}_k$, then we are done. Let us take $j = k+1$.

Then, since $k' \equiv (k-1) \ \ (mod \ \ m)$, by (5.4.4) we write

$$\mathbb{B}_{k+1} = \mathcal{H}\left(\bigcup_{j=0}^{\infty} \mathbb{M}^j_{(k+1)'}\mathbb{B}_k\right) = \mathcal{H}\left(\bigcup_{j=0}^{\infty} \mathbb{M}^j_{(k-m+1)'}\mathbb{B}_{k-m}\right) = \mathbb{B}_{k-m+1} = \mathbb{B}_k.$$

Thus, we have the desired result: $\mathbb{B}^* = \bigcup\limits_{j=0}^{\infty} \mathbb{B}_j = \mathbb{B}_k$, hence, by the boundedness of $\mathbb{B}_k$, $\mathbb{B}^*$ is bounded. □

The stability of time-continuous model $(\mathcal{CE})_{ext}$ describing gene expression profiles is strongly related with the stability of time-discrete system by the following theorem.

Theorem 5.16. *Let the map $\mathbb{E} \mapsto \mathbb{M}(\mathbb{E})$ be Lipschitzian. If time-discrete system $\mathbb{E}_{k+1} = \mathbb{E}_k + h_k\mathbb{M}(\mathbb{E}_k)\mathbb{E}_k \quad (k \in \mathbb{N}_0), \ \mathbb{E}_0 \in \mathbb{R}^{2n}$ and some appropriate $h_{max} > 0$ given, is stable for all values $h_k \in [0, h_{max}]$, then the continuous system $\dot{\mathbb{E}} = \mathbb{M}(\mathbb{E})\mathbb{E}$ is also stable.*

Proof. See [4, 5]. □

After deriving the set of matrices $\mathcal{M} = \{\mathbb{M}_0, \mathbb{M}_1, ..., \mathbb{M}_{l-1}\}$ by discretely approximating the set

$$\{\mathbb{M}(\mathbb{E}, h) | \ \ \mathbb{E} \in \mathbb{R}^{2n}, \ h \in [0, h_{max}]\},$$

where $\mathbb{M}(\mathbb{E}, h) := \mathbb{I} + h\mathbb{M}(\mathbb{E})$.

Since, however, we are using RK method, in our case, $\mathbb{M}(\mathbb{E}, h)$ takes the form:

$$\mathbb{M}(\mathbb{E}, h) := I + \frac{h}{2}\mathbb{M}(\mathbb{E}) + \frac{h}{2}\mathbb{M}(\mathbb{E} + h\mathbb{M}(\mathbb{E})\mathbb{E})(I + h\mathbb{M}(\mathbb{E})).$$

In fact, we are discretizing the function $\mathbb{M}(\mathbb{E}, h)$ in a way that the values of the implied matrix entries are taken at their maximal or minimal values, and h (by h_{max}) chosen extremally as well. When iteratively applying the resulting entire matrices to polyhedral sets, we represent and understand the worst-case growth behavior of any finite matrix multiplication, i.e., whether instability is holding.

In the next chapter, we present an implementation and a small simulation of the algorithm.

Chapter 6

Implementation of The Algorithm

6.1 Overview

We stated all prerequisites around the algorithm, derived from the construction theorem, that examine a finite set of matrices for stability.

In fact, we could refer to a neighborhood of O_{2n}, $\mathbb{B}_0$ as the closure of an open set in $2n$ dimensions. However, according to our reflections in Section 5.3 we may fix the last n components of the elements $\mathbb{E} \in \mathbb{B}_0$ and, herewith, $\mathbb{E} \in \mathbb{B}_k$, by $(1, 1, ..., 1)^T$. Then, our sets $\mathbb{B}_0$ and $\mathbb{B}_k$ become neighborhoods of O only in the first n components, and they are neighborhoods of $(O^T, e^T)^T$ in $\mathbb{C}^n \times \{e\}$, now.

Given a polyhedral region, a bounded neighborhood of the origin, $\mathbb{B}_0$, we iteratively construct the set of of extremal points as mentioned Section 5.4. After each step, the stopping and boundedness criterions are checked.

Through the numerical procedure for constructing the algorithm the absolute value of all eigenvalues of the matrices are required to be less then 1 and in case of equality algebraic multiplicity and geometric multiplicity are required to coincide.

Now, we present and follow the structure of modified *Brayton and Tong*'s algorithm done in [15]. Regarding our extended space now, we write the algorithm step by step similar to one presented in [4].

6.2 The Steps of the Algorithm

STEP 0: Initializing

```
set $\mathcal{E}(\mathbb{B}_0) := \{x \in \mathbb{R}^{2n} : x = (x_1, x_2, ..., x_n, 1, 1..., 1)^T, \forall i, \|x_i\| = 1\}$
set $k := 0$
set $l := 0$
```

We begin with initializing the set of extreme points $\mathcal{E}(\mathbb{B}_0)$. In practice, in our extended space, $\mathbb{B}_0$ is usually selected as simple as possible, i.e., it is chosen as the region defined by above.

Note that $\mathcal{E}(\mathbb{B}_0)$ is determined by these totally 2^n extreme points.

STEP 1: Formation of new set of extreme points

```
set $j := 0$
set $flag := 0$
set $TEMP := \mathcal{E}(\mathbb{B}_k)$
set $V := \mathcal{E}(\mathbb{B}_0) \cap \mathcal{E}(\mathbb{B}_{max\{k-m,0\}})$
          if $V = TEMP$ and $k \geq m$ exit *stable*
          otherwise go to Step 2.
```

At the beginning of each step, we check whether the newly constructed set of extreme points fulfills the condition stated in Lemma 5.15. Here, m stands for the number of matrices in the set $\mathcal{M}$.

STEP 2

```
set $POINT := Point\ j$ of $\mathcal{E}(\mathbb{B}_k)$
if $POINT \in \mathcal{E}(\mathbb{B}_0)$ and $POINT \in \mathcal{H}(TEMP)$
or $POINT \notin \mathcal{E}(\mathbb{B}_0)$ and $POINT \in \mathcal{H}(TEMP - POINT)$
                    set $TEMP := TEMP \setminus P0INT$
                    go to Step 4
otherwise $flag = 0$
if $POINT \in V$ and $k \geq m$
                    go to Step 4
otherwise $NewPOINT := POINT$
go to STEP 3.
```

In this step, for each of the points is controlled whether this point strictly lies in the convex hull of the set $TEMP$ or not. If it is contained in V, then we drop it from V and look for whether the condition stated in Lemma 5.10 is satisfied or not. By generating a new point we continue the algorithm.

```
STEP 3
    set $NewPOINT := \mathbb{M}_l NewPOINT$
    set $flag := 0$
        if $NewPOINT \notin \mathcal{H}(TEMP)$
            set $TEMP := TEMP \cup NewPOINT$
            set $flag := 1$
        otherwise go to Step 4.
```

Note that new points are obtained by matrix multiplication.

```
STEP 4
    if $\{\mathcal{E}(\mathbb{B}_0) \cap TEMP\} = \emptyset$
        exit unstable
    otherwise:
        if $flag = 1$
        go to Step 3
        otherwise set $j := j + 1$
        if $j \leq |\mathcal{E}(\mathbb{B}_k)|$
        go to Step 2
        else set $l := l + 1\ (mod\ m)$
        set $k := k + 1$
        set $\mathcal{E}(\mathbb{B}_k) := TEMP$
        go to Step 1.
```

6.3 An Example

For a better understanding of the algorithm we present our example:

Example 6.1. Let us consider the set $\mathcal{M} = \{\mathbb{M}_0, \mathbb{M}_1\}$ ($m = 2$) where

$$\mathbb{M}_0 = \begin{pmatrix} 0.6 & 1 & -0.2 & 0 \\ -0.4 & 0 & 0 & 0.1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad , \quad \mathbb{M}_1 = \begin{pmatrix} 0.1 & 0 & 0.15 & 0 \\ 0.13 & 0.5 & 0 & 0.22 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (n = 2).$$

Initializing with respect to *Step 0* gives:

$$\begin{aligned}
\mathcal{E}(\mathbb{B}_0) &= \{\begin{pmatrix}1\\1\\1\\1\end{pmatrix},\begin{pmatrix}-1\\-1\\1\\1\end{pmatrix},\begin{pmatrix}1\\-1\\1\\1\end{pmatrix},\begin{pmatrix}-1\\1\\1\\1\end{pmatrix}\},\\
k &= 0,\\
l &= 0.
\end{aligned}$$

Now, *Step 1* performs the following:

$$\begin{aligned}
j &= 1,\\
flag &= 0,\\
TEMP &= \{\begin{pmatrix}1\\1\\1\\1\end{pmatrix},\begin{pmatrix}-1\\-1\\1\\1\end{pmatrix},\begin{pmatrix}1\\-1\\1\\1\end{pmatrix},\begin{pmatrix}-1\\1\\1\\1\end{pmatrix}\},\\
V &= \{\begin{pmatrix}1\\1\\1\\1\end{pmatrix},\begin{pmatrix}-1\\-1\\1\\1\end{pmatrix},\begin{pmatrix}1\\-1\\1\\1\end{pmatrix},\begin{pmatrix}-1\\1\\1\\1\end{pmatrix}\}.
\end{aligned}$$

Since $m = 2$, $V = \mathcal{E}(\mathbb{B}_0) \cap \mathcal{E}(\mathbb{B}_{max\{-2,0\}}) = \mathcal{E}(\mathbb{B}_0) = TEMP$, but the condition $k \geq m$ does not hold. Thus, we pass the *Step 3*:
$POINT = (1,1,1,1,1)^T \in \mathcal{E}(\mathbb{B}_0)$ and $POINT \notin \mathcal{H}(TEMP - POINT)$, but

$POINT \in V$ implies

$$\begin{aligned} flag &= 1, \\ NewPOINT &= \mathbb{M}_0 POINT = \begin{pmatrix} 1.4 \\ -0.3 \\ 1 \\ 1 \end{pmatrix}, \\ flag &= 0. \end{aligned}$$

The following linear program is used for deciding whether $NewPOINT$ is contained in the set $\mathcal{H}(TEMP)$ or not: If $P \in \mathcal{H}\{P_1, P_2, ..., P_n\}$

$$\text{minimize } 0 \text{ subject to}$$
$$P = \sum_{i=1}^{n} \lambda_i P_i, \ \lambda_i \geq 0 \quad \text{and} \quad \sum_{i=1}^{n} \lambda_i = 1.$$

For our example, it is clear that $NewPOINT \notin \mathcal{H}(TEMP)$. Hence, we extend the set by adding this new point. In other words,

$$TEMP = \{ \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1.4 \\ -0.3 \\ 1 \\ 1 \end{pmatrix} \} \text{ and } \ flag = 1.$$

Generating new points once more, we get

$$\begin{aligned} NewPOINT := \mathbb{M}_0 NewPOINT &= \begin{pmatrix} 0.6 & 0.1 & -0.2 & 0 \\ -0.4 & 0 & 0 & 0.1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1.4 \\ -0.3 \\ 1 \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} 0.34 \\ -0.46 \\ 1 \\ 1 \end{pmatrix} \in \mathcal{H}(TEMP). \end{aligned}$$

By *Step 4*, we set $j = 2$ and again return to *Step 2* and take $POINT = 1, -1, 1, 1)^T$. Corresponding new point is

$$NewPOINT = \mathbb{M}_0 POINT = (-1.8, 0.5, 1, 1).$$

Similar to the previous case, there is no more points to be added. So we arrive at the set

$$TEMP = \{ \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1.4 \\ -0.3 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1.8 \\ -0.5 \\ 1 \\ 1 \end{pmatrix} \},$$

and $flag = 1$. We can easily verify that the third ($j = 3$) and fourth ($j = 4$) points of the set $TEMP$ cannot generate new extreme points. When $j = 5$, being larger than the number of points in $\mathcal{E}(\mathbb{B}_0)$, we write

$$\begin{aligned}
l &= 1\ (mod\ 2),\\
k &= 1,\\
\mathcal{E}(\mathbb{B}_1) &= \{\begin{pmatrix}1\\1\\1\\1\end{pmatrix},\begin{pmatrix}-1\\-1\\1\\1\end{pmatrix},\begin{pmatrix}1\\-1\\1\\1\end{pmatrix},\begin{pmatrix}-1\\1\\1\\1\end{pmatrix},\begin{pmatrix}1.4\\-0.3\\1\\1\end{pmatrix},\begin{pmatrix}1.8\\-0.5\\1\\1\end{pmatrix}\},\\
j &= 1,\\
flag &= 0,\\
TEMP &= \mathcal{E}(\mathbb{B}_1),\\
V &= \mathcal{E}(\mathbb{B}_1)\cap\mathcal{E}(\mathbb{B}_{max\{0,1-2\}}) = \mathcal{E}(\mathbb{B}_0).
\end{aligned}$$

Here, we see that our task with matrix $\mathbb{M}_0$ has been done. We repeat the same steps for the points of the new set $\mathcal{E}(\mathbb{B}_1)$. However, it is not possible to get new extreme point from the set $\mathcal{E}(\mathbb{B}_1)$ by matrix multiplication with $\mathbb{M}_1$. Thus, we end up with

$$\begin{aligned}
l &= 0\ (mod\ 2),\\
k &= 2,\\
\mathcal{E}(\mathbb{B}_2) &= \{\begin{pmatrix}1\\1\\1\\1\end{pmatrix},\begin{pmatrix}-1\\-1\\1\\1\end{pmatrix},\begin{pmatrix}1\\-1\\1\\1\end{pmatrix},\begin{pmatrix}-1\\1\\1\\1\end{pmatrix},\begin{pmatrix}1.4\\-0.3\\1\\1\end{pmatrix},\begin{pmatrix}1.8\\-0.5\\1\\1\end{pmatrix}\},\\
j &= 1,\\
flag &= 0,\\
TEMP &= \mathcal{E}(\mathbb{B}_2) = \mathcal{E}(\mathbb{B}_1),\\
V &= \mathcal{E}(\mathbb{B}_2)\cap\mathcal{E}(\mathbb{B}_{max\{0,2-2\}}) = \mathcal{E}(\mathbb{B}_1)\cap\mathcal{E}(\mathbb{B}_0) = \mathcal{E}(\mathbb{B}_0).
\end{aligned}$$

This part of the iteration defines $TEMP := \mathcal{E}(\mathbb{B}_3) = \mathcal{E}(\mathbb{B}_1)$. Therefore, if $k = 3$, then $V = \mathcal{E}(\mathbb{B}_3)\cap\mathcal{E}(\mathbb{B}_{max\{0,3-2\}}) = \mathcal{E}(\mathbb{B}_1) = TEMP$. We reach a stopping criterion in terms of boundedness as stated in Lemma 5.11. According to our simulation result, we finally conclude that the set $\mathcal{M} = \{\mathbb{M}_0, \mathbb{M}_1\}$ is ***stable***. This implies that, by Theorem 5.12, the corresponding time-continuous model is also stable.

We note that applications on real data require much more computational efforts because of parametric nonlinearities we have. However, nonlinear least-squares method can be applied to overcome this problem.

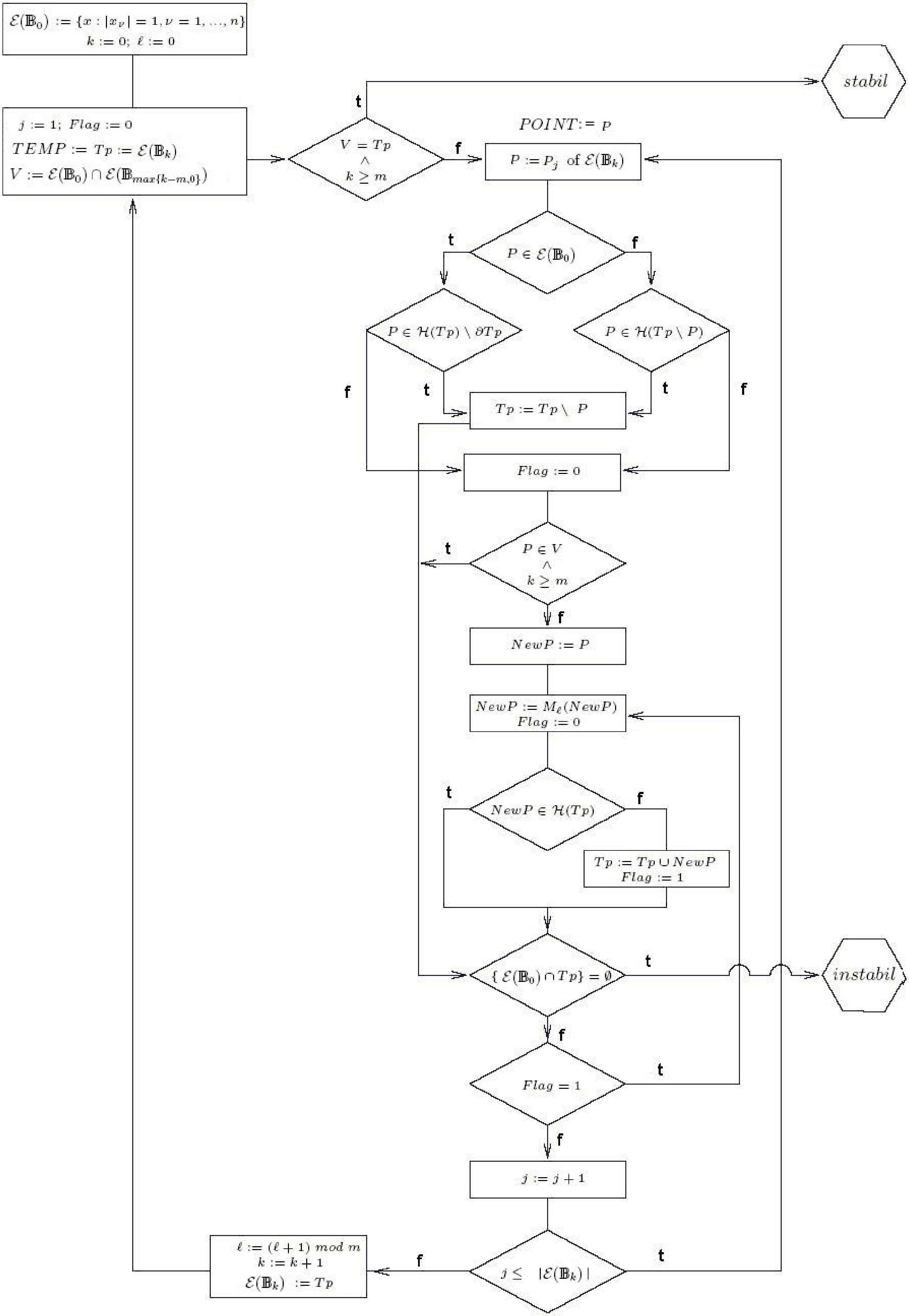
$\mathcal{E}(\mathbb{B}_0) := \{x : |x_\nu| = 1, \nu = 1, \dots, n\}$
$k := 0;\ \ell := 0$
$j := 1;\ Flag := 0$
$TEMP := Tp := \mathcal{E}(\mathbb{B}_k)$
$V := \mathcal{E}(\mathbb{B}_0) \cap \mathcal{E}(\mathbb{B}_{max\{k-m,0\}})$
t
$V = Tp$ $\wedge$ $k \geq m$
f
$stabil$
$POINT := P$
$P := P_j$ of $\mathcal{E}(\mathbb{B}_k)$
t
$P \in \mathcal{E}(\mathbb{B}_0)$
f
$P \in \mathcal{H}(Tp) \setminus \partial Tp$
$P \in \mathcal{H}(Tp \setminus P)$
f
t
t
f
$Tp := Tp \setminus P$
$Flag := 0$
t
$P \in V$ $\wedge$ $k \geq m$
f
$NewP := P$
$NewP := M_\ell(NewP)$
$Flag := 0$
t
$NewP \in \mathcal{H}(Tp)$
f
$Tp := Tp \cup NewP$
$Flag := 1$
$\{\ \mathcal{E}(\mathbb{B}_0) \cap Tp\} = \emptyset$
t
$instabil$
f
$Flag = 1$
t
f
$j := j + 1$
$\ell := (\ell + 1)\ mod\ m$
$k := k + 1$
$\mathcal{E}(\mathbb{B}_k)\ := Tp$
f
$j \leq\ |\mathcal{E}(\mathbb{B}_k)|$
t

Figure 6.1: Structure chart of modified Brayton's and Tong's algorithm [4].

Chapter 7

Conclusion and Future Work

This thesis is a part of advanced studies in the modern and emerging field of modeling, analysis and prediction based on DNA microarray experiments. Objects of these studies are DNA patterns investigated in their time-dependence and prediction, and, dynamical representations. In this field, at IAM of METU and with its international colleagues, a lot of knowledge and scientific experience has been gathered in the last years. The content of this work has been continuation of these studies, whereby methods of dynamical systems theory, of discrete mathematics and of statistical learning are rigorously applied to gain further insights.

In this study, from the viewpoint of dynamical system and for making models more realistic, approximative and better prepared for the purpose of forecasting, we improved the mathematical model and stability analysis. Here, we introduced affine shift terms, and we investigated the system by considering advantages of the Runge-Kutta methods. Thus, our study may help these techniques to achieve new insights by mathematical modeling, dynamical systems, optimization and algorithms. A modified powerful algorithmic stability analysis tool, *Brayton and Tong's algorithm* [5], has been used to detect stability or instability in combinatorial and geometrical terms.

Beside the stability analysis, models on gene expression patterns require also sensitivity analysis. There are several methods to do this. For example in [58], sensitivity analysis is applied with differential algebraic equation methods. For the future, it is our intention to combine both approaches which makes it algorithmically possible to determine regions of stability and instability.

Appendix A

Proof of Brayton and Tong's Theorem

Proof. [5] $' \Leftarrow'$: Let us assume that $\mathbb{B}^*$ is bounded. To show that $\mathcal{M}$ is stable we take $U = \mathbb{B}^*$ (see Definition 3). Let $\mathbb{M} \in \mathcal{M}'$ be in the form of $\mathbb{M}_{j_1}^{k_1}\mathbb{M}_{j_2}^{k_2}...\mathbb{M}_{j_n}^{k_n}$ where $\forall i, k_i \neq 0$ and $j_i \neq j_{i+1}$. Our aim is to show for any $\mathbb{M} \in \mathcal{M}'$, $\mathbb{M}\mathbb{B}^* \in \mathbb{B}^*$. In other words, for any $z \in \mathbb{B}^*$ we have $\mathbb{M}z \in \mathbb{B}^*$.

Here, $z \in \mathbb{B}^*$ implies $z \in \mathbb{B}_{i_0}$ for some i_0. Since $\mathbb{B}_0 \subseteq \mathbb{B}_1 \subseteq \mathbb{B}_2 \subseteq ... \subseteq \mathbb{B}^*$, $z \in \mathbb{B}_{i_0+1}, \mathbb{B}_{i_0+2}, ..., \mathbb{B}_{i_0+m}$. Remembering

$$\mathbb{B}_i := \mathcal{H}\left(\bigcup_{i=0}^{\infty}\mathbb{M}_{k'}^{i}\mathbb{B}_{k-1}\right), \quad \text{where} \quad k' = (k-1) \text{ modulo } m,$$

we can conclude $\mathbb{M}_{k'}^{i}z \in \mathbb{B}^*$ for $k' = 0, 1, 2, ..., m-1$. By induction,

- for the case $n = 1$, $\mathbb{M}z = \mathbb{M}_{k'}^{i}z \in \mathbb{B}^*$;
- assume that it holds for $n = r$, $\mathbb{M}z = \mathbb{M}_{j_1}^{k_1}\mathbb{M}_{j_2}^{k_2}...\mathbb{M}_{j_r}^{k_r}z \in \mathbb{B}^*$;
- take $n = r + 1$,

$$\begin{aligned} \mathbb{M}z &= (\mathbb{M}_{j_1}^{k_1}\mathbb{M}_{j_2}^{k_2}...\mathbb{M}_{j_r}^{k_r})(\mathbb{M}_{j_{r+1}}^{k_{r+1}}z) \\ &= (\mathbb{M}_{j_1}^{k_1}\mathbb{M}_{j_2}^{k_2}...\mathbb{M}_{j_r}^{k_r})z' \in \mathbb{B}^*. \end{aligned}$$

Since $\mathbb{B}^*$ is bounded, by our Lemma 5.4, $\mathcal{M}$ is stable.

$' \Rightarrow'$: Suppose $\mathcal{M}$ is stable. Then, there exists a bounded neighborhood of the origin $\mathbb{B}$ such that for each $\mathbb{M} \in \mathcal{M}'$, $\mathbb{M}\mathbb{B} \subseteq \mathbb{B}$. Since $\mathbb{B}_0$ is also bounded we can find $\rho > 0$ so that $\rho\mathbb{B}_0 \subseteq B$. By induction, we proceed as follows:

- assume that for $k \in \mathbb{N}$, $\rho\mathbb{B}_k \subseteq \mathbb{B}$;
- consider the $(k+1)^{th}$ case:

$$\begin{aligned} \rho\mathbb{B}_{k+1} &= \rho\mathcal{H}\left(\bigcup_{i=0}^{\infty} \mathbb{M}_{k'}^{i}\mathbb{B}_k\right) \subseteq \mathcal{H}\left(\rho\bigcup_{i=0}^{\infty} \mathbb{M}_{k'}^{i}\mathbb{B}_k\right) \\ &= \mathcal{H}\left(\bigcup_{i=0}^{\infty} \rho\mathbb{M}_{k'}^{i}\mathbb{B}_k\right) = \mathcal{H}\left(\bigcup_{i=0}^{\infty} \mathbb{M}_{k'}^{i}\underbrace{(\rho\mathbb{B}_k)}_{\subseteq B}\right) \\ &\subseteq \mathbb{B}. \end{aligned}$$

So, we conclude

$$\rho\mathbb{B}^* = \rho\bigcup_{i=0}^{\infty} \mathbb{B}_i = \bigcup_{i=0}^{\infty}(\rho\mathbb{B}_i) \subseteq \mathbb{B},$$

which implies that $\mathbb{B}^*$ is bounded. □

Appendix B

Proofs of Lemma 5.12 and Lemma 5.14

Proof of Lemma 5.12

Proof. [4, 5] $' :\Rightarrow'$ Suppose $\mathbb{B}_k = \mathcal{H}\{z_1, z_2, ..., z_r\}$. By the construction theorem we have $\mathbb{B}_k := \mathcal{H}\left(\bigcup\limits_{i=0}^{\infty} \mathbb{M}_{k'}^{i}\mathbb{B}_{k-1}\right)$. Thus, $\mathbb{M}_{k'}z_i = \mathbb{M}_{k'}\mathbb{M}_{k'}^{j}u_i = \mathbb{M}_{k'}^{j+1}u_i \in \mathbb{B}_k$

$' \Leftarrow:'$ Let us define $\mathcal{U} := \mathcal{H}\{z_1, z_2, ..., z_r\}$.

"$\mathbb{B}_k \subseteq \mathcal{U}$": If for all i, $\mathbb{M}_{k'}z_i \in \mathcal{U}$, then $\mathbb{M}_{k'}\mathcal{U} \subseteq \mathcal{U}$. Thus $\mathbb{M}_{k'}^{j}\mathcal{U} \subseteq \mathcal{U}$ and hence $\mathbb{B}_k := \mathcal{H}\left(\bigcup\limits_{i=0}^{\infty} \mathbb{M}_{k'}^{i}\mathbb{B}_{k-1}\right) \subseteq \mathcal{U}$.

"$\mathbb{B}_k \supseteq \mathcal{U}$": Assume that there exists a point $p \in \mathcal{U}$ but $p \notin \mathbb{B}_k$. Then there must be an extreme point of $\mathcal{U}$ not belonging to $\mathbb{B}_k$. However, this is a contradiction because the extreme points of $\mathcal{U}$ having the form $\mathbb{M}_{k'}^{j}u_i$ where $u_i \in \mathcal{E}(\mathbb{B}_k)$, must be in $\mathbb{B}_k$.

Therefore, $\mathcal{U} = \mathbb{B}_k$. □

Proof of Lemma 5.14

Proof. [4, 5] Suppose $\mathbb{B}^* := \bigcup\limits_{i=0}^{\infty} \mathbb{B}_i$ is bounded. Then by Brayton and Tong's Theorem $\mathcal{M}$ is stable and by Lemma 5.4 there is a bounded convex neighborhood of the origin $B \subseteq \mathbb{C}$ such that

$$\mathbb{M}B \subseteq B \text{ for all } \mathbb{M} \in \mathcal{M}.$$

Let us choose a suitable constant $\rho > 0$ so that $\partial\mathbb{B}_0 \cap \partial\mathbb{B}_k \neq \emptyset$ and $\rho\mathbb{B}_0 \subseteq B$. For any matrix $\mathbb{M}$ and any set of points P,

$$\mathbb{M}\mathcal{H}\{P\} = \mathcal{H}\{\mathbb{M}P\}.$$

$\square$

References

[1] Akhmet, M.U., Gebert, J., Öktem, H., Pickl, S.W., and Weber, G.-W., An improved method for analytical modeling and anticipation of gene expression patterns, to appear in Journal of Computational Technologies 10, 4 (2005).

[2] Akutsu, T., Miyano, S., and Kuhara, S., Identification of genetic networks from a small number of gene expression patterns under the Boolean network model, Pacific Symposium on Biocomputing 4 (1999) 17-28.

[3] Atherly, G.A., Girton, J.R., and McDonald, J.F., *The Science of Genetics*, Saunders College Publishing, 1999.

[4] Bechmann, E., *Analyse und Konstruktion eines Algorithmus zur Untersuchung der Stabilität dynamischer Systeme*, Mathematischen Institut der Universität zu Köln, diploma thesis 2005.

[5] Brayton, R.K., and Tong, C.H., Stability of dynamical systems: A constructive approach, IEEE Transactions on Circuits and Systems 26, 4 (1979) 224-234.

[6] Carbayo, M.S, Bornman, W., and Cardo C.C., DNA microchips: Technical and practical considerations, Current Organic Chemistry 4, 9 (2000) 945-971.

[7] Causton, C.H., Quackenbush, J., and Brazma, A., *A Beginner's Guide Microarray Gene Expression Data Analysis*, Blackwell Publishing, 2003.

[8] Chen, T., and He, H.L., Modeling gene expression with differential equations, Pacific Symposium on Biocomputing 4 (1999) 29-40.

[9] Chen, L., and Aihara, K., Stability of gene regulatory networks with time delays, IEEE Trans. Circuits Syst I., 49 (2002) 602-608.

[10] Dubois, D.M., and Kalisz, E., Precision and stability of Euler, Runga-Kutta and incursive algorithm for the harmonic oscillator, International Journal of Computing Anticipatory Systems 14 (2004) 21-36.

[11] Edwards, R., and Glass, L., Combinatorial explosion in model gene networks, Chaos 10 (2000) 691-704.

[12] Erb, R.S., and Michaels, G.S., Sensitivity of biological models to errors in parameter estimates, Pacific Symposium on Biocomputing 4 (1999) 53-64.

[13] Ergenç, T., and Weber, G.-W., Modeling and prediction of gene-expression patterns reconsidered with Runge-Kutta discretization, special issue at the occasion of seventith birthday of Prof. Dr. Karl Roesner, TU Darmstadt, of Journals of Computational Technologies 9, 6 (2004) 40-48.

[14] Friedman, N., Linial, M., Nachman, I., and Peer, D., Using Bayesian networks to analyze expression data, J. Comput. Biol. 7, 3-4 (2000) 601-620.

[15] Gebert, J., Lätsch, M., Pickl, S. W., Weber, G.-W., and Wünschiers, R., An algorithm to analyze stability of gene-expression pattern, Discrete Mathematics and Data Mining of Discrete Applied Mathematics, accepted for publication (2005).

[16] Gebert, J., Lätsch, M., Pickl, S.W., Weber, G.-W., and Wünschiers, R., Genetic networks and anticipation of gene expression patterns, *Computing Anticipatory Systems: CASYS03 - Sixth International Conference*, AIP Conference Proceedings 718 (2004) 474-485.

[17] Gebert, J., Radde, N., and Weber, G.W., Modeling gene regulatory networks with piecewise linear differential equations, preprint, Middle East Technical University, Institute of Applied Mathematics (2005).

[18] Gebert, J., Pickl, S.W., Radde, N., Weber, G.W., Yılmaz, F.B., and Öktem, H., Inference of gene expression patterns by using a hybrid system formulation - an algorithmic approach to local state transition matrices, to appear in the proceedings of IIAS (International Institute for Advanced

Studies), 16th International Symposium for Anticipatory and Predictive Systems, Baden Baden (2004).

[19] Gardner, J.E., Simmons, J.M., and Snustad, D.P, *Principles of Genetics*, John Willey and Sons, 1991.

[20] Glass, L., Classification of biological networks by their qualitative dynamics, J.Ther. Biol. 54 (1975) 85-107.

[21] Guckenheimer, J., and Holmes, P., *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*, Springer, 1997.

[22] Hahn, W., *Theory and Application of Liapunov's Direct Method*, Prentice-Hall, 1963.

[23] Hartwell, L.H., Leroy, H., Goldberg, L.M., Reynolds, A.E., Silver, L.M., and Veres, R.C., *Genetics: From Genes to Genome*, McGraw-Hill, 2000.

[24] Hartl, D.L., and Jones E.W., *Essential Genetics*, Jones and Bartlett Publishers, 2002.

[25] Hastie, T., Tibshirani, R., and Friedman, J., *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, Springer, 2001.

[26] Heath, M., *Scientific Computing: An Introductory Survey*, McGraw-Hill, 2002.

[27] Heckerman, D., A tutorial on learning with Bayesian networks, in: M.I. Jordan, ed., *Learning in Graphical Models*, Kluwer, 1998.

[28] Hirsch, M.W., and Smale, S., *Differential Equations, Dynamical Systems and Linear Algebra*, Academic Press, 1974.

[29] Hoon, M.D., Imoto, S., Kobayashi, K., Ogasawara, N. and Miyano, S., Inferring gene regulatory networks from time-ordered gene expression data of Bacillus subtilis using differential equations, Proc. Pacific Symposium on Biocomputing, (2003) 17-28.

[30] Jong, H.D., Modeling and simulation of genetic regulatory systems: A literature review, J. Comput. Biol. 9 (2002) 103-129.

[31] Jongen, H.T., and Weber, G.-W., On parametric nonlinear programming, Annals of Operations Research 27 (1990) 253-284.

[32] Jordan, D.W., and Smith, P., *Nonlinear Ordinary Differential Equations*, Clarendon Press, 1977.

[33] Kauffman, S.A, *Origins of order: Self-Organization and Selection in Evolution*, Oxford University Press, 1993.

[34] Klug, W.S., and Cummings, M.R., *Concepts of Genetics*, Prentice Hall, 2003.

[35] Koç, C., *Topics in Linear Algebra*, MV METU 2000.

[36] Kot, M., *Elements of Mathematical Ecology*, Cambridge University Press, 2001.

[37] Lee, M.T., *Analysis of Microarray Gene Expression Data*, Kluwer Academic Publishers, 2004.

[38] Lewin, B., *Genes VII,* Oxford University Press, 2000.

[39] Liang, S., Fuhrman, S., and Brown P.O., REVEAL, a general reverse engineering algorithm for inference of genetic network architectures, Pacific Symposium on Biocomputing 98 (1998) 18-29.

[40] Mestl, T., Lemay, C. and Glass, L., Chaos in high-dimensional neural and gene networks, Physica D, 98 (1996) 33-52.

[41] Miller, R.K., and Michel, A.N., *Ordinary Differential Equations*, Academic Press, 1982.

[42] Murphy, K., and Mian, S., Modeling gene expression data using dynamic Bayesian networks, technical report, University of California at Berkeley, Berkeley, CA. (unpublished report, 1999).

[43] National Human Genome Research Institute, Division of Intramural Research, http://www.genome.gov/11508889.

[44] Oppenheim, A.V., *Signals and Systems*, Prentice-Hall, 1983.

[45] Öktem, H., Pearson, P., and Egiazarian, K. An adjustable aperiodic model class of genomic interactions using continuous time Boolean networks (Boolean delay equations), Chaos 13 (2003) 1167-1174.

[46] Rockafellar, R.T., *Convex Analysis*, Princeton University Press, 1970.

[47] Sakamoto, E., and Iba, H., Inferring a system of differential equations for a gene regulatory network by using genetic programming, Proc. Congress on Evolutionary Computation (2001) 720-726.

[48] Schena, M., *DNA Microarrays*, Oxford University Press, 2000.

[49] Sinanoglu, O., 50 Yıllık Moleküler Biyolojinin Fizik, Kimya Temelleri Sarsılırken Şimdi Yeni Ufuklara Doğru: Hayatın Örgüsü, conference talk held at METU, Ankara, Turkey (2005).

[50] Spellman, P., Sherlock, G., Zhang, M., Iyer, V., Anders, K., Eisen, M., Brown, P., Botstein, D., and Futcher, B., Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization, Mol. Biol. Cell 9 (1998) 3273-3297.

[51] Spirtes, P., Glymour, C., and Scheines, R., *Causation, Prediction, and Search*, Springer-Verlag, New York, 1993.

[52] Stanford microarray database, http://genome-www5.stanford.edu.

[53] Suhai, S., *Genomics and Proteomics: Functional and Computational Aspects*, Kluwer Academics, 2000.

[54] Taştan, M., Pickl, S.W., and Weber, G.-W., Challenges in the optimization of biosystems II: Mathematical modeling and stability analysis of

gene-expression patterns in an extended space and with Runge-Kutta discretization, preprint, Institute of Applied Mathematics, METU, submitted to proceedinds of International Scientific Annual Conference ”Operations Research Proceedings 2005”, Bremen, Germany, 7-9 September.

[55] Taştan, M., Ergenç, T., Pickl, S.W., and Weber, G.-W., Stability analysis of gene expression patterns by dynamical systems and by a combinatorial algorithm, to appear in International Symposium on Health Informatics and Bioinformatics Proceedings 2005, Antalya, Turkey, 10-12 November.

[56] Thomas, R., and Kauffman, M., Multistationary, the basis of cell differentiation and memory, I. Structural conditions of multistationary and other nontrivial behavior, Chaos 11 (2001) 170-179.

[57] Turing, A.M., The chemical basis of morphogenesis, Proceedings of the Royal Philosophical Society of London, Series B: Biological Sciences 237 (1952) 37-72.

[58] Weber, G.-W., Özoğur, S., and Karasözen, B., Challanges in the optimization of biosystems I: parameter estimation of enzymatic reactions with genetic algorithm, preprint, Institute of Applied Mathematics, METU, submitted to proceedinds of International Scientific Annual Conference ”Operations Research Proceedings 2005”, Bremen, Germany, 7-9 September.

[59] Woese, C.R., *The Genetic Code; The Molecular Basis for Genetic Expression*, Harper and Row, 1967.

[60] Wuensche, A., Genomic regulation modeled as a network with basins of attraction, Pacific Symposium on Biocomputing 3 (1998) 89-102.

[61] Vierstraete, A., http://users.ugent.be/ avierstr/

[62] Yamamoto, K.R., Steroid receptor regulated transcription of specific genes and gene networks, Ann. Rev. Genetics. 19 (1985) 209-252.

[63] Yılmaz, F.B., *A Mathemeatical Modeling and Approximation of Gene Expression Patterns by Linear and Quadratic Regulatory Relations and Analysis of Gene Networks*, Institute of Applied Mathematics, METU, MSc Thesis (2004).

[64] Yılmaz, F.B., Öktem, H., and Weber, G.-W., Mathematical modeling and approximation of gene expression patterns and gene networks, Operations Research Proceedings at the occasion of International Conference on Operations Research, Tilburg, The Netherlands, September 2004, eds.: F. Fleuren, D. den Hertog and P. Kort (2005) 280-287.