2018

Dynamic modeling and analysis of worm data via linear operator

Zhelun Cao
Iowa State University

Follow this and additional works at: https://lib.dr.iastate.edu/etd

Part of the Electrical and Electronics Commons

Recommended Citation
Cao, Zhelun, "Dynamic modeling and analysis of worm data via linear operator" (2018). Graduate Theses and Dissertations. 17152.
https://lib.dr.iastate.edu/etd/17152

This Thesis is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Graduate Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.
Dynamic modeling and analysis of worm data via linear operator

by

Zhelun Cao

A thesis submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Electrical Engineering (Systems and Controls)

Program of Study Committee:
Umesh Vaidya, Major Professor
Sourabh Bhattacharya
Pandey Santosh

The student author, whose presentation of the scholarship herein was approved by the program of study committee, is solely responsible for the content of this thesis. The Graduate College will ensure this thesis is globally accessible and will not permit alterations after a degree is conferred.

Iowa State University

Ames, Iowa

2018

Copyright © Zhelun Cao, 2018. All rights reserved.
DEDICATION

I would like to dedicate this thesis to my father Changhong Cao, my mother Guofang Chen and my girlfriend Teresa Tian. Thank you for the accompany and support. I would not make it so far without you.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIST OF FIGURES</td>
<td>v</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>vi</td>
</tr>
<tr>
<td>NOMENCLATURE</td>
<td>vii</td>
</tr>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>viii</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>ix</td>
</tr>
<tr>
<td>CHAPTER 1. INTRODUCTION: THESIS FORMATTING</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Background</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Research Motivation</td>
<td>2</td>
</tr>
<tr>
<td>1.3 Thesis Organization</td>
<td>2</td>
</tr>
<tr>
<td>CHAPTER 2. REVIEW OF LITERATURE</td>
<td>4</td>
</tr>
<tr>
<td>CHAPTER 3. DATA-DRIVEN APPROXIMATION OF LINEAR OPERATOR</td>
<td>7</td>
</tr>
<tr>
<td>3.1 Linear Operator</td>
<td>7</td>
</tr>
<tr>
<td>3.1.2 The Koopman Operator</td>
<td>7</td>
</tr>
<tr>
<td>3.2 DMD And EDMD Approaches</td>
<td>9</td>
</tr>
<tr>
<td>3.2.1 EDMD</td>
<td>10</td>
</tr>
<tr>
<td>3.2.2 DMD</td>
<td>11</td>
</tr>
<tr>
<td>3.2.4 Dictionary Functions</td>
<td>12</td>
</tr>
<tr>
<td>3.2.4.1 Polynomial Basis</td>
<td>12</td>
</tr>
<tr>
<td>3.2.4.2 Gaussian Basis</td>
<td>12</td>
</tr>
<tr>
<td>3.3 Robust Computation of Koopman Operator</td>
<td>12</td>
</tr>
<tr>
<td>3.3.1 Robust approximation of Koopman operator</td>
<td>13</td>
</tr>
<tr>
<td>3.4 Design of Predictor</td>
<td>14</td>
</tr>
<tr>
<td>CHAPTER 4. SIMULATION RESULTS</td>
<td>16</td>
</tr>
<tr>
<td>4.1 Preprocess</td>
<td>16</td>
</tr>
<tr>
<td>4.1.1 Image process</td>
<td>16</td>
</tr>
<tr>
<td>4.1.2 Midline Evolution</td>
<td>17</td>
</tr>
<tr>
<td>4.1.3 Data manipulation</td>
<td>17</td>
</tr>
<tr>
<td>4.2 Worm dynamic system decomposition and prediction</td>
<td>18</td>
</tr>
<tr>
<td>4.2.1 Robust DMD and Robust EDMD with Polynomial Basis</td>
<td>19</td>
</tr>
<tr>
<td>4.2.1.1 Worm Decomposition</td>
<td>19</td>
</tr>
<tr>
<td>4.2.1.2 Prediction</td>
<td>21</td>
</tr>
<tr>
<td>4.2.2 EDMD with Gaussian Basis</td>
<td>23</td>
</tr>
<tr>
<td>4.2.3 Robust DMD with other data sets</td>
<td>25</td>
</tr>
<tr>
<td>4. 3 Drugs Comparison</td>
<td>32</td>
</tr>
</tbody>
</table>
CHAPTER 5. SUMMARY AND DISCUSSION .......................................................... 35
REFERENCES .................................................................................................. 36
# LIST OF FIGURES

| Figure 4-1 | Process of segmenting the nematode from the background and displaying the result to the user [2] | 16 |
| Figure 4-2 | Process of updating midline points from frame to frame [2] | 17 |
| Figure 4-3 | Spectrum plots for Ach 104-1, Ach 104-4, Ach 106-5, Ach 108 and Ach 108-4a | 19 |
| Figure 4-4 | Sorted eigenvalue for Ach 104_1 | 20 |
| Figure 4-5 | Top 5 Dominant Eigenvectors | 20 |
| Figure 4-6 | Test data and Prediction result | 21 |
| Figure 4-7 | Error Plot | 22 |
| Figure 4-8 | Random Steps Error plots | 22 |
| Figure 4-9 | Worm Movements Changes with Colors | 23 |
| Figure 4-10 | Eigenvalues and Sorted Eigenvalues | 24 |
| Figure 4-11 | Top 5 Dominant Eigenvectors | 24 |
| Figure 4-12 | Test data and Prediction result using EDMD with Gaussian basis | 25 |
| Figure 4-13 | Error Comparison between two methods | 25 |
| Figure 4-14 | Prediction for Ach 104-4 | 26 |
| Figure 4-15 | Prediction for Ach 106-5 | 27 |
| Figure 4-16 | Prediction for Ach 108 | 28 |
| Figure 4-17 | Prediction for Ach 108-4a | 29 |
| Figure 4-18 | Test data, Prediction and Error plot using Robust DMD while 50% data is used as training data and the left as test data | 30 |
| Figure 4-19 | Snapshot of Error Comparison Video | 31 |
| Figure 4-20 | Test data, Prediction and Error plot using all data sets to approximate the Koopman operator | 31 |
LIST OF TABLES

<table>
<thead>
<tr>
<th>Table Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 1 Raw Worm Movement Data</td>
<td>18</td>
</tr>
<tr>
<td>Table 2 Velocity for each worm at each frame</td>
<td>33</td>
</tr>
<tr>
<td>Table 3 Distance with identity matrix for each drug combination</td>
<td>34</td>
</tr>
</tbody>
</table>
**NOMENCLATURE**

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DMD</td>
<td>Dynamic Mode Decomposition</td>
</tr>
<tr>
<td>EDMD</td>
<td>Extended Dynamic Mode Decomposition</td>
</tr>
<tr>
<td>WHO</td>
<td>World Health Organization</td>
</tr>
<tr>
<td>C-elegans</td>
<td>Caenorhabditis elegans</td>
</tr>
<tr>
<td>$EC_{50}$</td>
<td>Median Effective Concentration</td>
</tr>
<tr>
<td>FSC</td>
<td>Feedback System Control</td>
</tr>
</tbody>
</table>
ACKNOWLEDGMENTS

I would like to thank my major professor, Dr. Umesh Vaidya for his guidance and support during my graduate research and this thesis. Thank you for giving me this opportunity to learn all these.

I would like to thank my committee members, Dr. Sourabh Bhattacharya and Dr. Pandey Santosh, for their guidance and support throughout the course of this research.

In addition, I would also like to thank my friends and colleagues, Bowen Huang and Xu ma, for their help and support during my graduate student life. I want to also offer my appreciation to those who were willing to participate in my surveys and observations, without whom, this thesis would not have been possible.
ABSTRACT

In this thesis, we demonstrate the application of linear operator theory for data-driven dynamic modeling and analysis of worm data. In particular, Koopman and Perron-Frobenius operators are used for the dynamic modeling of two different worms namely Brugia Malayi and C elegans. Time-series data in the form of video is used to generate reduced order dynamics model to capture the moment of these two worms under different operating conditions. While the moment of the worm is in general modeled as a nonlinear dynamical system, our proposed linear operator theoretic framework provides for a linear representation of the nonlinear dynamics. The linear representation is made possible by shifting the focus from the state space to the space of functions. We exploit this linear representation for data-driven modeling of worm dynamics. For data-driven dynamic modeling, we construct a finite dimensional approximation of these linear operators. Two popular algorithms, Dynamic Mode Decomposition (DMD) and Extended Dynamic Mode Decomposition (EDMD) are used for the finite dimensional approximation of the linear Koopman operator from time series data. The data-driven model is used for prediction of worm dynamics and the comparison of worm movement under different operating conditions caused by the exposure of worm to different drug cocktails. The developed dynamic model will be used to understand the impact of different drug cocktails on worm moment thereby providing a systematic data-driven approach for drug discovery.
CHAPTER 1. INTRODUCTION: THESIS FORMATTING

1.1 Background

Infections from parasitic nematode have a huge impact on humans and livestock. Lymphatic filariasis, also known as elephantiasis, is one kind of human filarial infection caused by nematode parasites. There are almost 150 million people are affected by this kind of disease in the developed countries [1]. *Brugia Malayi* is one type of the filarial worms which will cause the Lymphatic filariasis. Thus, it is important to study the behavior and the movement of the nematode parasites.

In the previous work, a cross-platform nematode tracking program has been designed which is able to detect and track the complex posture by the *Brugia Malayi* [2]. Several different parameters including movement trends, head velocity and curvature are tracked in order to characterize the movement behavior of *Brugia Malayi*. With the raw movement data, we can use linear operator based method to analyze the behavior of the nematode worm. Furthermore, based on the characteristics of the Koopman transfer operator, we are able to predict the worm movement using different approaches which will be illustrated in the rest of the paper.

People with an active infection can take medications to kill the worms in the blood. These medications stop the spread of the illness to others, but they do not completely kill all the parasites [3]. More researches are created to find a better way to cure the parasitic infection. To test the effect of drugs on parasitic nematodes, the free living *C-elegans* are good examples to determine the drug targets. *C-elegans* are organisms which share many important biological characteristics with human bodies. *C-elegans* are only 1mm long and they are easy to grow in agarose plates [4]. All cells of its body can be seen with a
microscope and its life cycle is only 2-3 weeks [5]. Thus, different methods have been
developed around *C-elegans* to test how drugs affect their movement and behavior.

### 1.2 Research Motivation

To test the effects of potential drugs on *Brugia Malayi*, the movement of the parasite
is required to be quantified. One of the challenges in analyzing the worm behavior is to find
some underlying characteristics in complex motor actions [6]. Several methods are used to
analyze the behavior of another worm, *C. elegans*, which are the most widely used nematode
model organism. Greg J. Stephens and Bethany Johnson-Kerner were able to show that the
space of shapes of *C.Elegans* are low-dimensional with just four dimensions which they
called as four eigenworms accounting for 95% of shape variance [6]. They then predicted the
role of each eigenworms and predict the movement based on this property.

Zachery was able to compute key behavior parameters to quantify the movement of the
nematode. With the sharing data, we are considering the movement of worms as a linear
dynamic system. Thus, we are able to apply linear operator based method to study the
behavior of the dynamic worm system. Furthermore, we want to predict the movement of
worms so that we can determine the effect of different drugs on *Brugia Malayi*.

Since Koopman operator based method is able to determine the behavior of nematode,
we can use this method to test the effect of different drugs. We can compare the results to
show which drug combination is more effective.

### 1.3 Thesis Organization

In Chapter 2, the thesis began with a review for the study of behavior of *Brugia Malayi*
and drug test methods on *C-elegans*. History of transfer operator based method with different
approaches will be introduced.
Chapter 3 contains the knowledge of linear operator, Koopman operator, different approached of data driven approximation of Koopman operator and the design of the predictor.

Chapter 4 shows all the simulation results including preprocess of the worm data, plots of spectrum and prediction results and a comparison between different drug combinations using Koopman operator based method.

Chapter 5 summarizes all the chapters in the thesis and purpose the future work of worm behaviors.
CHAPTER 2. REVIEW OF LITERATURE

Because of the need for safe and effective drugs on human filarial parasite, *Brugia Malayi*, many potential drugs have been examined to test the effect on filarial nematodes in vitro and in vivo. These drugs include nitazoxanide and tizoxanide [7]. Meanwhile, the tracking methods for measuring the movements of nematodes are developed. Multiple parameters including movement, velocity and curvatures are able to be tracked by nematode tracking programed developed by Zachery Njus and Panvey Santosh [2]. These data can be further analyzed to study the behavior of *Brugia Malayi*.

Different approaches have been made to study the behavior of another nematode model organism, *C.elegan*. Through microfluidic assays and kinematic analysis, different forms of locomotion have been found [8] [9]. Another approach has made through the dimensionality and dynamics of *C.elegans* [6]. The motor behaviors of worms are treated as a low dimensional system. These dimensions are able to provide a summary of worm behavior. Besides, they can be used to predict the movement of worms. The challenging part is that the system of worm’s movement is a nonlinear system which is hard to analyze. However, this approach motivates us to use the linear operator based methods.

Despite the disease caused by parasitic nematode, some advances have developed multidrug resistance in parasites [10]. There are a number of drugs approved by the World Health Organization(WHO). Due to the cost and time of new drugs, some considered to combine different drugs to let the drug more effective than each of the drugs [10]. A Feedback Control System (FSC) is used to choose different drug combinations from four individual drugs [11]. Basically, every single drug is tested and the median effective concentration ($EC_{50}$) value for each drug is calculated. The FSC then choose six
concentration keys for each drug. A concentration key is a value for a percentage of the dose of an individual drug where the first concentration key is the $EC_{50}$ value and the second one is the half of the percent concentration and so on. For each iteration, the FSC will choose eight drug combinations to test which is a combination of four concentration key from four different drugs (For example, P1, ..P4). Average centroid velocity and track curvature of worms are used to represent the behavior of worms in order to test the effect of different drugs. Eight combinations of drugs are compared (P1 versus T1, P2 versus T2 and so on). The combination with lower average velocity is considered as more effective. Four better performed combinations are remained and considered as new P1, P2, P3 and P4. The FSC will choose another four combinations to compare with the remaining four combinations for the next iteration. The process will be repeated until the best four combinations are tested. However, we want to provide a new Koopman operator based method to show that it can efficiently show the effect of different combinations of drugs.

Linear operator based methods including Koopman and Perron-Frobenius operator are used to analyze nonlinear dynamic systems [12] [13] [14] [15]. The basic idea is to shift the focus from the system which is nonlinear to the system which is linear. The advantage of linearity can be used to study and analyze the behavior of a nonlinear system. One of the challenge is to construct an accurate finite dimensional approximation. Multiple data-driven approximations of transfer operator have been developed including Dynamic Mode Decomposition(DMD) and Extended Dynamic Mode Decomposition(EDMD) [15] [16]. Besides that, a robust computation of Koopman Operator is developed to control the measurement and process noise in the random dynamical systems(RDS) [17].
More importantly, the Linear transfer operator allows us to map data back to the nonlinear state space which can be used to do some predictions. S. Sinha, H. Bowen, U. Vaidya used robust approximation algorithm to predict the IEEE 9 bus system [17]. All methods will be used to form a finite dimensional approximation of Koopman operator and further prediction. More introduction of Koopman operator and approximations will be illustrated in the next few chapters.
CHAPTER 3. DATA-DRIVEN APPROXIMATION OF LINEAR OPERATOR

In this chapter, we will introduce the linear operator based method and how to analyze the nonlinear system. We will give reviews on different data-driven approximations of linear operator including DMD and EDMD. Both theoretic review and algorithms will be shown. Besides, a robust computation of Koopman operator will be explained.

3.1 Linear Operator

Linear operator method are usually used to analyze and design a nonlinear dynamic system. What it does is to map the points from the state space where the system is nonlinear to the space of functions where the system is linear. In general, we use a set of observables to compute the functions defined on the state space of a system. With set of larger functions of observables, the original properties of the dynamic system will be governed by a new evolution law depending the choice of observables [15]. Furthermore, we can choose which observables to design own evolution law for the dynamic system, the system would be completely determined by the spectrum of the evolution operator. [15] The advantage of the approach is that we can apply effective algorithm of linear system to nonlinear system for analysis and design problems.

3.1.2 The Koopman Operator

Since we are using the Koopman operator during the experiment and it is central to our projects. We will overview the Koopman operator theoretic concepts.

Consider a discrete time dynamic system

\[ x_{t+1} = F(x_t) \]

Where \( x_t \in M \subseteq \mathbb{R}^N \) is the state vector and \( M \) is the state space, \( F : M \rightarrow M \) is a function which describes the system evolution [15]. Let \( n \in \mathbb{Z} \) be the discrete time so that we have a
discrete dynamic system \((M, n, F)\). Let \(\mathcal{F}\) be the space of observables and \(\phi \in \mathcal{F}\) with \(\phi : M \rightarrow \mathbb{C}\). The Koopman operator \(K\) acts on functions of state space:

\[
K\phi = \phi \circ \mathcal{F}
\]

where \(\circ\) denotes the composition of \(\phi\) with \(\mathcal{F}\) [15].

The Koopman operator defines a new dynamic system \((\mathcal{F}, n, K)\). The new system governs the evolution of observables. Because the Koopman operator is linear and infinite-dimensional, even when the original system is nonlinear and evolves on a finite-dimensional space, the final system is linear. The goal of Koopman operator is to apply the tools developed for linear system to a nonlinear system. Thus, we can have a linear approximation of a nonlinear system [15].

Assume \(\mu\) be the eigenvalues of Koopman eigenvalues, \(\varphi\) be the eigenfunctions of Koopman operators and \(v\) be the Koopman operator modes. The relation between eigenvalues and eigenfunctions are defined as:

\[
K\varphi = \varphi \mu
\]

The set of all Koopman eigenvalues are called the spectrum of the koopman operator [18]. It also follows:

\[
\varphi(x_t) = \mu^t \varphi(x_0)
\]

Let \(g(x) = x\) and \(g \in \mathcal{F}\). Assume that every element of \(g\) is in the span of Koopman operator eigenfunctions, then \(g\) can be obtained as [12]:

\[
g(x) = \sum_{k=1}^{K} u_k \varphi_k(x)
\]
where \( u_k \) is the \( k \)-th Koopman mode and \( \varphi_k \) is the \( k \)-th Koopman eigenfunctions. Thus, we can obtain the future system state either by evolving \( x \) or by evolvint the observable through Koopman operator [15]:

\[
F(x) = (Kg)x = \sum_{k=1}^{K} \mu_k u_k \varphi_k(x)
\]

We will use this equation in the following parts to do the approximation and prediction.

Koopman Operator is a linear but infinite dimensional operator which governs the evolution of observables of the system state [13]. Normally, A Koopman operator matrix will be obtained which the eigenvalues, eigenfunctions and modes enable us to reconstruct the state of the original system from our new linear system [15]. From this perspective, we can apply Koopman operator-based framework to nonlinear worm system and analyze the behavior of the worm movement. More importantly, this approach enables us to map back from the linear system to the nonlinear system. It allows us to make some predictions based on the property of the Koopman operator matrix. Dynamic Mode Decomposition(DMD) and Extended Dynamic Mode Decomposition(EDMD) are two popular algorithms for approximating the spectrum of Koopman operator. The procedure of DMD and EDMD will be explained in the below sections.

### 3.2 DMD And EDMD Approaches

The Dynamic Mode Decomposition (DMD) was introduced to analyze the nonlinear fluid flow [19]. Extended DMD was developed to better approximate the Koopman operator eigenvalues, eigenvectors and modes. I will briefly explain the method of EDMD and DMD and their relations.
3.2.1 EDMD

Consider that we have a data set of snapshot pairs,

\[ \bar{X} = [x_1, x_2, \ldots, x_M], \quad \bar{Y} = [y_1, y_2, \ldots, y] \]

where \( x_i \in X \) and \( y_i \in Y \). The two pair of data sets are assumed to be two consecutive snapshots i.e., \( y_i = T(x_i) \). Now we choose \( \mathcal{D} = [\psi_1, \psi_2, \ldots, \psi_K] \) as the dictionary basis of observables. We assume that the dictionary function are belong to \( \psi_i \in L_2(X, B, \mu) = \mathcal{G} \), where \( \mu \) is some positive measure not necessarily the invariant measure of \( T \) [12]. Let \( \mathcal{G}_D \) denote the span of \( \mathcal{G} \) such that \( \mathcal{G}_D \subset \mathcal{G} \). The goal is to find the best approximation of Koopman operator on the space spanned by the dictionary basis that we choose. We also define the function \( \Psi: X \to \mathbb{C}^K \)

\[ \Psi(x) := [\psi_1(x), \psi_2(x), \ldots, \psi_K(x)] \]

During this process, we are mapping the data from state space to the space of dictionary basis. By definition, any function \( \phi \in \mathcal{G}_D \) can be written as [12]:

\[ \phi = \sum_{k=1}^{K} a_k \psi_k = \Psi^T a \]

Now we want to find a finite dimensional approximation of the Koopman operator. Assume \( K \in \mathbb{R}^{k \times K} \) be the Koopman operator matrix. Since \( \mathcal{G}_D \) is not an invariant subspace of the Koopman operator, we have some residuals in the problem [15]:

\[ K\phi = \Psi(Ka) + r \]

To find the optimal approximation \( K \) for the Koopman operator, we need to minimize the residual which leads us to a least square problem. \( K \) will be the solution of:
\[
\min_K \| G K - A \|_F
\]

\[
G = \frac{1}{M} \sum_{m=1}^{M} \Psi(x_m)^T \Psi(x_m)
\]

\[
A = \frac{1}{M} \sum_{m=1}^{M} \Psi(x_m)^T \Psi(y_m)
\]

where \( G, A \in \mathbb{C}^{k \times k} \). The optimization problem can be solved as another problem to get the K:

\[
K_{EDMD} = G^+ A
\]

Where + denotes pseudoinverse. As a result, the K matrix is a finite dimensional approximation of Koopman operator. The eigenvalues of K are the EDMD approximation of the Koopman operator. The right eigenvectors of K generate the approximation of the eigenfunctions and the left eigenvectors of K generate the approximation of the Koopman modes [15].

3.2.2 DMD

The DMD operator can be defined as [16]

\[
K_{DMD} = \bar{Y} \bar{X}^+
\]

Where \( \bar{Y} \) and \( \bar{X} \) are defined in (3.1). DMD can be considered as a special case of EDMD where the basis functions are linear monomials which is similar to a one-term Taylor expansion [15]. DMD can have an accurate approximation of Koopman operator if the eigenfunctions can be approximated by linear monomials. The advantage of EDMD is that the approximation is governed by the chosen dictionary basis. In a word, EDMD expanded
the choice of basis functions and as a result EDMD may result in a better approximation of linear operator for different problems.

3.2.4 Dictionary Functions

The accuracy of EDMD depends on what dictionary functions we choose and the method used to obtain the data. The choice of dictionary functions should be rich enough to govern the original system and approximate the linear operator. The optimal dictionary functions are difficult to find because the domain on the original system is not necessarily known [15]. In our experiment, besides the linear basis using in DMD, we are using two common basis functions. They are Polynomial Basis and Gaussian Basis.

3.2.4.1 Polynomial Basis

Polynomial Basis is often used in problem where the data are normally distributed. The functions are the products of the Hermite polynomials in a single dimension [15].

3.2.4.2 Gaussian Basis

Gaussian Basis, or radial basis functions(RBF) is another choice of dictionary functions for approximation. It is defined as:

\[ \phi_j(x) = \exp\left(-\frac{\|x-\mu_j\|^2}{2\sigma^2}\right) \] [20]

Where \( \mu \) is the centers of the raw data which will be calculated by k-means clustering and \( \sigma \) is the standard deviation which will be defined in RBF.

3.3 Robust Computation of Koopman Operator

Besides DMD and EDMD approaches, we are using another method called robust computation of Koopman operator, S. Sinha, H. Bowen, U. Vaidya [17]. We are assuming that there is some noise during the measurement or process. If the data set has some
uncertainty, it will affect the accuracy and convergence of the approximation. We turn this uncertainty in data set to a deterministic norm bounded uncertainty. By doing this, a min-max optimization problem is designed for the approximation of linear operator. The robust optimization also balance between the quality of approximation and the complexity of approximation which show a better prediction result than DMD and EDMD. We will show the simulation in next chapter.

### 3.3.1 Robust approximation of Koopman operator

We now assume that the data points are perturbed by some noise:

$$\delta x_k = x_k + \delta, \delta \in \Delta \ [17]$$

where $\Delta$ represents an uncertainty set. Then we choose $\mathcal{D} = [\psi_1, \psi_2, ..., \psi_K]$ as the dictionary basis of observables. The choice of the dictionary functions is the same as what we used in EDMD. Defined the the function $\Psi: X \to \mathbb{C}^K$

$$\Psi(x) := [\psi_1(x) \ \psi_2(x) \ ... \ \psi_K(x)]$$

We then need to find the matrix $K$, which is a finite dimensional approximation of Koopman operator while minimizing the residual term $r$:

$$\left[\frac{1}{M} \sum_{m=1}^{M} \Psi(x_m)^T \Psi(x_m + \delta x_m) K - \Psi(x_m)^T \Psi(y_m) \right] a \ [15]$$

However, we need to maximize the residual term in the robust optimization equation because the uncertainty has an opposite effect. Hence, we have a min-max optimization function as follows [17]:

$$\min_{K} \max_{\delta \in \Delta} \|G_\delta K - A\|_F$$

Where
\[ G = \frac{1}{M} \sum_{m=1}^{M} \Psi(x_m)^T \Psi(x_m + \delta x_m) \]

\[ A = \frac{1}{M} \sum_{m=1}^{M} \Psi(x_m)^T \Psi(y_m) \]

The problem is in general nonconvex so that we need to convex it and transfer it as:

\[ \min K \max \delta \in \Delta \| (G + G_{\delta})K - A \|_F \]

In the convex problem, \( \delta G \) is bounded as:

\[ \| \delta G \|_F \leq \lambda \Lambda \Gamma \]

Thus, \( \| \delta G \|_F \leq \lambda \). With this proof, we can change out min-max function to a least square function. The problem 3.3 is equivalent to:

\[ \min K \| GK - A \|_F + \lambda \| K \|_F \]

where \( r \) represents a regularization term penalize the Frobenius norm of matrix \( K \) and is also known as Tikhonov regularization [17].

### 3.4 Design of Predictor

One of the advantage of linear operator is that we can map data back to the state space after we have the approximation. Thus, we are able to do some prediction problems using the characteristic of the matrix \( K \) of Koopman operator. For the design of the prediction, we first use part of the data as training data. We use the same way as mentioned in DMD, EDMD and robust computation to approximate the Koopman operator. After we got the finite dimensional matrix \( K \), we can select some part of data as test data. Let
\{x_0, ..., x_M\} be the training data and we first map the initial condition $x_o$ from state space to the space of observables:

$$\overline{x}_0 = \Psi(x_0) =: z \in \mathbb{R}^K$$

We can propagate the next steps using Koopman operator as:

$$z_n = K^n z$$

We can map these points back and have predictions using:

$$\overline{x}_n = Cz_n$$

Where we can get $C$ by solving a least square problem:

$$\min_C \sum_{i=1}^M \|x_i - C\Psi(x_i)\|_2^2$$

Thus, we create a new dynamic system

$$x_t = C(K^t\Psi(x_0))$$

All the simulation results will be shown in the next chapter.
CHAPTER 4. SIMULATION RESULTS

In this chapter, we will show all the simulation results using Koopman operator to predict the *Brugia Malayi*. First of all, we will show you how to preprocess the worm frame to get raw movement data. Next, we will show the simulation results using different approaches of approximation of the Koopman Matrix. The prediction results and error plots will be shown and compared. At last, we will show the comparison between different combinations of drugs using Koopman operator.

4.1 Preprocess

We got all the worm movement data from Dr. Panvey Santosh and Zachery Njus. First of all, I will explain how they measure the worm and show the pixel binary data to users. We then convert the data points to apply the Koopman operator Method.

4.1.1 Image process

They first changed the original image to a gray-scale image and then they calculated the mean pixel intensity. The threshold is then set to 75% of mean pixel and intensity and the image is converted to binary [2].

![Image](image.png)

Figure 4-1 Process of segmenting the nematode from the background and displaying the result to the user [2].
4.1.2 Midline Evolution

They then updated midline points from frame to frame. They captured a smoothed border and they calculated normal vectors pointing to the body at every point. At each iteration, a body point is located within 150% width of the body between border points [2]. The body points are moved in the direction of normal line to border point until the combined movement of all points falls below the threshold [2].

Figure 4-2 Process of updating midline points from frame to frame [2].

4.1.3 Data manipulation

Totally, we have 5 videos of worm's movement. For different videos, each worm is divided into n=75 points and then its shape is automatically described by 75 points on the worm. We capture 1020 steps of the worm's movement as time-series data so that we have a 75×2040 matrix for each video. Let \( \{ x_1, x_2 \ldots x_M \} \) be the data set where \( x_k \in \mathbb{R}^L \), since each point has two x and y coordinates in two dimensional space, we arrange the vector \( x_k \) such that the first two coordinates of \( x_k \) corresponding to the x and y position of the first point on the worm and so on. Hence, the state space dimension here is \( L = 2n = 150 \) and the matrix is transformed to 150×1020. The first 75 rows will be the evolution of x positions.
and the next 75 rows will be the evolution of y positions. Totally, each point will evolve 1020 steps.

<table>
<thead>
<tr>
<th>Table 1. Raw Worm Movement Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_1 )</td>
</tr>
<tr>
<td>( y_1 )</td>
</tr>
</tbody>
</table>

4.2 Worm dynamic system decomposition and prediction

We have 5 data sets which describes the movement of worms. They are Ach 104-1, Ach 104-4, Ach 106-5, Ach 108 and Ach 108-4a. We want to do the predictions for all of the data sets. We are using three different method to approximate the Koopman Operator matrix. Furthermore, we compare the prediction results for each methods. The six different approaches are:

- Robust DMD (linear basis)
- EDMD with Gaussian Basis
- Robust EDMD with Polynomial Basis

In the next few sections, we will show you the results with three different approaches. The Robust DMD and Robust EDMD with Polynomial Basis gave us very close prediction
and we will talk them together. Another method is EDMD with Gaussian basis. We will show the prediction and we will compare these two methods.

4.2.1 Robust DMD and Robust EDMD with Polynomial Basis

4.2.1.1 Worm Decomposition

We want to emphasis these two methods because not only both prediction error plots are small but also their results are very similar. For EDMD with polynomial basis, we are using degree with 1. In order to study the dynamic features of worms, we plot the spectrum of matrix $K$ from the Koopman operator. The spectrum plot for all the data sets of worm movements is shown in Fig 4.3. All the eigenvalues are inside a unit cycle which means the system is stable.

![Spectrum plots for Ach 104-1, Ach 104-4, Ach 106-5, Ach 108 and Ach 108-4a](image)

We also plot the sorted eigenvalue which will help us identify the dominant eigenvalues. Here is an example of the sorted eigenvalue for Ach 104-1. While some of the eigenvalues...
are zero or very small, we only care about the eigenvalues close to 1 which their mode are stable.

![Figure 4-4 Sorted eigenvalue for Ach 104_1](image)

Each eigenvalue correspond to one eigenvector. We plot the eigenvectors with five dominant eigenvalues. These eigenvectors show the shape that the worm look like during the time.

![Figure 4-5 Top 5 Dominant Eigenvectors](image)
4.2.1.2 Prediction

In this section, we use linear operator method to predict the worm's movement. First we choose the first 400 steps of data as training data and the next 100 steps as test data which means we are using 80-20 rule. We plot the test data and prediction result in 3-D. The x-y coordinates are the points value and the z coordinate is the time. We first plot data for Ach 104-1

![Test data and Prediction result](image)

a) Test data  

b) Prediction

Figure 4-6 Test data and Prediction result

An error plot is made to show the difference between test data and our prediction. We normalize the difference for each point at every step. Since there are 75 points on each worm, the error equation for each step $i$ is:

$$Error_i = \frac{||Test_i - Prediction_i||_2}{75}$$

The error plot is shown:
From the plot, the x-axis shows the number of time steps for test data and the average error for each step is plotted along y-axis. We can tell that the error for each point at every step is less than ±1.5. It shows that the robust DMD and Robust EDMD with polynomial basis can give us an accurate prediction. Besides the error plot, we also randomly select five steps inside the test data and we can compare the error in 2-D plots.
We can tell from Fig.4.8 that our prediction method can capture the curvature in the first two plots. With the increase of steps, the error increases. This is because our k matrix is not exactly the matrix which can map the system from state space to the function space. Because of the error of K, when we power the K matrix, the error increases. For better understanding the error between the test data and the prediction result, we plot the total steps in 2-D in one figure. The shape of the worm for the first step is plotted as red. Then the color gradually changes to green. The shape of the worm at the last step is then plotted as green. The figure in shown as:

![Worm Movements Changes with Colors](image)

**Figure 4-9 Worm Movements Changes with Colors**

### 4.2.2 EDMD with Gaussian Basis

Now we change to another approach, EDMD with Gaussian Basis. We want to compare the results with the above one. Since the range of the raw data varies over 100 pixels, we choose $\sigma = 2.5$ which is the standard deviation. Furthermore, we choose the number of basis function to 1000 so that it can be rich enough to govern the original system. We also plot the spectrum of the matrix K using new approach and the first 5 dominant eigenvectors. Ach 104-1 is used as data for all the simulation results.
Figure 4-10 Eigenvalues and Sorted Eigenvalues

Figure 4-11 Top 5 Dominant Eigenvectors

The prediction result using EDMD with Gaussian Basis is shown in Figure 4.11. We can tell that the EDMD method cannot give us a good prediction about the worm movement. At future steps, the approximation cannot fully predict the worm’s movement.
We also plot an error figure which shows the error between test data and prediction results using EDMD with Gaussian Basis. The error is around ± 20 to ± 40 which is larger than the error plot using Robust DMD. From the plot, we can tell that Robust DMD gave us a better prediction for almost every step.

### 4.2.3 Robust DMD with other data sets

We also use the Robust DMD method to predict other data sets. Test data, prediction result and error plots are shown for each data set. Besides, two graphs about how the worm movement changes with invariant color for test data and prediction are shown.
Figure 4-14 Prediction for Ach 104-4
Figure 4-15 Prediction for Ach 106-5
Figure 4-16 Prediction for Ach 108
Furthermore, we changed the number of training data set and test data set to do the further prediction. This time, we use 50% of the total data as training data and the left 50% as test data. All the simulation results are shown below. From the plots, Robust DMD also can give us an accurate prediction with small average point error.
Figure 4-18  Test data, Prediction and Error plot using Robust DMD while 50% data is used as training data and the left as test data
Two videos are also created to visually compare the error between test data and prediction results using two different approaches. The above dashed red line is the movement of the test data while the below black line is the predicted movement. Two videos are attached and a snapshot is shown.

Figure 4-19 Snapshot of Error Comparison Video

Finally, we combine every data set as one data and approximate the K matrix using Robust DMD method. However, when we do the prediction, we use 40% of each data set as training data and the next 10% as test data. We can show that the approximated Koopman operator can predict the worm’s movement.

Figure 4-20 Test data, Prediction and Error plot using all data sets to approximate the Koopman operator.
4. 3 Drugs Comparison

In this section, I will first briefly explain the procedure provided in [10]. The four drugs we are choosing is levamisole, pyrantel, tribendimidine and methyridine. Each drug concentration is tested on the *C-elegans* and the median effective concentration ($EC_{50}$) is calculated. A feedback control system (FSC) is implemented to choose eight drug combinations to test (which is called P1 to P4 and T1 to T4). In each combination, the concentration of each drug is less or equal to $EC_{50}$. For each combination, the drug is tested on at least six worms. From the test, the average centroid velocity is obtained and normalized. The performance of P and T groups is compared which is P1 versus T1, P2 versus T2 and so on. The drug combinations with lower average centroid velocity will be the better performance combinations and they will be retained as P1 to P4. This is the first iteration. For the next iteration, FSC will choose another four drug combinations as T1 to T4 and the performance of eight drug combinations are compared. The procedure is repeated until the winning drug combination is achieved.

During the process, the FSC determines a better drug combination by comparing the average centroid velocity. The drug combination with lower average centroid velocity has a better performance. With the raw normalized velocity, we propose a new Koopman Operator based method to compare the performance of different drug combinations. We argue that the drug combination with lower centroid velocity does not mean that it is more effective. We define that the drug combination which the worm velocity is closer to an invariant set has better performance. Thus, we compare the worm velocity for identity matrix to determine the drug effects.
Table 2. Velocity for each worm at each frame

<table>
<thead>
<tr>
<th></th>
<th>Worm 1</th>
<th>Worm 2</th>
<th>Worm 3</th>
<th>...</th>
<th>Worm 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>91.11</td>
<td>50.93</td>
<td>10.73</td>
<td>...</td>
<td>129.74</td>
</tr>
<tr>
<td>2</td>
<td>33.95</td>
<td>16.97</td>
<td>16.97</td>
<td>...</td>
<td>15.18</td>
</tr>
<tr>
<td>3</td>
<td>454.36</td>
<td>454.32</td>
<td>115.90</td>
<td>...</td>
<td>136.88</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>599</td>
<td>80.71</td>
<td>16.97</td>
<td>...</td>
<td>...</td>
<td>147.42</td>
</tr>
</tbody>
</table>

Table 2 shows an example of the velocity for every second (totally 10 minutes) for P1 at the first iteration. We choose velocity of six worms as our datasets. Then we approximate the Koopman operator using robust DMD since it has a better prediction results. We can tell that if the velocity is stable, for example, the snapshot pair $x_m = y_m$, then $G = A$ which the $K$ matrix will be an identity matrix since

$$\min_K \|GK - A\|_F + \lambda \|K\|_F$$

Thus, by comparing the difference between $K$ matrix approximated from each drug combination and the identity matrix, we can tell which one has the better performance.

Using the raw velocity data, we approximate each matrix $K$ for all eight drug combinations and comparing the distance with identity matrix. In Table 3, we show the distance with identity matrix for each drug combination. From the table, we can tell that based on the distance, T1, T2, T3, T4 has better performance than P1, P2, P3 and P4 using the same comparing method in [10]. While in [10], all the comparisons are the same except P1 has
lower average velocity than T1. The reason is that even though P1 has lower average velocity, the velocity trend for P1 is not stable.

Table 3. Distance with identity matrix for each drug combination

<table>
<thead>
<tr>
<th></th>
<th>Distance</th>
<th></th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>P1</strong></td>
<td>1.67</td>
<td><strong>T1</strong></td>
<td>0.92</td>
</tr>
<tr>
<td><strong>P2</strong></td>
<td>0.94</td>
<td><strong>T2</strong></td>
<td>0.93</td>
</tr>
<tr>
<td><strong>P3</strong></td>
<td>0.98</td>
<td><strong>T3</strong></td>
<td>0.97</td>
</tr>
<tr>
<td><strong>P4</strong></td>
<td>0.98</td>
<td><strong>T4</strong></td>
<td>0.93</td>
</tr>
</tbody>
</table>
CHAPTER 5. SUMMARY AND DISCUSSION

In this paper, we explore the application of linear operator based method to analyze the behavior of worm, *Brugia Malayi*. The data driven approaches, Dynamic Mode Decomposition (DMD), Extended DMD and a robust version of Koopman operator, are used to approximate the finite dimensional approximation of Koopman operator for dynamic nonlinear worm movement. We approximate the spectrum of Koopman operator using robust DMD, robust EDMD with Polynomial Basis and EDMD with Gaussian Basis. We show that the linear operator based method can capture the worm behavior. Besides that, We also predict the worm’s movement using the different approaches. We plot the error difference for test data and prediction results using different set of training data. By showing the error plot, we can tell that the robust version can give a better approximation of Koopman operator compared with EDMD-based approximation.

For the *C-elegans*, we can show that Koopman operator based method can be used to test the effect of different drug combinations by comparing the distance of each Koopman matrix and the identity matrix. We show a comparison matrix using the data and we compare the results using other methods.

In the future, we can use the Koopman operator based method to test the effect of different drugs on *Brugia Malayi* which can help the development of new effective drugs.
REFERENCES


