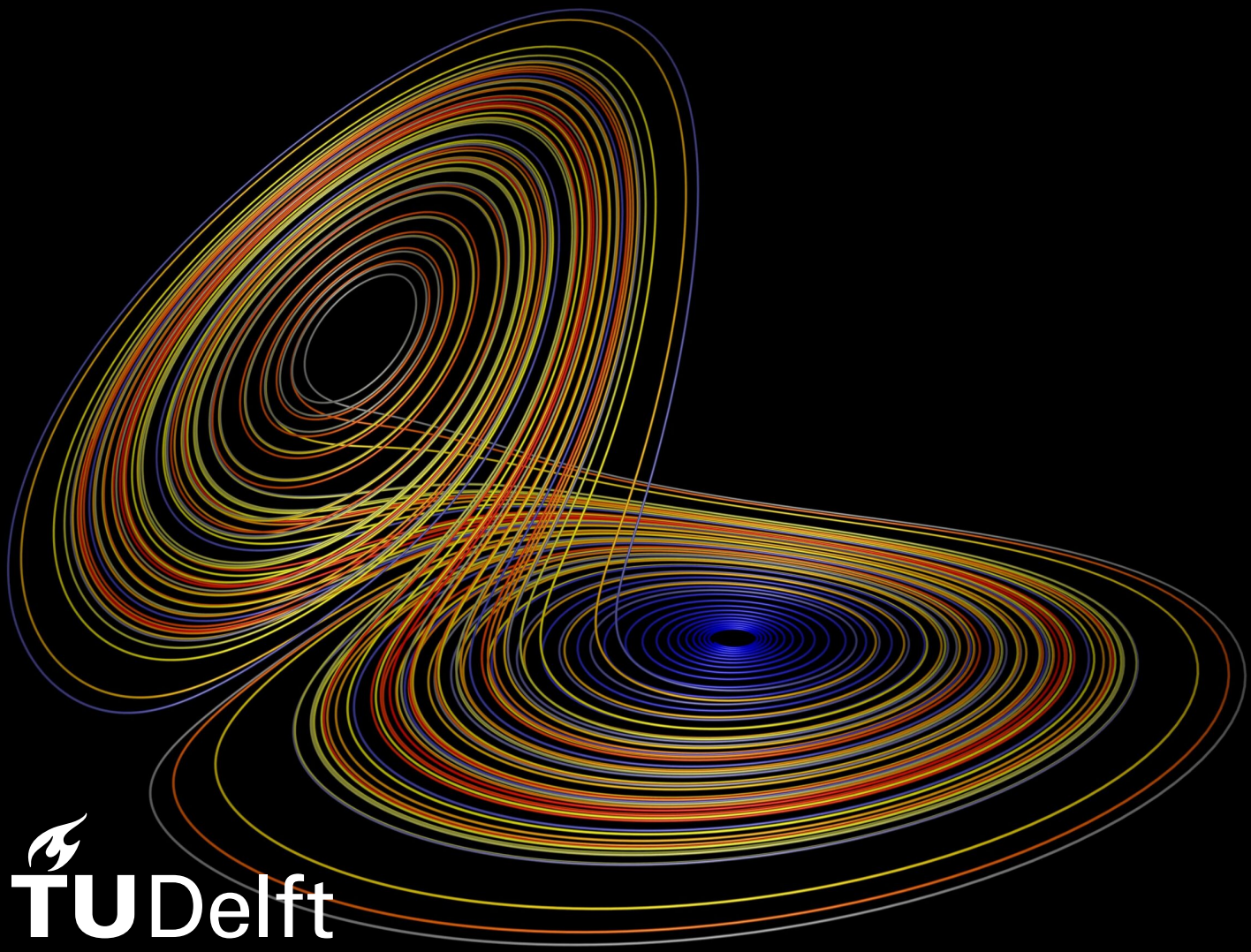
Cluster-based Reduced-order Modeling and Control of Chaotic Systems

Master of Science Thesis
Aerospace Engineering

Yuxuan Yang

Delft University of Technology



TU Delft

Cluster-based Reduced-order Modeling and Control of Chaotic Systems

by

Yuxuan Yang

Student Name	Student Number
Yuxuan Yang	5724872

Instructor:	Dr. N.A.K.Doan
Project Duration:	December, 2023 - August, 2024
Faculty:	Faculty of Aerospace Engineering, Delft

Cover:	Chaos Theory: Do we really know what we think we know? by Vibhor Singh
Style:	TU Delft Report Style, with modifications by Daan Zwaneveld

TU Delft

Preface

This thesis represents the culmination of my academic journey in pursuit of a Master's degree in TU Delft, and I am deeply grateful for the support and guidance I have received along the way.

First and foremost, I would like to express my sincere gratitude to my supervisor, Dr. N.A.K.Doan. His insightful guidance, encouragement, and expertise have been invaluable throughout this research. His belief in my abilities and unwavering support have motivated me to push beyond my limits and strive for excellence. I could not have asked for a better mentor.

I would also like to extend my thanks to Faculty of Aerospace Engineering in TU Delft. The academic environment, resources, and opportunities provided by this esteemed institution have played a crucial role in shaping my intellectual growth and research skills. I spent months in the lab of aerodynamics, and the kind colleagues and the nice working environment plus free coffee helped me a lot to go through all the struggles during the thesis. I am proud to have been part of such a vibrant and supportive academic community.

Finally, I would like to acknowledge the support of my family and friends, whose encouragement and understanding have been a source of strength throughout this journey.

Thank you all for being a part of this significant milestone in my life. As this chapter of my life ended with the following thesis, a new one awaits.

Yuxuan Yang
Delft, August 2024

Abstract

This thesis explores the development and application of clustering-based reduced-order modeling (ROM) for chaotic systems, with an emphasis on both predictive modeling and control strategies. Chaotic systems, characterized by their sensitivity to initial conditions and complex spatio-temporal dynamics, present significant challenges in terms of prediction and control. Traditional numerical methods for solving these systems are computationally intensive and often impractical for real-time applications. Reduced-order modeling techniques, which seek to represent high-dimensional systems in lower-dimensional subspaces, offer a promising alternative.

Clustering-based ROM is highlighted as a potentially effective method due to its data-driven nature and computational efficiency. This study focuses on studying the influence of different parameters in the clustering-based ROM approach in modeling and developing a robust control algorithm to mitigate extreme events. This is investigated on three different chaotic systems of increasing complexity: the Lorenz system, the truncated Charney-DeVore (CDV) system and the Moehlis-Faisst-Eckhart (MFE) system. The ability of the clustering-based ROM in reproducing those systems' statistics is confirmed. The influence of the number of clusters and the order of modeling on the ROM accuracy is explored, and a quantitative method to determine the number of clusters when modeling with clustering-based ROM is proposed.

For the control part, a clustering-based control strategy is applied to the CDV system and MFE system. In the CDV system, the control aims to move the system away from the blocked state, and in the MFE system, the objective is to prevent extreme events (which take the form of quasi-relaminarisation events). In both cases, the clustering-based control manages to achieve these objectives, with a reduction of 90% of extreme events in the MFE case. These results highlight the potential of clustering-based control.

This research contributes to the field by providing a viable approach to managing chaotic systems with reduced computational demands, offering potential applications in various engineering and scientific domains.

Keywords: Chaotic system, Reduced-order modeling, Clustering, Extreme events, System control

Contents

Preface i

Abstract ii

List of Figures iv

Nomenclature vii

1 Introduction 1

2 Research Background 3
2.1 Chaotic Systems . . . 3
2.2 Reduced-order Modeling (ROM) . . . 5
2.3 Overview of Existing Methods . . . 5
2.3.1 Proper Orthogonal Decomposition (POD) . . . 5
2.3.2 Dynamic-Mode Decomposition (DMD) . . . 5
2.3.3 Autoencoders (AE) . . . 6
2.3.4 Temporal ROM Techniques . . . 6
2.3.5 Other Black-box Model . . . 6
2.3.6 Cluster-based Modeling Methods . . . 7

3 Research Questions 9

4 Methodology 10
4.1 Clustering . . . 10
4.2 Cluster-based Markov Model (CMM) . . . 11
4.3 Cluster-based Network Model (CNM) . . . 13
4.4 Cluster-based Control (CBC) Algorithm . . . 14

5 Test cases Set Up 16
5.1 Lorenz System . . . 16
5.2 Truncated Charney-DeVore (CDV) System . . . 18
5.3 Moehlis-Faisst-Eckhart (MFE) System . . . 20

6 Modeling Results 24
6.1 Lorenz System . . . 24
6.1.1 CMM Results of the Lorenz System . . . 24
6.1.2 CNM Results of the Lorenz System . . . 27
6.1.3 Discussion . . . 28
6.2 Truncated Charney-DeVore System . . . 33
6.2.1 CMM Results of the CDV System . . . 33
6.2.2 CNM Results of the CDV System . . . 36
6.3 Moehlis-Faisst-Eckhart System . . . 41
6.3.1 CMN Results of the MFE System . . . 41
6.3.2 CNM Results of the MFE System . . . 43

7 Control Results **46**
7.1 Truncated Charney-DeVore System 46
7.2 Moehlis-Faisst-Eckhart System 49

8 Conclusion and Recommendations **52**
8.1 Conclusion 52
8.2 Recommendations 53

References **55**

A Nine modes of MFE system **58**

B Nelder-Mead Method **60**

List of Figures

2.1 Chaotic dynamical system . . . 3
2.2 Examples of attractors . . . 4

4.1 Sketch of times and periods employed in the cluster-based network model [25] . 13

5.1 Lorenz system variables with time . . . 17
5.2 Lorenz attractor . . . 17
5.3 CDV system time evolution . . . 19
5.4 CDV system projection on $x_1 - x_4$ latent space . . . 20
5.5 Time series of MFE system . . . 22
5.6 Extreme events in MFE system . . . 23
5.7 Extreme events in MFE system in phase space . . . 23

6.1 Clustering result of the Lorenz system . . . 25
6.2 Network graph representation of the Lorenz system . . . 25
6.3 Transition matrix of Lorenz system . . . 26
6.4 Distance matrix of Lorenz system . . . 26
6.5 Accuracy of prediction for Lorenz system with CMM . . . 27
6.6 Original Lorenz system . . . 27
6.7 Reduced order Lorenz system with CNM . . . 27
6.8 x,y and z evolution in original Lorenz system . . . 28
6.9 x,y and z evolution in reduced-order Lorenz system with CNM . . . 28
6.10 Comparison of Auto-correlation function when $N_{cl} varies$ for Lorenz system . . 29
6.11 Comparison of distribution for Lorenz system when N_{cl} varies . . . 30
6.12 J_m of reduced-order Lorenz system when N_{cl} increases . . . 32
6.13 Auto-correlation for Lorenz system when $N_{CL} = 20$ and $L = 3$ . . . 32
6.14 Auto-correlation for Lorenz system when $N_{CL} = 20$ and $L = 22$ . . . 32
6.15 Distribution for Lorenz system when $N_{CL} = 20$ and $L = 3$ . . . 33
6.16 Distribution for Lorenz system when $N_{CL} = 20$ and $L = 22$ . . . 33
6.17 J_m of the reduced-order CDV system when N_{cl} increases . . . 34
6.18 Network graph representation of the CDV system . . . 34
6.19 Transition matrix of the CDV system . . . 35
6.20 Distance matrix of the CDV system . . . 35
6.21 Accuracy of prediction for the CDV system with CMM . . . 35
6.22 Results of phase space clustering and auto-correlation for CDV system . . . 36
6.23 Time series for the CDV system without modeling . . . 37
6.24 Time series for the CDV system when $N_{cl} = 20$ and $L = 3$ . . . 38
6.25 Time series for the CDV system when $N_{cl} = 50$ and $L = 3$ . . . 38
6.26 Time series for the CDV system when $N_{cl} = 100$ and $L = 3$ . . . 38
6.27 Time series for the CDV system when $N_{cl} = 200$ and $L = 3$ . . . 38
6.28 Auto-correlation for the CDV system when $N_{cl} = 50$ and $L = 3$ . . . 39
6.29 Auto-correlation for the CDV system when $N_{cl} = 50$ and $L = 20$ . . . 39
6.30 Time series for the CDV system without modeling . . . 40

6.31 Time series for the CDV system when $N_{cl} = 50$ and $L = 3$ 40
6.32 Auto-correlation for the CDV system when $N_{cl} = 50$ and $L = 20$ 40
6.33 J_m of the reduced-order MFE system when $N_{cl}changes$ 41
6.34 Network graph representation of the MFE system 41
6.35 Transition matrix of the MFE system 42
6.36 Distance matrix of the MFE system 42
6.37 Accuracy of prediction for the MFE system with CMM 42
6.38 Comparison of Auto-correlation function when N_{cl} varies for the MFE system . 43
6.39 Comparison of Auto-correlation function when L varies for the MFE system . . 44
6.40 Time series for the MFE system without modeling 45
6.41 Time series for the MFE system when $N_{cl} = 80$ and $L = 20$ 45

7.1 $x_1 - x_4$ projection for the original CDV system 47
7.2 $x_1 - x_4$ projection for the controlled CDV system with control strategy 1 . . . 47
7.3 $x_1 - x_4$ projection for the controlled CDV system with control strategy 2 . . . 47
7.4 Time series for the original CDV system 48
7.5 Time series for the controlled CDV system with control strategy 1 48
7.6 $k - \epsilon$ plot for kinetic energy control 50
7.7 $k - \epsilon$ plot for dissipation rate control 50
7.8 ϵ plot for dissipation rate control 51
7.9 k plot for dissipation rate control 51
7.10 Probability density function of the uncontrolled and controlled MFE system . . 51

Nomenclature

Abbreviations

Abbreviation	Definition
AE	Autoencoders
BVE	Barotropic Vorticity Equation
CAE	Convolutional Neural Network Based Autoencoder
CBC	Cluster-based Control
CDV	Charney-DeVore
CMM	Cluster-based Markov Model
CNM	Cluster-based Network Model
CNN	Convolutional Neural Network
CTM	Cluster Transition Matrix
DMD	Dynamic-Mode Decomposition
DOF	Degrees of Freedom
ERA	Eigensystem Realization Algorithms
ESN	Echo-State Network
HiCNM	Cluster-based Hierarchical Network Model
K-L	Kullback–Leibler
KNN	K nearest neighbor
LT	Lyapunov time
LTSM	Long Short-Term Memory
MFE	Moehlis-Faisst-Eckhart
PCA	Principal-Component Analysis
PDF	Probability Density Function
POD	Pproper Orthogonal Decomposition
RNN	Recurrent Neural Networks
ROM	Reduced-order Modeling
SINDy	Sparse Identification of Nonlinear Dynamics
SVD	Singular-Value Decomposition
tCNM	Trajectory-optimized Cluster-based Network Model

Symbols

Symbol	Definition
b	Forcing term for control
$/b_k/$	Cluster-based control amplitude
c_k	k^{th} Cluster-based centroids
$c_k^{(t)}$	k^{th} Cluster-based centroids at time t
$\mathscr{C}_k$	k^{th} Clusters
$\mathscr{C}_k^{(t)}$	k^{th} Clusters at time t
J_i	Inter-clusters variance

Symbol	Definition
J_m	Mean variance of the samples in each clusters and the the centroid of the cluster
J_w	Within-cluster variance
$\mathscr{J}$	Objective function for cluster-based control
$\mathscr{J}_{obj}$	Objective term in objective function for cluster-based control
$\mathscr{J}_e$	Energy cost term in objective function for cluster-based control
k	Kinetic energy in the MFE system
L	Order of model
N	Total number of samples
N_{cl}	Number of clusters
N_e	Normalized factor for energy cost term for cluster-based control
N_{ij}	Number of snapshots that move from $\mathscr{C}_j$ to $\mathscr{C}_i$
N_k	Number of samples in cluster $\mathscr{C}_k$
$N_{kj,l}$	Number of snapshots that move to $\mathscr{C}_l$ before it visited $\mathscr{C}_k$ and $\mathscr{C}_j$ in succession
N_{kj}	Number of all snapshots that visited $\mathscr{C}_k$ and $\mathscr{C}_j$ in succession
N_{obj}	Normalized factor for objective term for cluster-based control
p_K	Probability of the system to in k^{th} cluster
p_K^n	Probability of the system to in k^{th} cluster at n^{th} time step
$\mathbf{p}$	Probability vector of the snapshots
$\mathbf{p}^n$	Probability vector of the snapshots at n^{th} time step
P_{ij}	Transition probability from cluster $\mathscr{C}_i$ to $\mathscr{C}_j$
$\mathbf{P}$	Cluster transition matrix
R	Auto-correlation function
s	State of the system
t	Time
$\mathbf{T}$	Direct transition time matrix
T_{ij}	Direct transition time from i^{th} cluster to j^{th} cluster
$\boldsymbol{u}(t)$	The prediction of the state of the system at time t
β	Feedback gain in cluster-based control
ϵ	Mean dissipation rate in the MFE system
τ_n	Residence time of n^{th} cluster for certain trajectory
τ_{ij}	Transition time from i^{th} cluster to j^{th} cluster for certain trajectory

Symbols that are directly related to specific test cases are not listed here. The definitions can be found in related sections.

1

Introduction

Typical chaotic dynamical systems such as climate, stock market, neural network in brain, and turbulence are important to study but quite difficult to predict, control and optimize. These systems usually have a large range of time and spatial scales, which lead to high dimensionality in addition to nonlinear dynamics. Dynamic modeling is needed to better understand the principles of these systems and control them. Traditional method to model these systems is to numerically solve the governing equations [1] based on a spatial and temporal discretization. However, since these systems are extremely high dimensional and chaotic, to obtain precise results often require large computational resources. Meanwhile, the control strategies based on the full order model will become excessively complicated. Therefore, reduced-order modeling (ROM) techniques has been developed in past decades. These techniques focused on representing the system in a much lower dimension subspace, which can substantially reduce the computational cost. The fast development in machine learning contributed to this area in recent years [2].

This study mainly focuses on one specific reduced-order modeling method, the cluster-based ROM, which is utilized to discover the dynamic evolution of the chaotic systems. Furthermore, a cluster-based control strategy is proposed to prevent the dynamical systems from extreme events. These techniques are applied on the complex systems as Lorenz attractor, truncated Charney-DeVore (CDV) system, and Moehlis-Faisst-Eckhart(MFE) system.

The structure of this thesis is organized into eight chapters, each contributing to the overarching narrative of the research. Below is a brief overview of the contents of each chapter:

Chapter 1 introduces the thesis by providing the introduction, where the origins and motivations behind the research are discussed. This chapter sets the stage by highlighting the significance and urgency of exploring chaotic systems and reduced-order modeling in the current scientific landscape. And in this chapter, the structure of the whole thesis is represented.

Chapter 2 delves into the Research Background, offering a review of the current state of research in chaotic systems and reduced-order modeling. This background serves as a foundation for identifying the research gaps that are addressed in this thesis. Building on this,Chapter 3 outlines the Research Questions, clearly defining the core issues that this study seeks to solve and articulating the specific goals of the research.

Chapter 4 provides an in-depth discussion of the Clustering-Based Reduced Order Modeling and Control algorithm, detailing its various components. This chapter begins by introducing

the clustering algorithm, which plays a crucial role in reducing the complexity of the system by grouping similar states or behaviors. Following this, two distinct reduced-order modeling techniques are presented. The chapter includes an explanation of the corresponding control algorithms as well, which are specifically designed to operate within the reduced-order framework, ensuring effective and efficient system control.

Chapter 5, the three Test Cases used to evaluate the proposed algorithms are presented. How to generate the data for testing is also explained in this chapter. This chapter also includes a detailed explanation of the modeling and control strategies implemented in these cases.

Chapter 6 presents the Modeling Results, showcasing the effectiveness and accuracy of the developed models through extensive experimentation and analysis.

Chapter 7 focuses on the Control Results, discussing the design and implementation of control strategies and evaluating their performance across different test cases.

Finally, Chapter 8 concludes the thesis by summarizing the key findings from Chapter 6 and Chapter 7, and offering suggestions for future research directions.

2

Research Background

In this chapter, the basic background of chaotic dynamical system and reduced-order modeling is discussed. Concepts such as phase space and attractors in dynamical systems are explained. In the reduced-order modeling part, several prevailing methods are introduced. Their advantages and short-comes are discussed as well, which leads to the the research gaps that will be filled by this thesis.

2.1. Chaotic Systems

A dynamical system is a concept used across various disciplines, including mathematics, physics, engineering, biology, economics, and more. At its core, it refers to a system that evolves over time, where the state of the system at any given moment depends not only on its current conditions but also on its history and external influences [3].

In mathematics and physics, dynamical systems are often described using differential equations, which capture how the system's variables change with time. These equations may represent anything from the motion of celestial bodies to the behavior of chemical reactions.

Engineering applications of dynamical systems can range from control systems in robotics and industrial processes to the design of complex mechanical systems like aircraft and automobiles. Understanding the dynamic behavior of these systems is crucial for predicting their performance, stability, and response to various inputs.

If given the equations of motion together with the initial and boundary conditions, the evolution of the system can be computed as a function of time, the system can be called deterministic [4]. For example, if the initial velocity and position of a single pendulum are known, the velocity and position of that pendulum can be described as a function of time easily. However, this is

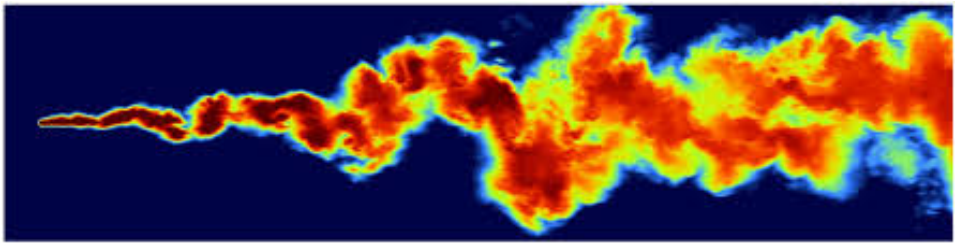


Figure 2.1: Chaotic dynamical system

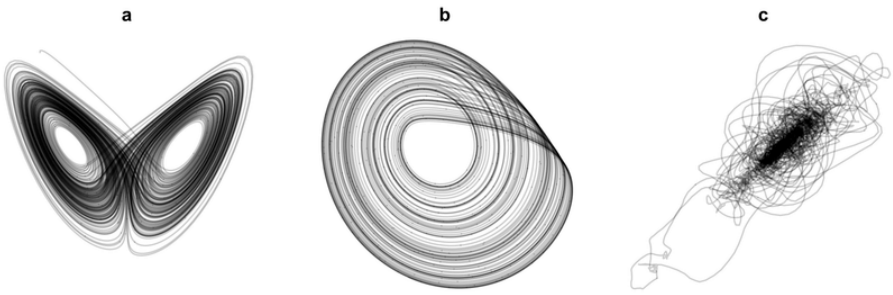


Figure 2.2: Examples of attractors

not always the case in practice. Some of the dynamics systems are sensitive to the initial and boundary conditions. This suggests that only with infinite accuracy of initial and boundary conditions can the accurate solution of the system be obtained [5]. Once a tiny perturbation is added into the initial or boundary condition, the prediction of the equations will be totally differently. This property widely occurs in systems involving non-linear terms. These systems are defined as ill-posed and known as deterministic chaotic systems.

There are various famous examples. Newton successfully solved the two-body problem, where he found the elliptic Kepler-trajectories as the solution, which are considered the hallmark of a completely predictable solution [4]. When the French mathematician Henri Poincaré tried to solve the three-body problem using the same method, he found that the solution is not integrable [6]. The three-body system is chaotic.

Turbulence is another important instance of chaotic systems. With the Navier-Stokes equations and proper initial and boundary conditions, we can define the problem as deterministic [7]. However, tiny variations in the starting state can lead to vastly different outcomes over time. In turbulent flows, even minuscule disturbances can cascade through the system, amplifying into large-scale eddies and swirls that contribute to the overall complexity and unpredictability.

In dynamical systems and control theory, a phase space——also known as a state space——provides a framework for representing all possible "states" of a dynamical or control system. Each unique state is represented as a distinct point within this space. For mechanical systems, the phase space generally includes all potential values of position and momentum variables. This space is essentially the Cartesian product of direct space and reciprocal space. The concept of phase space was first introduced in the late 19th century by Ludwig Boltzmann, Henri Poincaré, and Josiah Willard Gibbs [8].

The dimensions of the phase space are always equal to the number of degrees of freedom (DOF) of the system. Once the phase space is established, the state of the system at certain time can be represented as a point in the phase space. The time evolution of the system can be seen as trajectories in the phase space, representing the progression of the system from one state to the next.

Attractor, in the context of the dynamical systems, is a set of states towards which a system tends to evolve, for a wide variety of initial conditions of the system. In other words, the solution intends to stay in this attractor structure in the phase space. When the evolving variable involves two or three dimensions, the attractor of the dynamic process can be visually represented in two or three dimensions. An attractor can be a single point, a finite collection of points, a curve, a manifold, or even a complex set exhibiting a fractal pattern, referred to as a strange attractor. In cases where the variable is a scalar, the attractor is confined to a

subset of the real number line. The characterization of attractors within chaotic dynamical systems stands as a significant accomplishment within chaos theory.

A good example of singular attractor is the famous Lorenz system's butterfly shaped attractor [9]. This will be discussed later in Chapter 5.

2.2. Reduced-order Modeling (ROM)

Since some chaotic dynamical systems are rather high dimensional in space, reduced-order modeling techniques are significant to reduce the computational cost. Meanwhile, other methods more focuses on simplifying temporal evolution of the system to reduce cost. And some research combines both methods and achieve better results.

The main reduced-order modeling techniques can be divided into two categories. One is knowledge-based, which means that they usually need the governing equations. These methods include proper orthogonal decomposition (POD) with Galerkin projection [10] on Navier-Stokes equations. In contrast are the methods that are purely data-driven. An important example is dynamic-mode decomposition (DMD) [11]. Both sets of methods are discussed below.

2.3. Overview of Existing Methods

2.3.1. Proper Orthogonal Decomposition (POD)

One of the most used ROM techniques is proper orthogonal decomposition (POD) or principal component analysis (PCA) with Galerkin projection [10]. Snapshots of the system are used to extract dominant energy-containing structures, or orthogonal modes to create a reduced-order representation that captures the essential features and dynamic behavior of the system. The temporal evolution of the system is solved with a Galerkin projection, which means projecting the governing equations onto these POD modes, reducing the system to a lower-dimensional model that captures the dominant dynamics with fewer variables. This facilitates dimensionality reduction and provides insights into coherent structures for more efficient analysis and modeling. POD shares close ties with principal-component analysis (PCA) and singular-value decomposition (SVD) [12], both being fundamental dimensionality-reduction methods widely employed in data-driven modeling. This method depends on a linear subspace to approximate data, even though many systems clearly involve nonlinear manifolds. Further, Galerkin projection require the governing equation to model the temporal properties. That forbids this reduced-order method to be used for systems where the set of governing equations is unknown.

2.3.2. Dynamic-Mode Decomposition (DMD)

Dynamic-mode decomposition (DMD) is a purely data-driven method that is also widely used to reduce the dimension of the system [11]. While POD decomposes data into spatial modes and temporal coefficients, DMD decomposes it into dynamic oscillatory modes with a singular frequency [2]. In other words, DMD reduces the order of the system both temporally and spatially. DMD excels in capturing transient dynamics, whereas POD is better suited for quasi-steady flow. DMD has a strong relation with the Koopman mode decomposition [13]. Rowley [14] proposed that DMD modes are a numerical approximation of the Koopman modes. DMD can model the systems' spatial and temporal properties simultaneously. DMD also use a linear subspace to approximate the data, which weakens its ability to capture the nonlinear properties of the system. DMD is sensitive to data length, and its performance may be limited when the time series data length is insufficient, resulting in inaccurate patterns being extracted. When the input data becomes enormous, the computational cost of DMD will increase extremely. Thus, the selection of data length becomes an important problem for DMD. Moreover, DMD is rather sensitive to the noise in the input data. Noise may affect the accuracy and stability

of the extracted patterns.

2.3.3. Autoencoders (AE)

With the development of neural networks, these strong tools are also used in reduced-order modeling. Deep nonlinear autoencoders involve neural networks that are used to extract low dimensional manifolds from the snapshots data. Convolutional neural network-based autoencoder (CAE) shows impressive ability to reduce the spatial dimensions with little error when applied on Kolmogorov flow [15]. The latent space found by CAE [15] not only can be related to the physical space directly, but also exhibits good generality.

Isomap-K nearest neighbor (KNN) manifold learner is another new method to discretize the systems spatially [16]. The tensors obtained in the encoder have clear physical explanations. This method is anticipated to be highly significant in estimation, dynamic modeling, and control across a wide array of configurations characterized by dominant coherent structures.

Autoencoder methods present notable challenges in fluid modeling. Firstly, they demand extensive datasets for effective training, a requirement that can be particularly daunting in the realm of fluid dynamics, where obtaining large volumes of data, especially for intricate systems, can prove to be costly or impractical. Additionally, the computational demands of training sizable autoencoder models are significant, especially when dealing with high-resolution or complex fluid simulations, necessitating ample computational resources and time. Moreover, while autoencoders offer powerful modeling capabilities, their black-box nature limits interpretability, hindering the comprehensive understanding of fluid phenomena essential for accurate modeling and prediction. These shortcomings underscore the importance of carefully considering the trade-offs and exploring strategies to mitigate these challenges, such as integrating complementary techniques or refining model architectures to enhance performance and applicability in fluid modeling tasks.

2.3.4. Temporal ROM Techniques

Some other methods are more focused on modeling the temporal nonlinear dynamics of the system. Long short-term memory (LSTM) network [17], echo-state network (ESN) [18] are two examples of neural network that are capable to capture the temporal evolution of the system. Advances in parsimonious modeling have given rise to the “sparse identification of nonlinear dynamics” (SINDy) [19] algorithm, which efficiently identifies accurate models from data. However, all these approaches face challenges in scalability, proving computationally expensive for even moderately dimensional feature spaces. Thus, they are always used in cooperation with spatial reduced-order techniques.

2.3.5. Other Black-box Model

There are also some pure black box techniques, including Volterra series, autoregressive models, eigensystem realization algorithms, and neural network models [20]. However, their limited interpretability constrains their usage. And some of them require a large volume of data, which is not always available.

In addition to autoencoder methods, several other techniques in dynamical system modeling fall under the category of pure black box methods. These include the Volterra series [21], autoregression models [22], eigensystem realization algorithms [23], and neural network models [20]. Each of these methods offers its own set of advantages and challenges in modeling chaotic dynamical systems.

The Volterra series is a mathematical framework used to represent nonlinear systems. It offers

the advantage of capturing nonlinear interactions within the chaotic system. However, like autoencoders, Volterra series models are often considered black box models due to their complex mathematical formulations, which can limit their interpretability. Moreover, accurately estimating Volterra series coefficients typically requires a large volume of data, which may not always be readily available in fluid dynamics experiments or simulations.

Autoregressive models, on the other hand, are a class of statistical models that predict future values based on past observations. While these models can be relatively straightforward to implement and interpret, they may struggle to capture the complex dynamics of fluid systems, particularly in cases involving turbulence or other nonlinear phenomena. Additionally, autoregressive models may require a significant amount of training data to accurately capture the underlying dynamics of the system, posing challenges in data-constrained scenarios.

Eigensystem realization algorithms (ERA) are data-driven techniques used to identify dynamic modes or coherent structures within fluid flow data. ERA offers the advantage of being able to extract dominant flow features directly from data without relying on predetermined models. However, like autoencoders and Volterra series, ERA algorithms may suffer from limited interpretability, as the identified modes may not always correspond directly to physically meaningful quantities. Moreover, accurate identification of dynamic modes using ERA typically requires a sufficient amount of high-quality flow data, which may be challenging to obtain when simulation or experiment of the dynamical system becomes too expensive.

Neural network models, including deep learning architectures like convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have gained popularity in fluid dynamics due to their ability to learn complex patterns and relationships from data. While neural network models can offer high predictive accuracy and flexibility in capturing nonlinear dynamics, they are often criticized for their black box nature, as understanding the underlying mechanisms learned by the network can be challenging. Additionally, training neural network models typically requires a large volume of data to generalize well to unseen scenarios, which may be a limiting factor in fluid dynamics applications where data availability is constrained.

In summary, while pure black box techniques such as Volterra series, autoregression models, eigensystem realization algorithms, and neural network models offer powerful modeling capabilities in fluid dynamics, their limited interpretability constrains their usage. Additionally, the requirement for a large volume of data, which is not always available, poses challenges in applying these techniques effectively to real-world fluid dynamics problems. As such, a careful consideration of the trade-offs between model complexity, interpretability, and data requirements is necessary.

2.3.6. Cluster-based Modeling Methods

A novel reduced-order method is cluster-based modeling. This modeling of the system is human interpretable by representing dynamics by a handful of coherent structures and their transitions. Kaiser et al [24] first applied cluster-based Markov model (CMM) to a mixing layer. The snapshots are projected into a latent phase space and clustered using the K-means algorithm. Then the temporal evolution is modelled as a probabilistic Markov model of the transition dynamics. The state vector of cluster probabilities may initially start in a single cluster but eventually diffuses to a fixed point representing the post-transient attractor. One obstacle for CMM lies in temporal evolution, where the state rapidly diffuses across the entire attractor, typically within a single time period. This challenge is mitigated by the cluster-based network model (CNM) by Li et al [25], which incorporates time-resolved data to alleviate the loss of dynamic information. Instead of using fixed time period, the dynamics is modeled on

a directed network based only on motion between the clusters. This enables the model to have a more accurate long-time behavior. Illustrating this motion through an airport analogy, the centroids can be likened to nodes, akin to airports, and the flights represent the edges, resembling a deterministic-stochastic flight schedule. This schedule permits only a select few possible flights with associated probabilities and flight times that align with the provided data.

Trajectory-optimized cluster-based network model (tCNM) was proposed by Hou et al [26]. This model optimizes the centroid position and promotes the performance of the model. Deng et al [27] propose a self-supervised cluster-based hierarchical reduced-order modeling methodology to analyze the complex dynamics of a two-dimensional fluidic pinball flow, revealing distinct shedding regimes and providing a visual representation of multi-frequency, multi-attractor behavior through a cluster-based hierarchical network model (HiCNM).

For complex systems such as turbulence, just like POD-Galerkin method, cluster-based modeling contains two stages. First is to find a latent phase space so that the snapshots can be represented as points and be clustered. Then the time evolution of these clusters is modeled. While most researchers focus on the time evolution of the clusters, it is also important to study the method to find the proper latent space. Since the cluster-based modeling can easily give the time evolution of the clusters, if the clusters can be connected to coherent structures in turbulence, the understanding of evolution of coherent structures in turbulence could be developed profoundly.

Cluster-based ROM also showed an impressing potential for dynamical system control. A.G. Nair et al [28] introduced a cluster-based feedback control strategy for post-stall separated flows over airfoils. By partitioning flow trajectories into clusters and optimizing feedback control laws, they minimized the power consumption in aerodynamic flight. This model-free approach effectively redirects flow trajectories to favorable regions, enhancing aerodynamic performance. Applied to turbulent separated flows over a NACA 0012 airfoil, the optimized control laws mitigate high-drag clusters, reducing mean drag. This work addresses challenges in feedback control design for turbulent separated flows at moderate Reynolds numbers.

Wang and Deng [29] proposed a Cluster-Based Control (CBC) strategy aimed at achieving model-free feedback drag reduction. This strategy utilizes multiple actuators and full-state feedback. The full-state CBC method further elucidates the evolution of control flow associated with centroids, thereby contributing to a more comprehensive physical interpretation of the feedback control process. However, CBC has so far not been applied to chaotic systems that can exhibit extreme events.

An important problem for cluster-based reduced-order modeling is that, when dealing with systems with extremely high degree of freedom, the selection of the latent space in which clustering is applied is quite arbitrary. It does not always ensure that the model can preserve the properties of the system. Thus, a general method to find the latent space is required.

3

Research Questions

Chaotic systems are omnipresent. They demonstrate a high sensitivity to their initial conditions and evolve patterns that appear to be random [5]. However, in many areas, having quick and reasonably accurate solution of these systems are necessary. Thus, reduced-order modeling for chaotic dynamical systems is important, since it is significant for predicting and controlling the systems. Various methods have been proposed, including both knowledge-based methods and pure data-driven methods. This array of methodologies underscores the multi-faceted efforts aimed at addressing the challenges posed by chaotic systems. Among these methods, cluster-based reduced-order modeling appears to be a promising one. This method is purely data-driven with quite a limited computational cost and has a good potential for system control.

Based on the drawbacks identified in the previous section, the research questions of this study is established as follows:

How to model and control chaotic systems which can exhibit extreme events using cluster-based reduced-order method?

And the sub-questions are listed as followed:

- What are the key parameters influencing the performance of cluster-based reduced-order methods in capturing coherent structures?
- Does these parameters influence the performance of modeling both in statistical and dynamical aspects?
- Is there a method that can determine these parameters quantitatively?
- Can control algorithm based on cluster-based ROM be used to keep the system in certain state and prevent extreme event from happening?

4

Methodology

As mentioned the Chapter 3, this study focuses on using cluster-based modeling techniques to model the dynamics of chaotic systems and endeavor to propose a cluster-based control algorithm to prevent the system from extreme events.

In this chapter, several aspects of the research methodology are introduced, including clustering, cluster-based Markov model, cluster-based network model, and cluster-based control strategies.

4.1. Clustering

K-means clustering algorithm [30] is used in this study to cluster the snapshots in the (latent) phase space.

K-means algorithm is an unsupervised classification algorithm where samples are partitioned into K representative clusters. Every cluster, denoted as $\mathscr{C}_k$, is characterized by its centroid, c_k, computed as the mean over all observations within that cluster. These centroids serve to reduce the number of degrees of freedom within the feature space.

For a set of cluster-based centroids $\{c_k\}_{k=1}^{N_{cl}}$, the within-cluster variance (J_w) as defined in the work by Goutte and colleagues [31], are given by

$$J_w = \frac{1}{N} \sum_{k=1}^{N_{cl}} \sum_{s \in \mathscr{C}_k} \|s - c_k\|^2 \tag{4.1}$$

Here, s denotes the snapshots, N is the total number of observations, and N_{cl} is the number of clusters. And c_k, the cluster centroids is defined as:

$$c_k = \frac{1}{N_k} \sum_{s \in \mathscr{C}_k} s \tag{4.2}$$

Where N_k is the number of observations in cluster $\mathscr{C}_k$

The optimization problem is to find a set of centroids that can minimize J_w.

$$(c_1, ...c_{N_{cl}}) = \underset{\mathscr{C}}{argmin} J_w \tag{4.3}$$

After the random initialization of the centroids, the optimization process can be done in two stages [32], and they are run iteratively to find the optimized set of centroids.

The first stage is Assignment. In this process, each observation is assigned to its nearest centroids. Mathematically, this entails partitioning the observations based on the Voronoi diagram defined by the cluster centroids.

$$\mathscr{C}_i^{(t)} = \left\{ s : \left\| s - c_i^{(t)} \right\|^2 \leq \left\| s - c_j^{(t)} \right\|^2 \forall j, 1 \leq j \leq k \right\} \tag{4.4}$$

where each s is assigned to exactly one $\mathscr{C}_i^{(t)}$, even if it could be assigned to two or more of them.

The second stage is Update. The means of the observations assigned to each cluster are recalculated and these centroids serve as the nearest centroids for the next iteration.

$$c_i^{(t+1)} = \frac{1}{N_k} \sum_{s \in \mathscr{C}_k} s \tag{4.5}$$

K-means algorithm has converged when the assignments no longer change. The algorithm is not guaranteed to find the global optimum, sometimes it leads to a local optimum. As a result, the initialization is rather significant for K-means algorithm. The initial guess of the centroids will not only determine the quality of the clustering but also the iteration times of the algorithm. In this study, K-means++ is utilized. This algorithm spreads the initial guess of centroid: the first cluster center is chosen uniformly at random from the data points that are being clustered, after which each subsequent cluster center is chosen from the remaining data points with probability proportional to its squared distance from the point's closest existing cluster center [33]. Though this seeding method itself cost more computation time than the original K-means, the initial guess of centroids provides a much shorter convergence time for the clustering part.

4.2. Cluster-based Markov Model (CMM)

The cluster-based Markov model simplifies the evolution of dynamical system into transitions between each cluster with a Markov process. And the transitions are elucidated as a cluster transition matrix (CTM), which serves as the propagator in terms of probability.

Since we assume the dynamical system as a Markov process, the state in the next time step is only dependent on the state at the current time step. Firstly, the state is defined as a vector that represent the probability of the snapshots to fall in each clusters.

$$\mathbf{p}(t) = [p_1, ..., p_{N_{cl}}]^T \tag{4.6}$$

The CTM is noted as $\mathbf{P}$. P_{ij} represents the transition probability from cluster $\mathscr{C}_i$ to $\mathscr{C}_j$ in one forward time-step, the elements of the resulting CTM can be inferred by

$$P_{ij} = \frac{N_{ij}}{N_j} \tag{4.7}$$

where N_{ij} is the number of snapshots that move from $\mathscr{C}_j$ to $\mathscr{C}_i$, and N_j is the number of snapshots in $\mathscr{C}_j$.

Every element in $\mathbf{P}$ are positive by definition.

$$P_{ij} \geq 0, \quad i, j = 1, 2, ..., N_{cl}. \tag{4.8}$$

The column sum of $\mathbf{P}$ is equal to unity, as the total probability for all transitions is one. This property is necessary to preserve the normalisation condition of the probability vector.

$$\sum_{j=1}^{N_{cl}} P_{jk} = 1 \tag{4.9}$$

Two important properties of the cluster probability vector at time step n $\mathbf{p}^n$ can be derived from Equation 4.8 and Equation 4.9, and are given in Equation 4.10 and Equation 4.11.

$$\sum_{k=1}^{N_{cl}} p_k^n = 1 \tag{4.10}$$

$$p_k^n \geq 0 \tag{4.11}$$

With this CTM, the dynamic property of the system can be identified. The probability of state at time step $n+1$ is equal to the step at time step n times $\mathbf{P}$.

$$\mathbf{p}^{n+1} = \mathbf{p}^n \mathbf{P}, \quad n = 0, 1, 2... \tag{4.12}$$

Let $\mathbf{p}^0$ be the initial probability distribution vector. Then, the cluster probability vector at time $n+1$ $\mathbf{p}^{n+1}$ is compactly given by

$$\mathbf{p}^{n+1} = \mathbf{p}^0 \mathbf{P}^n \tag{4.13}$$

The CTM, which serves as the propagator, defines a time-homogeneous Markov chain with well-known properties [34].

The long-term behavior can be analyzed by considering the powers of the CTM as defined in Equation 4.13. The asymptotic probability distribution is obtained by

$$\mathbf{p}^\infty := \lim_{n \to \infty} \mathbf{P}^n \mathbf{p}^0 \tag{4.14}$$

and the infinite-time CTM $\mathbf{P}^\infty$ is defined as

$$\mathbf{P}^\infty := \lim_{n \to \infty} \mathbf{P}^n \tag{4.15}$$

4.3. Cluster-based Network Model (CNM)

While the Markov model sees the dynamics of the system as transitions between each centroid, cluster-based network model resolves the deterministic transition time and allow the model to predict the state as a function of time. Discarding the critical transition time step in CMM, CNM model defines two important time parameters.

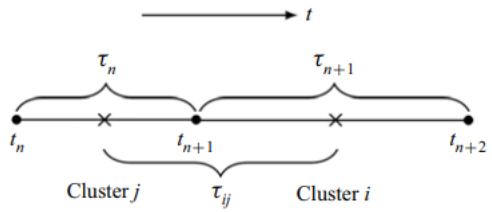


Figure 4.1: Sketch of times and periods employed in the cluster-based network model [25]

The residence time for each cluster is defined as

$$\tau_n = t_{n+1} - t_n \tag{4.16}$$

The process is illustrated in Figure 4.1. Equation 4.16 states that the residence time is the time when the system state stays in that cluster, which is equal to the exit time minus the entrance time of that cluster.

Let j and i be the indices of the clusters after t_n and t_{n+1} respectively. Then the transition time from j to i is defined as half of the residence time of both clusters.

$$\tau_{ij} = \frac{\tau_n + \tau_{n+1}}{2} \tag{4.17}$$

It is the half of sum of the residence time in the two clusters. And the direct transition time T_{ij} from cluster j to cluster i is defined as the average of all values of τ_{ij}.

$$T_{ij} = \langle \tau_{ij} \rangle \tag{4.18}$$

With this direct transition time matrix $\mathbf{T}$ and the CTM, the dynamics of the system can be modelled. Notably here the definition of CTM has been changed. In CNM, rather than assuming a Markov process, more previous states can be taken into account. And this is quantified by the order of model L, which is defined as the number of clusters the system have previously visited before entering the current one that will have an effect on determining its next destination. Thus, the CTM becomes much more complicated. Instead of a square matrix with the rows representing the "from states", and the columns representing the "to states", the CTM for CNM has much more rows than columns. The "from states" is now a sequence of clusters from which possible transition is happening. The calculation for the CTM is similar to Equation 4.7. For instance, if the order of model is set to be $L = 2$, the probability of the system to move to $\mathscr{C}_l$, which has previously visited $\mathscr{C}_k$ and $\mathscr{C}_j$ is given by

$$P_{kj,l} = \frac{N_{kj,l}}{N_{kj}} \tag{4.19}$$

where $N_{kj,l}$ is the number of snapshots that move to $\mathscr{C}_l$ before it visited $\mathscr{C}_k$ and $\mathscr{C}_j$ in succession, and N_{kj} is the number of all snapshots that visited $\mathscr{C}_k$ and $\mathscr{C}_j$ in succession.

For each transition between clusters, the next cluster is firstly selected based on the probability given by CTM. And the direct transition time of this process is given by the direct transition time matrix and added to the total time. The states within that time period are interpolated between two centroids. The prediction for $t \in [t_n, t_{n+1}]$ is given by

$$\boldsymbol{u}(t) = \alpha_n(t) c_{k_n} + [1 - \alpha_n(t)]\, c_{k_{n+1}}, \quad \alpha_n = \frac{t_{n+1} - t}{t_{n+1} - t_n}. \tag{4.20}$$

where c_{k_n} represents the centroid of the cluster that the system enters at time $t = t_n$, and $c_{k_{n+1}}$ represents the centroid of the cluster that the system enters at time $t = t_{n+1}$.

4.4. Cluster-based Control (CBC) Algorithm

After the latent space is selected and the cluster analysis is performed, a cluster-based control strategy can be developed. A forcing term can be added into the governing equations and the forcing amplitude can be defined as [28]:

$$b\left(b_k, s(t)\right) = \beta \frac{\sum_{k=1}^{N_{cl}} b_k \mathrm{e}^{-\|s(t)-c_k\|^2/J_i}}{\sum_{k=1}^{N_{cl}} \mathrm{e}^{-\|s(t)-c_k\|^2/J_i}} \tag{4.21}$$

where β is the feedback gain which is set to unity, unless otherwise noted. And J_i denotes the inter-clusters variance.

$$J_i = \frac{1}{N} \sum_{k=1}^{N_{cl}} N_k \|c_k - \bar{c}\|^2 \tag{4.22}$$

And $\bar{c} = (1/N) \sum_{k=1}^{N_{cl}} N_k c_k$.

The system is controlled by a proportional feedback controller depending on the current state in the feature space $s(t)$. To find the cluster-based control amplitude $\{b_k\}$, an objective function whose input are control amplitude $\{b_k\}$ is needed. Thus, the problem becomes an optimization problem for a nonlinear multivariate objective function. In each test cases, the objective function varies based on the purpose of control.

In this study, simplex searching method is chosen since it does not require the computation of the gradient or derivatives of the objective function. And it can be quite effective for optimizing smooth and well-behaved functions in moderate-dimensional spaces.

Simplex search method, also known as the Nelder-Mead method, proceeds through a series of iterations, gradually narrowing down the region of interest in the parameter space until a satisfactory solution is found. The detail of Nelder-Mead method is presented in Appendix B

The iteration contains four steps: initialization, ordering, reflection and expansion, contraction and shrinkage.

This iterative process continues until a termination criterion is satisfied, typically resulting in a parameter vector that corresponds to a local minimum (or maximum) of the objective function. However, due to the lack of global convergence guarantees, multiple runs from different

initial conditions may be necessary to obtain the global optimum in non-convex optimization problems.

Once the amplitude for each cluster is found, the controller will be able to lead the system to certain direction and achieve the control objective.

5

Test cases Set Up

The modeling algorithm is applied on three different chaotic dynamical systems varying in complexity to validate the accuracy of prediction. They are Lorenz system, Truncated Charney-DeVore (CDV) System, and Moehlis-Faisst-Eckhart (MFE) System.

The control algorithm is applied on Truncated Charney-DeVore (CDV) system and Moehlis-Faisst-Eckhart (MFE) system with different purpose of control. Different kinds of objective is used to examine the robustness of the cluster-based reduced-order control method.

5.1. Lorenz System

The Lorenz system was abstracted from atmospheric convection studies. Specifically, it was developed by meteorologist Edward Lorenz in 1963 [9] while he was investigating simplified models of fluid motion in the atmosphere, particularly to understand and simulate chaotic behavior in weather patterns. This system can be expressed as a set of three ordinary differential equations which is known as the Lorenz equations.

$$\begin{cases} \dfrac{\mathrm{d}x}{\mathrm{d}t} = \delta(y - x), \\ \dfrac{\mathrm{d}y}{\mathrm{d}t} = x(\rho - z) - y, \\ \dfrac{\mathrm{d}z}{\mathrm{d}t} = xy - \beta z \end{cases} \tag{5.1}$$

The equations describe the behavior of a two-dimensional fluid layer that is uniformly heated from below and cooled from above. Specifically, they represent the rate of change over time of three key variables: x, which is proportional to the convection rate; y, related to the horizontal temperature variation; and z, corresponding to the vertical temperature variation. The constants δ, ρ, and β are system parameters that are proportional to the Prandtl number, Rayleigh number, and certain physical dimensions of the fluid layer.

One of the most widely used parameter set of δ, ρ, and β is $\delta = 10$, $\rho = \frac{8}{3}$, and $\beta = 28$, which is also assumed in this thesis. With these parameters, the Lorenz equations can be easily solved with the Runge-Kutta method when initial conditions are given. A typical initial value of the system is $(x_0, y_0, z_0) = (-3, 0, 31)$, and it is used for all Lorenz system test cases in this thesis.

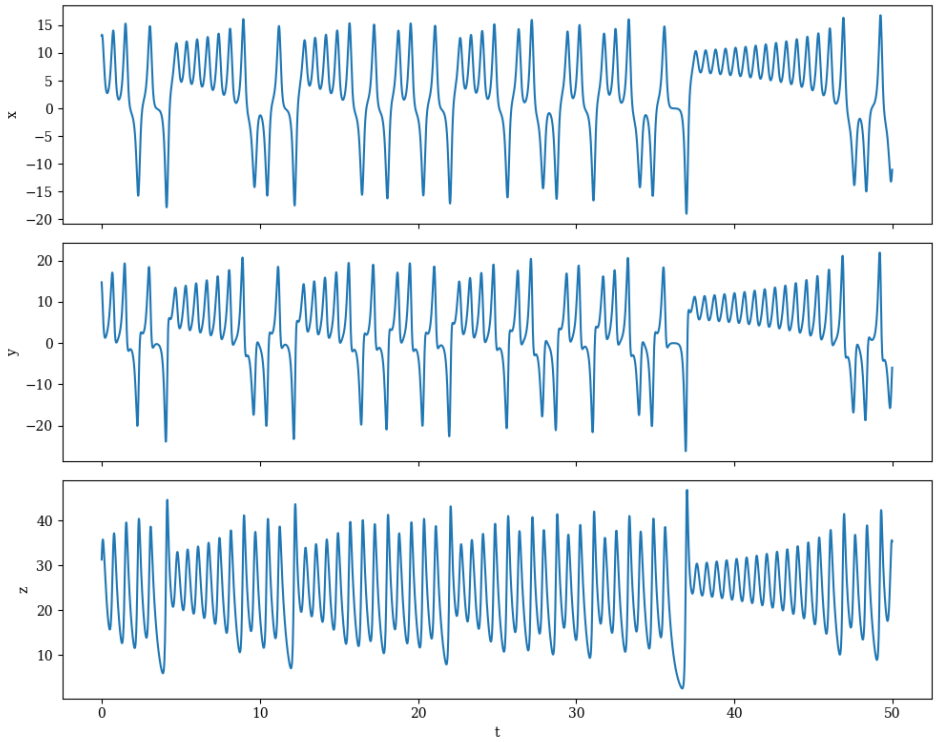


Figure 5.1: Lorenz system variables with time

The time evolution of the three variables are shown in Figure 5.1, where a bimodal property can be identified clearly. Since Lorenz system only has three variables, the selection of the phase space is quite simple. A vector space with x, y, and z as the axis is determined to be the phase space, and the system can be expressed as the famous Lorenz attractor in it. The shape of the attractor looks like a butterfly, and can be found in Figure 5.2.

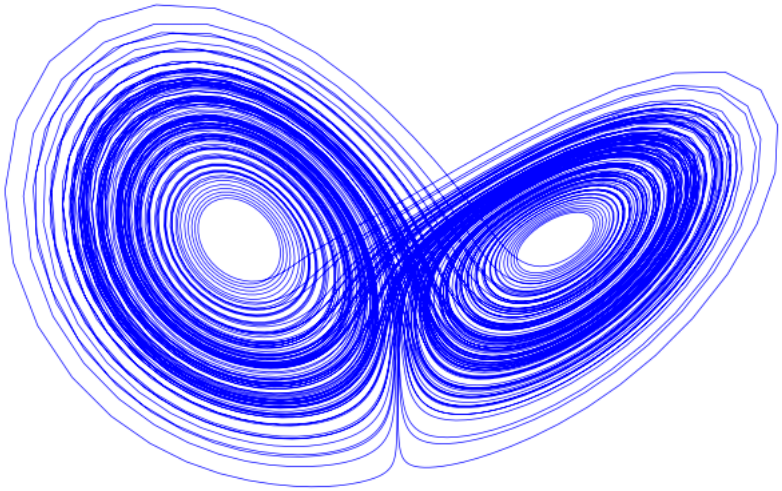


Figure 5.2: Lorenz attractor

Lorenz system is mainly used to analyse the parameters in cluster-based reduced-order modeling, since it is simple and no effort is needed to find a latent space to applied the clustering techniques. Notably, the first 5% of the data is truncated to eliminate the influence of the initial conditions. Unless otherwise specified this truncation will be applied to all the test cases in this thesis.

5.2. Truncated Charney-DeVore (CDV) System

The Charney-DeVore system is used to describe the regime behavior of the atmosphere. The regime behavior of the atmosphere encompasses the various stable or shifting patterns and states that define how the atmosphere behaves over different time scales and conditions. Understanding these regimes helps in predicting weather patterns, assessing climate impacts, and studying the dynamics of atmospheric circulation. However, to understand and model such behavior is extremely difficult. Charney and DeVore [35], who stated that flow regimes should be identified with equilibrium solutions of the equations describing the evolution of large-scale atmospheric flow, proposed a set of differential equations to model such behavior.

The model is derived by applying a Galerkin projection and truncation to the barotropic vorticity equation (BVE) within a β-plane channel.

The finite-dimensional model of the Charney-DeVore system consists of six ordinary differential equations as:

$$\begin{aligned}
\dot{x_1} &= \gamma_1^* x_3 - C\left(x_1 - x_1^*\right) \\
\dot{x_2} &= -\left(\alpha_1 x_1 - \beta_1\right) x_3 - C x_2 - \delta_1 x_4 x_6 \\
\dot{x_3} &= \left(\alpha_1 x_1 - \beta_1\right) x_2 - \gamma_1 x_1 - C x_3 + \delta_1 x_4 x_5 \\
\dot{x_4} &= \gamma_2^* x_6 - C\left(x_4 - x_4^*\right) + \epsilon\left(x_2 x_6 - x_3 x_5\right) \\
\dot{x_5} &= -\left(\alpha_2 x_1 - \beta_2\right) x_6 - C x_5 - \delta_2 x_4 x_3 \\
\dot{x_6} &= \left(\alpha_2 x_1 - \beta_2\right) x_5 - \gamma_2 x_4 - C x_6 + \delta_2 x_4 x_2
\end{aligned} \tag{5.2}$$

With:

$$\begin{aligned}
\alpha_m &= \frac{8\sqrt{2}m^2\left(b^2+m^2-1\right)}{\pi\left(4m^2-1\right)\left(b^2+m^2\right)}, \quad \beta_m = \frac{\beta b^2}{b^2+m^2}, \\
\delta_m &= \frac{64\sqrt{2}}{15\pi}\frac{b^2-m^2+1}{b^2+m^2}, \quad \gamma_m^* = \gamma\frac{4\sqrt{2}mb}{\pi\left(4m^2-1\right)}, \\
\epsilon &= \frac{16\sqrt{2}}{5\pi}, \quad \gamma_m = \gamma\frac{4\sqrt{2}m^3 b}{\pi\left(4m^2-1\right)\left(b^2+m^2\right)}
\end{aligned}$$

for $m = 1, 2$, where the model coefficients are:

$x_1^* = 0.95$; $x_4^* = -0.76095$; $\beta = 1.25$; $\gamma = 0.2$; $b = 0.5$; $C = 0.1$.

The initial condition is set to $(0.11, 0.22, 0.33, 0.44, 0.55, 0.66)$. Since the CDV system is a low dimensional ordinary differential equation as well, the Runge-Kutta method is used to obtain data.

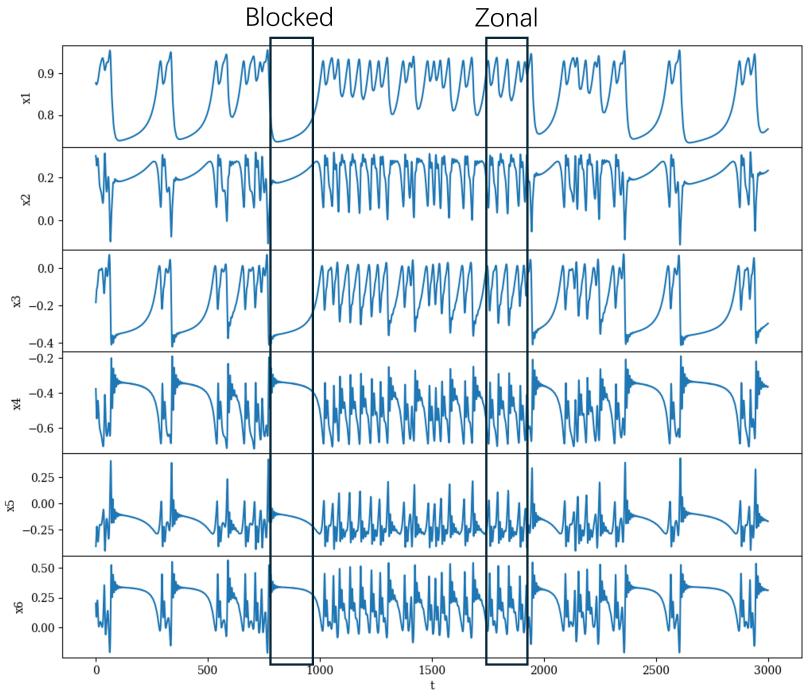


Figure 5.3: CDV system time evolution

The time evolution of each variables is shown in Figure 5.3. A bifurcation can be identified. The system alternates in between two representative regimes, the zonal regime and the blocked regime, both of which originate from the combination of topographic and barotropic instabilities. The zonal regime reveals strong fluctuations of all modes, while the blocked regime shows a slow evolution and a large decrease in x_1. Clustering will be directly applied on the phase space with the six variables considered as the basis. To identify the regime of the system more obviously, the system is projected into a 2-dimensional latent space, using x_1 and x_4 as the basis. The corresponding regimes are clearly illustrated. A singular attractor is shown in Figure 5.4.

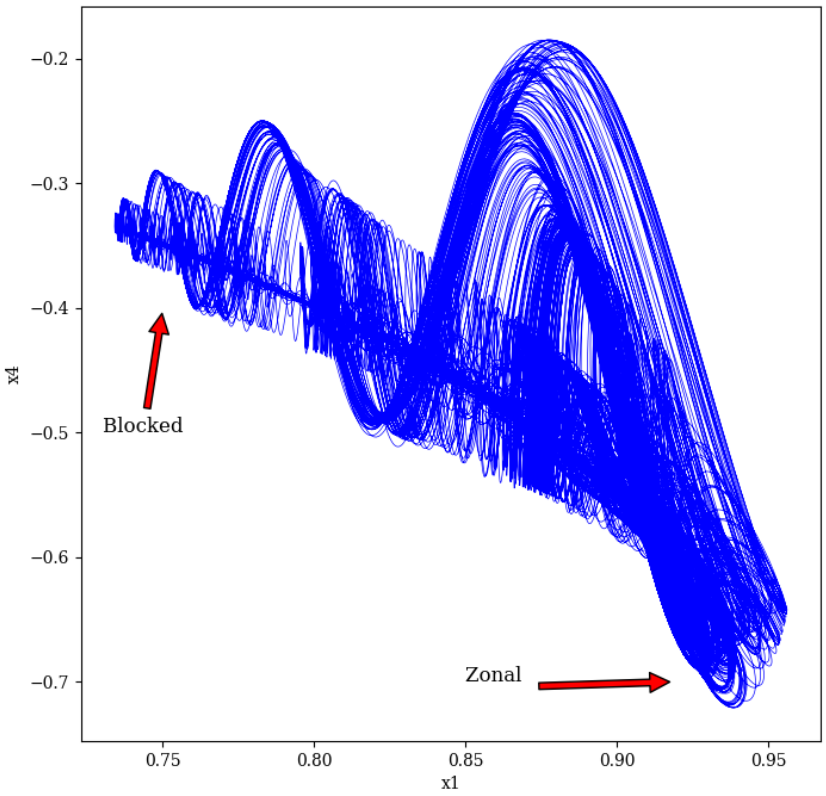


Figure 5.4: CDV system projection on $x_1 - x_4$ latent space

The purpose of this projection is to provide a simple way to determine the regime of the system. This information is essential for the implementation of the control algorithm, as the primary objective for the CDV system will be to maintain the system within the zonal regime. Specifically, this entails attempting in constrain the attractor towards the lower right corner of the projected phase space. The details of how to achieve this will be elucidated in Chapter 7.

5.3. Moehlis-Faisst-Eckhart (MFE) System

Moehlis-Faisst-Eckhart (MFE) system is a low-dimensional model for turbulent shear flows [36]. The system models a sinusoidal shear flow with nine ordinary differential equations, in which the fluid flows between two free-slip walls subjected to a sinusoidal body force. a_1 to a_9 represent the amplitude of each modes, describing the basic mean velocity profile and its modification, downstream vortices, streaks, and instabilities of streaks, with other modes being a consequence of the non-linear interactions. Once the time evolution of the nine modes' amplitude is obtained, the flow field can be reconstructed by multiplying with the orthogonal basis modes, whose expressions can be found in Appendix A.

The differential equations for the nine modes are

$$
\begin{aligned}
\dot{a_1} + \zeta_1 a_1 &= \zeta_1 + \xi_{11} a_6 a_8 + \xi_{12} a_2 a_3 \\
\dot{a_2} + \zeta_2 a_2 &= \xi_{21} a_4 a_6 - \xi_{22} a_5 a_7 - \xi_{23} a_5 a_8 - \xi_{24} a_1 a_3 - \xi_{25} a_3 a_9 \\
\dot{a_3} + \zeta_3 a_3 &= \xi_{31} \left(a_4 a_7 + a_5 a_6\right) + \xi_{32} a_4 a_8 \\
\dot{a_4} + \zeta_4 a_4 &= -\xi_{41} a_1 a_5 - \xi_{42} a_2 a_6 - \xi_{43} a_3 a_7 - \xi_{44} a_3 a_8 - \xi_{45} a_5 a_9 \\
\dot{a_5} + \zeta_5 a_5 &= \xi_{51} a_1 a_4 + \xi_{52} a_2 a_7 - \xi_{53} a_2 a_8 + \xi_{54} a_4 a_9 + \xi_{55} a_3 a_6 \\
\dot{a_6} + \zeta_6 a_6 &= \xi_{61} a_1 a_7 + \xi_{62} a_1 a_8 + \xi_{63} a_2 a_4 - \xi_{64} a_3 a_5 + \xi_{65} a_7 a_9 + \xi_{66} a_8 a_9 \\
\dot{a_7} + \zeta_7 a_7 &= -\xi_{71} \left(a_1 a_6 + a_6 a_9\right) + \xi_{72} a_2 a_5 + \xi_{73} a_3 a_4 \\
\dot{a_8} + \zeta_8 a_8 &= \xi_{81} a_2 a_5 + \xi_{82} a_3 a_4 \\
\dot{a_9} + \zeta_9 a_9 &= \xi_{91} a_2 a_3 - \xi_{92} a_6 a_8
\end{aligned}
\tag{5.3}
$$

Where:

$$
\begin{aligned}
&\zeta_1 = \frac{\beta^2}{Re}, \quad \zeta_2 = \frac{1}{Re}\left(\frac{4\beta^2}{3} + \gamma^2\right), \quad \zeta_3 = \frac{\beta^2 + \gamma^2}{Re}, \\
&\zeta_4 = \frac{3\alpha^2 + 4\beta^2}{3Re}, \quad \zeta_5 = \frac{\alpha^2 + \beta^2}{Re}, \quad \zeta_6 = \frac{3\alpha^2 + 4\beta^2 + 3\gamma^2}{3Re}, \\
&\zeta_7 = \zeta_8 = \frac{\alpha^2 + \beta^2 + \gamma^2}{\mathrm{Re}}, \quad \zeta_9 = 9\zeta_1
\end{aligned}
$$

And:

$$
\begin{aligned}
\xi_{11} = \xi_{62} = \xi_{66} = \xi_{92} = \sqrt{\frac{3}{2}}\frac{\beta\gamma}{\kappa_{\alpha\beta\gamma}}, \quad \xi_{12} = \xi_{24} = \xi_{25} = \xi_{91} = \sqrt{\frac{3}{2}}\frac{\beta\gamma}{\kappa_{\beta\gamma}}, \\
\xi_{21} = \frac{5\sqrt{2}}{3\sqrt{3}}\frac{\gamma^2}{\kappa_{\alpha\gamma}}, \quad \xi_{22} = \frac{\gamma^2}{\sqrt{6}\kappa_{\alpha\gamma}}, \quad \xi_{23} = \xi_{53} = \frac{1}{\sqrt{6}}\kappa_2, \quad \xi_{31} = \xi_{55} = \frac{2}{\sqrt{6}}\kappa_1, \\
\xi_{32} = \frac{\beta^2\left(3\alpha^2 + \gamma^2\right) - 3\gamma^2\left(\alpha^2 + \gamma^2\right)}{\sqrt{6}\kappa_{\alpha\gamma}\kappa_{\beta\gamma}\kappa_{\alpha\beta}}, \quad \xi_{41} = \xi_{45} = \xi_{51} = \xi_{54} = \xi_{61} = \xi_{65} = \xi_{71} = \frac{\alpha}{\sqrt{6}}, \\
\xi_{42} = \frac{10\alpha^2}{3\sqrt{6}\kappa_{\alpha\gamma}}, \quad \xi_{43} = \sqrt{\frac{3}{2}}\kappa_1, \quad \xi_{44} = \sqrt{\frac{3}{2}}\frac{\alpha^2\beta^2}{\kappa_{\alpha\gamma}\kappa_{\beta\gamma}\kappa_{\alpha\beta\gamma}}, \quad \xi_{52} = \frac{\alpha^2}{\sqrt{6}}\kappa_{\alpha\gamma}, \\
\xi_{63} = \frac{10}{3\sqrt{6}}\frac{\alpha^2 - \gamma^2}{\kappa_{\alpha\gamma}}, \quad \xi_{64} = 2\sqrt{\frac{2}{3}}\kappa_1, \quad \xi_{72} = \frac{1}{\sqrt{6}}\frac{\gamma^2 - \alpha^2}{\kappa_{\alpha\gamma}}, \quad \xi_{73} = \frac{1}{\sqrt{6}}\kappa_1, \\
\xi_{81} = \frac{2}{\sqrt{6}}\kappa_2, \quad \xi_{82} = \frac{\gamma^2\left(3\alpha^2 - \beta^2 + 3\gamma^2\right)}{\sqrt{6}\kappa_{\alpha\gamma}\kappa_{\beta\gamma}\kappa_{\alpha\beta\gamma}}
\end{aligned}
$$

With:

$$
\kappa_{\alpha\gamma} = \sqrt{\alpha^2 + \gamma^2}, \quad \kappa_{\beta\gamma} = \sqrt{\beta^2 + \gamma^2}, \quad \kappa_{\alpha\beta\gamma} = \sqrt{\alpha^2 + \beta^2 + \gamma^2}
$$

And:

$$
\kappa_1 = \frac{\alpha\beta\gamma}{\kappa_{\alpha\gamma}\kappa_{\beta\gamma}}, \quad \kappa_2 = \frac{\alpha\beta\gamma}{\kappa_{\alpha\gamma}\kappa_{\alpha\beta\gamma}}
$$

The parameters of the system were equal to: $\alpha = 2\pi/L_x$; $\beta = \pi/2$; $\gamma = 2\pi/L_z$.

The size of the domain is set to $L_x = 4\pi$, $L_z = 2\pi$ for the whole thesis. The Reynolds number is determined to be 800 [37] to ensure a complete turbulent system. The initial condition is set to $(1.0, 0.07066, -0.07076, 0.001x, 0.0, 0.0, 0.0, 0.0, 0.0)$, where x is a random number between

0 and 1 [38]. Similar to the Lorenz system and the CDV system, with all the parameters and initial condition, MFE system data can be generated with the Runge-Kutta method. The time evolution of the MFE system is shown in Figure 5.5.

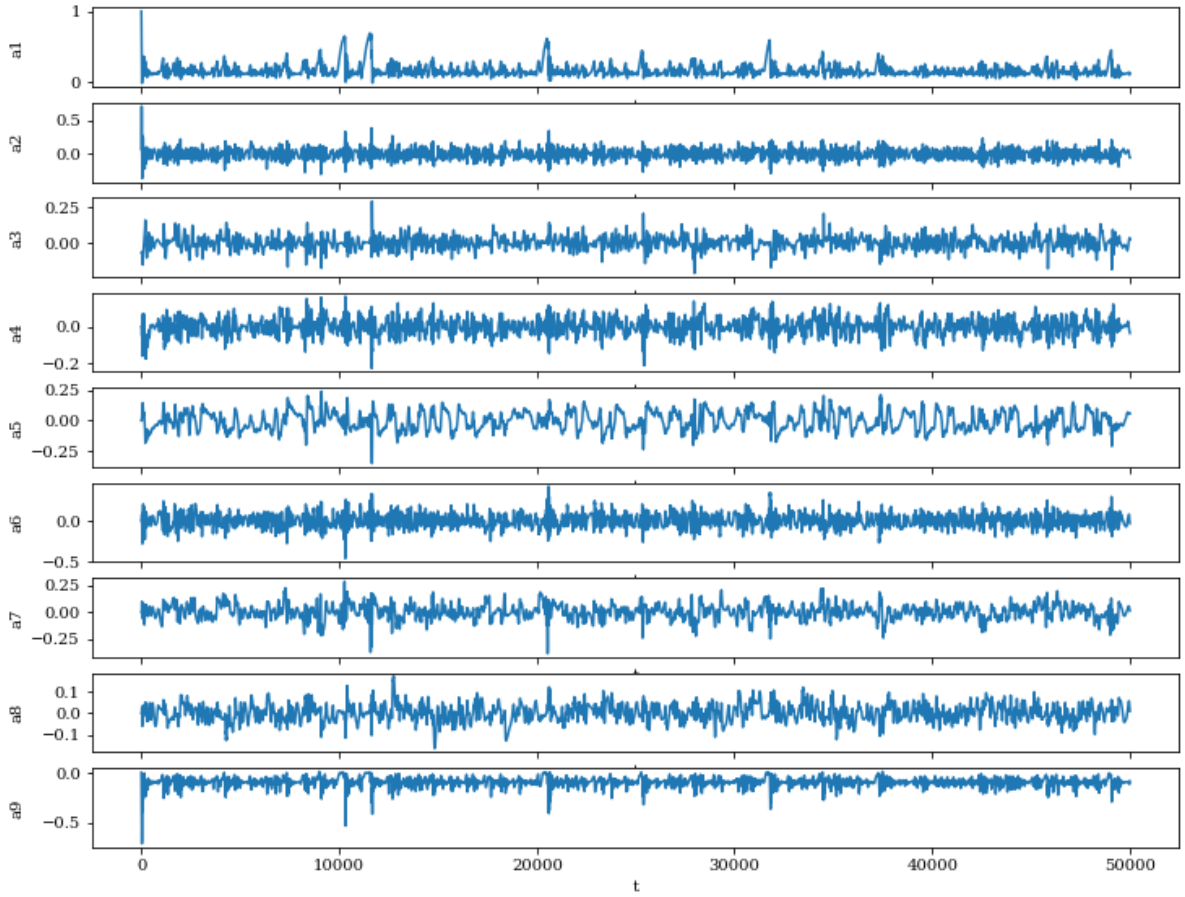


Figure 5.5: Time series of MFE system

However, the time evolution of the amplitude of each modes does not provide much information regarding the prevention of extreme events. Even identifying extreme events is quite hard when only given fluctuation on the nine modes' amplitude. Thus, another latent space is needed to identify and control the extreme events. The time evolution of kinetic energy k and mean dissipation rate ϵ are calculated to define where the extreme events happen. Sudden spikes in kinetic energy and dissipation rate are used as indicators of extreme events, as demonstrated in Figure 5.6.

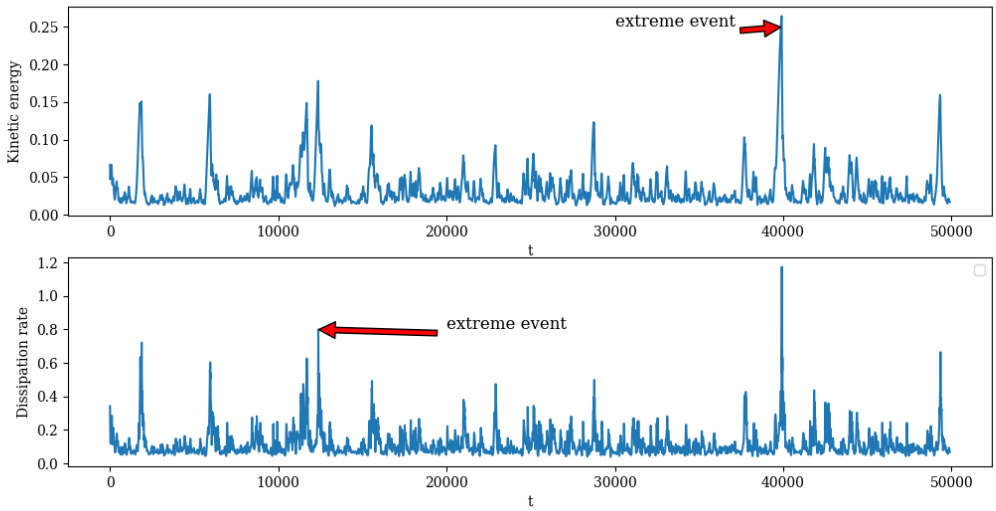


Figure 5.6: Extreme events in MFE system

Furthermore, the extreme events of the system can be more clearly seen by projecting the MFE system on to the $k - \epsilon$ phase space. Curves that are far from the origin with high kinetic energy and dissipation rate are the indicators of extreme events. This approach also provides an effective means to explicitly evaluate the performance of the control algorithm. Given that the objective of the MFE system is to reduce extreme events, a well-functioning control algorithm should ensure that the projection of the controlled system remains close to the origin, with minimal deviations or distant trajectories.

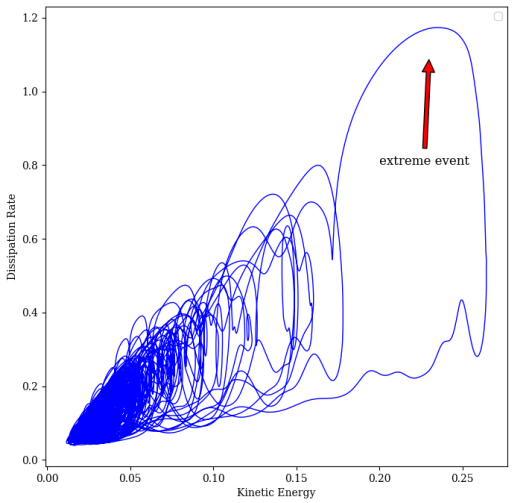


Figure 5.7: Extreme events in MFE system in phase space

6

Modeling Results

In this chapter, the results of reduced-order modeling are illustrated and discussed for all three systems. Additionally, the effect of two main modeling parameters, the number of clusters N_{cl} and the order of model L are studied for the CNM method. The results for these parameters are compared from both statistical and dynamical perspective. Furthermore, a general approach to determining the optimal number of clusters N_{cl} is proposed.

6.1. Lorenz System

6.1.1. CMM Results of the Lorenz System

$N_{cl} = 10$ is selected in this section to check if CMM can capture the features of the Lorenz system. The clustering result is shown in Figure 6.1. Each color-coded cluster represents distinct regions in the phase space, with the central black dots indicating the centroids of these clusters. The dense clustering near the attractor's lobes reflects the system's propensity to spend more time in these areas, effectively capturing the dynamic structure of the Lorenz attractor. These clusters can simplify the analysis and control of the chaotic system by focusing on significant states and their interactions, facilitating the creation of reduced-order models that approximate the complex dynamics of the Lorenz system.

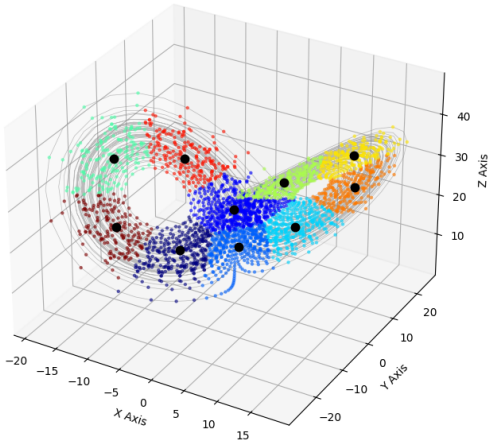


Figure 6.1: Clustering result of the Lorenz system

A simplified network graph of the Lorenz system when applying CMM method is shown in Figure 6.2. This represents the Lorenz system with reduced-order, where each node corresponds to a cluster identified in the system's state space, and directed edges indicate transitions between these clusters. The self-loops on nodes suggest that the system frequently returns to the same cluster. The connections between different nodes illustrate the dynamic pathways through which the system evolves over time, revealing the likely transitions between clusters. This graph-based representation simplifies the complex dynamics of the Lorenz system into a manageable structure, allowing for a clearer understanding of the system's behavior and facilitating the analysis of state transitions.

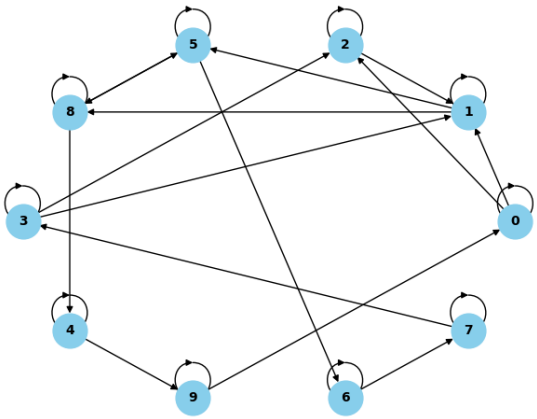


Figure 6.2: Network graph representation of the Lorenz system

The cluster transition and distance matrix are illustrated in Figure 6.3 and Figure 6.4. The cluster transition matrix (CTM) provides a quantitative perspective to study the transition between and within the clusters. Figure 6.3 displays the likelihood of transition from one cluster to another, with darker shades indicating higher probabilities. The diagonal dominance

in this matrix suggests that the system has a high probability of remaining in the same cluster, reflecting the system's tendency to persist in specific states. The off-diagonal elements, while generally lighter, indicate occasional transitions between different clusters, providing insight into the dynamics and flow of the Lorenz attractor. This is considered to be an inherent drawback of the CMM method. Because when predicting the system's time evolution with CTM, it is very likely that the system is stuck in the same cluster. The distance matrix, which is a diagonal matrix, shows the distances between each cluster in the phase space, with darker shades representing larger distances. This matrix reveals how distinct the clusters are from each other, indicating significant separation for certain pairs, which suggests that the clustering is successful.

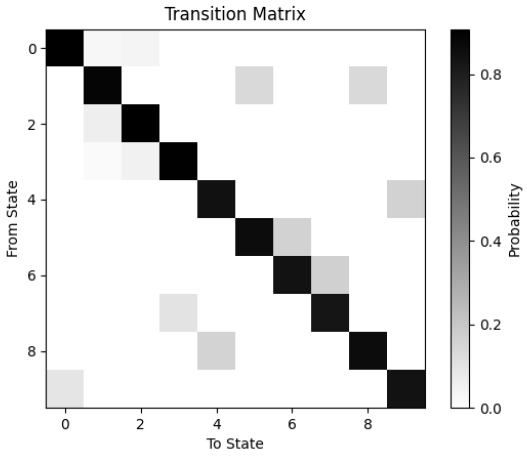


Figure 6.3: Transition matrix of Lorenz system

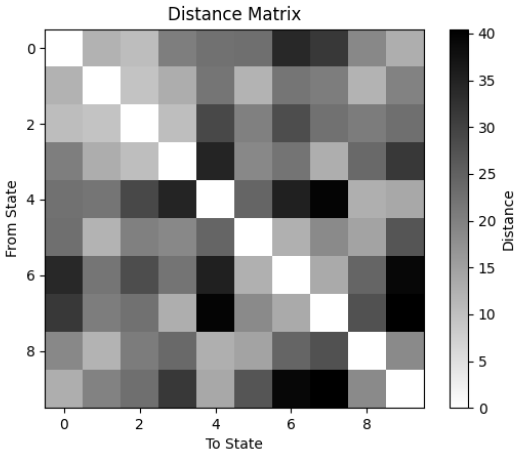


Figure 6.4: Distance matrix of Lorenz system

A comparison between the distribution of Lorenz system staying in each cluster and the distribution predicted by CMM model is shown in Figure 6.5. The red bars represent the distribution of samples in each cluster calculated from the original data, while the black bars represent the distribution predicted by the CTM over a sufficiently large number of time steps. The close alignment between the red and black bars indicates that the CMM model's predictions are quite successful from a statistic perspective. This agreement serves as a validation of the CMM model, demonstrating its reliability in predicting the proportion of time the system will spend in each cluster over a long period. From a practical standpoint, this successful prediction implies that the CMM model can be effectively used to approximate the long-term behavior of the Lorenz system, which is particularly useful in applications requiring an understanding of steady-state distributions in chaotic dynamics, such as climate modeling. However, while the statistic predictions are accurate, CMM cannot predict the state of the system in particular time, and can only suggests the centroids of each clusters as the state of the system.

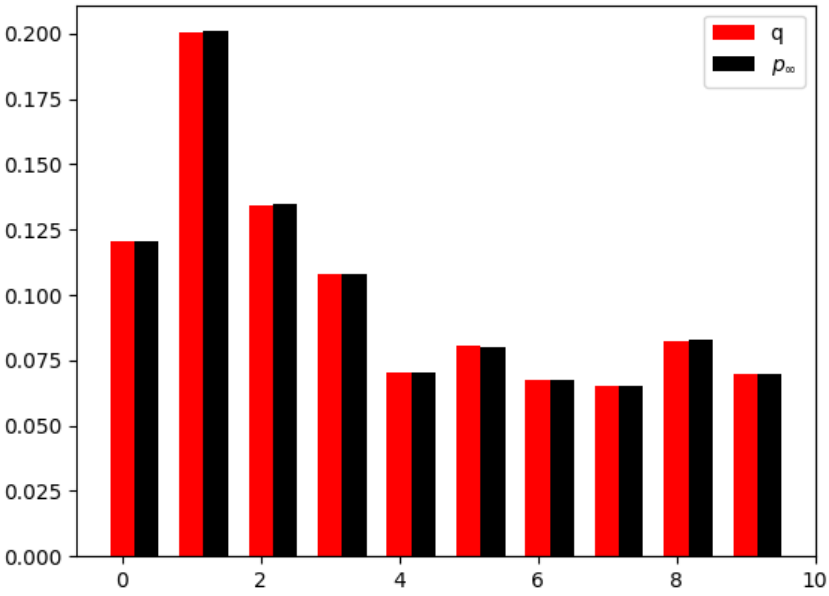


Figure 6.5: Accuracy of prediction for Lorenz system with CMM

6.1.2. CNM Results of the Lorenz System

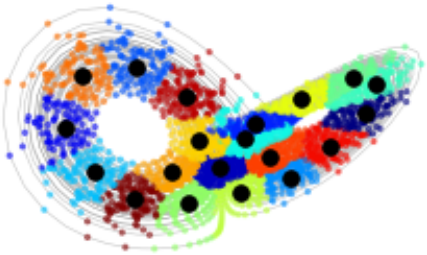


Figure 6.6: Original Lorenz system

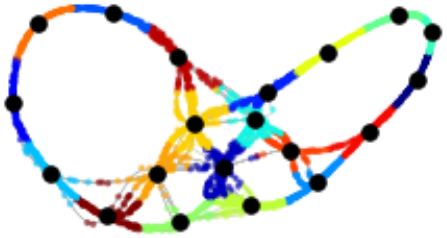


Figure 6.7: Reduced order Lorenz system with CNM

CNM can give prediction at any given time, allowing a time evolution of x, y and z in the reduced-order Lorenz system to be obtained. An illustration of the reduced-order model using CNM is shown in Figure 6.7. Compared to the original clustered system shown in Figure 6.6, the reduced-order system simplifies the complex dynamics of the Lorenz system by reducing it to transitions between centroids of clusters. This model utilizes interpolation to generate sample data at any given time point, creating smooth transitions between clusters rather

than direct jumps. The edges connecting these centroids indicate the paths taken by the system. This approach ensures that intermediate states are captured, showing the system's position between two centroids over time. By using these interpolated transitions, the CNM model captures the essential dynamics of the Lorenz system in a more computationally efficient manner, making it easier to analyze and predict the system's behavior, although simplifying some features.

The x,y and z evolution in the original Lorenz system and the reduced-order Lorenz system with CNM are shown in Figure 6.8 and Figure 6.9. Although there is a noticeable phase shift between the original and the reduced-order model, the overall prediction by the CNM model is quite accurate. The red curves closely follow the general pattern and amplitude of the black curves, demonstrating that the CNM model effectively captures the essential dynamics of the Lorenz system. Despite the phase differences, the reduced-order model maintains the integrity of the system's behavior, reflecting its capability to accurately predict the chaotic evolution of the Lorenz attractor.

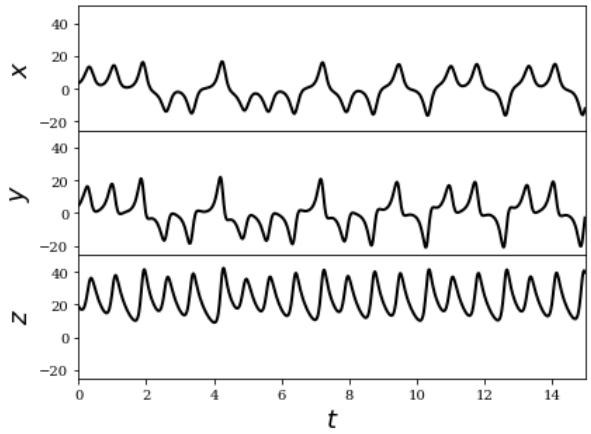


Figure 6.8: x,y and z evolution in original Lorenz system

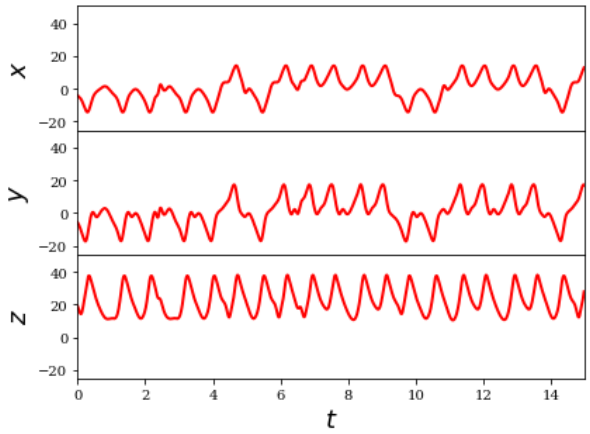


Figure 6.9: x,y and z evolution in reduced-order Lorenz system with CNM

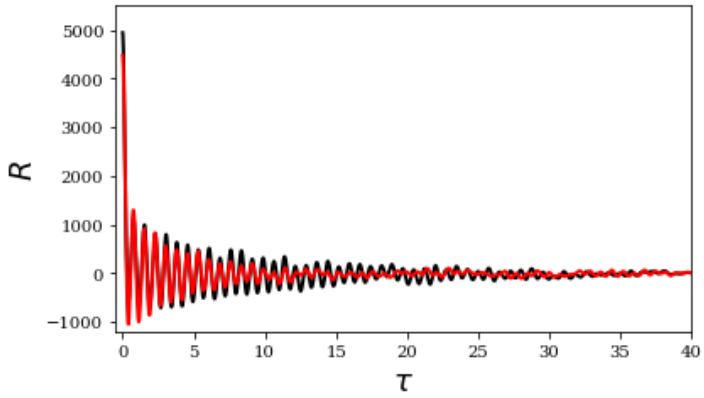
6.1.3. Discussion

Two main parameters are studied here to study their effects on the model: the order of the model and the number of clusters, both of which are evaluated from a dynamical perspective using auto-correlation function and from statistic perspective using distribution of x, y and z. Specially, auto-correlation function reads

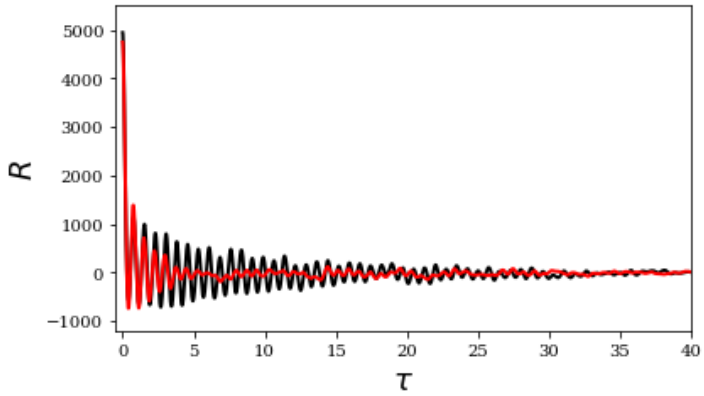
$$R(\tau) := \frac{1}{T-\tau}\int_{\tau}^{\mathrm{T}}\int_{\Omega}\boldsymbol{u}(\boldsymbol{x},t-\tau)\cdot\boldsymbol{u}(\boldsymbol{x},t)\mathrm{d}\boldsymbol{x}\mathrm{d}t, \quad \tau\in[0,T) \tag{6.1}$$

This function quantifies the similarity or correlation between data points in a time series as a function of time lags. It measures how much the system is related to itself at different time lags, which can detect periodic patterns in a time series. If the series has a periodic component, the auto-correlation function will show peaks at time lags corresponding to the period of the signal. This property of auto-correlation function helps to reveal the dynamical properties of the system, excluding the influence of phase shift.

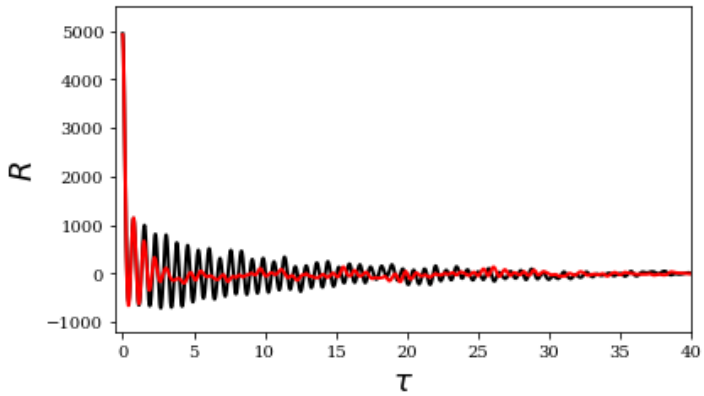
Number of clusters



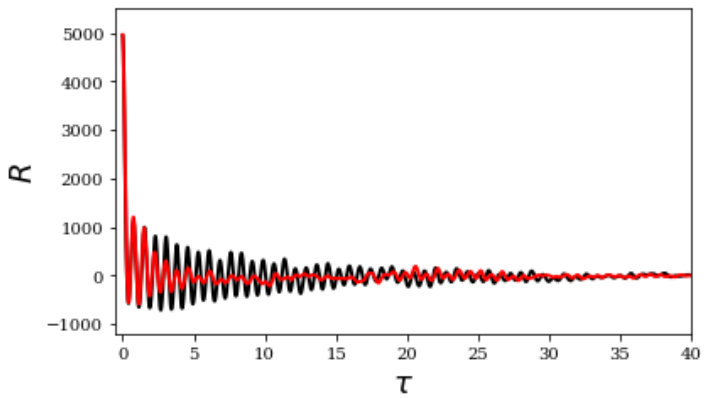
(a) Auto-correlation for Lorenz system when $N_{CL} = 10$ and $L = 3$



(b) Auto-correlation for Lorenz system when $N_{CL} = 20$ and $L = 3$

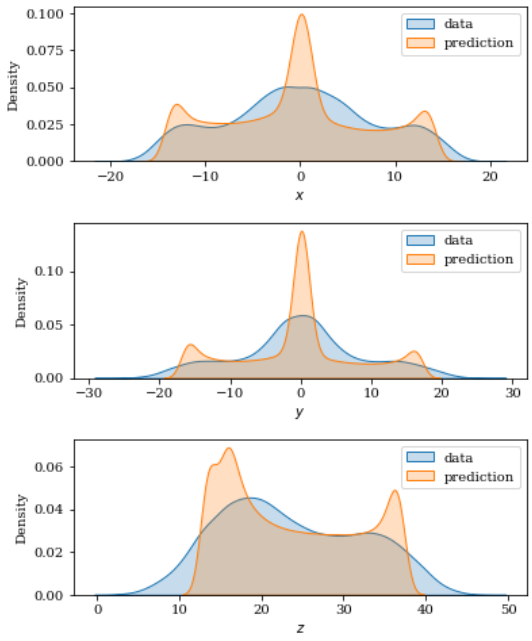


(c) Auto-correlation for Lorenz system when $N_{CL} = 50$ and $L = 3$

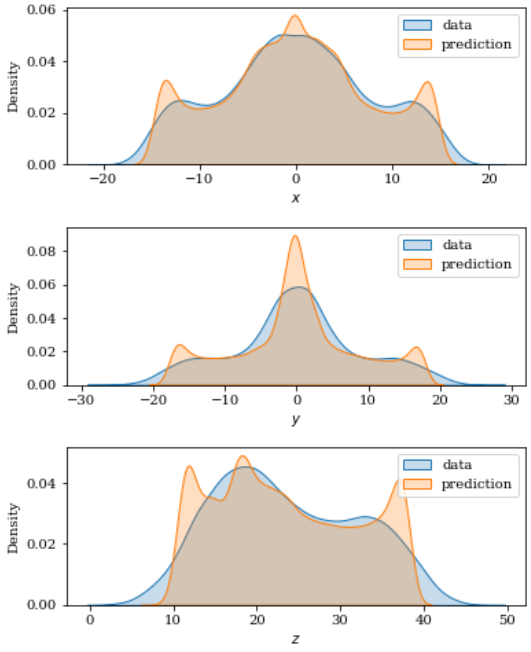


(d) Auto-correlation for Lorenz system when $N_{CL} = 100$ and $L = 3$

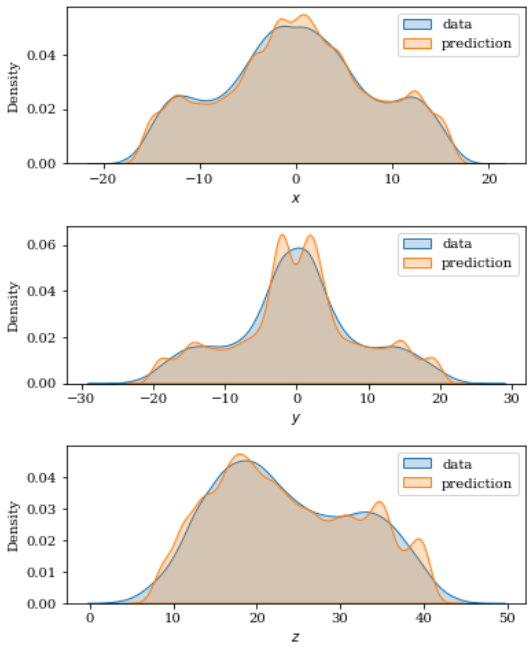
Figure 6.10: Comparison of Auto-correlation function when $N_{cl} varies$ for Lorenz system



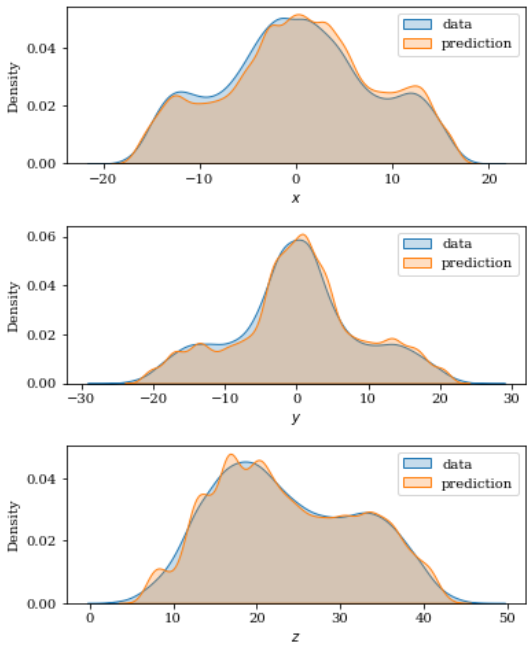
(a) Distribution for Lorenz system when $N_{CL} = 10$ and $L = 3$



(b) Distribution for Lorenz system when $N_{CL} = 20$ and $L = 3$



(c) Distribution for Lorenz system when $N_{CL} = 50$ and $L = 3$



(d) Distribution for Lorenz system when $N_{CL} = 100$ and $L = 3$

Figure 6.11: Comparison of distribution for Lorenz system when N_{cl} varies

Figure 6.10 shows the auto-correlation functions of the reduced-order Lorenz system using the CNM model with different numbers of clusters (10, 20, 50, and 100) while keeping the model order fixed at 3. The black curves in each sub-figures represent the uncontrolled Lorenz system, serving as a reference.

It can be observed that in each sub-figures, the red curve (reduced-order Lorenz system) shows high consistency with the black curve (original system) in terms of the auto-correlation function

for small time shift. This indicates that, regardless of the number of clusters, the reduced-order Lorenz system effectively captures the dynamic characteristics of the original system in the short term. The good fit in the short term indicates that the CNM model is effective in capturing the initial response and short-term dynamics of the system. When comparing the sub-figures with each other, increasing the number of clusters N_{cl} does not improving the CNM model's capability to capture the dynamical properties of Lorenz system. On the contrast, the consistency of the predicted system and original system with larger number of clusters becomes worse compared to the case of $N_{cl} = 10$. This suggests that simply increasing the number of clusters does not enhance the model's dynamical predictive capability.

Figure 6.11 illustrates the statistic distribution of x, y and z of the reduced-order Lorenz system using the CNM model with different numbers of clusters (10, 20, 50, and 100) while keeping the model order fixed at 3. The blue shaded areas serve as a reference for comparison, representing the distribution of the original Lorenz system. In Figure 6.11a with 10 clusters, the noticeable peaks on the orange curves (predictions), which represent the centroids of the clusters, show deviations from the blue curves (actual data) across all three variables, indicating that the model lacks the ability to accurately capture the statistic distribution of the system's states from a continuous perspective. Figure 6.11b with 20 clusters shows an improved alignment, but there are still noticeable discrepancies, particularly in the distributions of the variable z. Figure 6.11c with 50 clusters demonstrates a much closer match, with the predicted distributions aligning well with the actual distributions, indicating that the model now has sufficient detail to capture the statistic properties of the system. Figure 6.11d with 100 clusters shows the closest alignment, with the orange curves almost perfectly overlapping with the blue curves, suggesting that the model can accurately predict the statistic distribution of the system's states. This progression indicates that increasing the number of clusters enhances the CNM model's ability to make accurate statistic predictions, allowing it to capture finer details in the distribution.

It is intuitive that infinitely increasing the number of clusters will make the predicted distribution more closely match the original system, as this approach would converge to a model where each cluster contains only one sample, effectively becoming the original system itself. Thus, one approach to quantitatively determine the optimal number of clusters is proposed. The mean variance of the samples in each cluster and the the centroid of the cluster is given by

$$J_m = \frac{1}{N_{cl}} \sum_{k=1}^{N_{cl}} \frac{1}{N_k} \sum_{s \in \mathscr{C}_k} \|s - c_k\|^2 \tag{6.2}$$

Figure 6.12 illustrates the relationship between the number of clusters and the J_m for the Lorenz system. As the number of clusters increases, J_m decreases, reflecting a finer representation of the data. However, the rate of decrease is not uniform. Initially, there is a steep decline in J_m as the number of clusters increases. Beyond 50 clusters, the reduction in J_m becomes much less pronounced, indicating diminishing returns in model improvement. Therefore, the optimal number of clusters N_{cl} is selected at around 50. This point balances model complexity and accuracy, ensuring that the clusters are sufficiently detailed to capture the system's features without unnecessary computational cost.

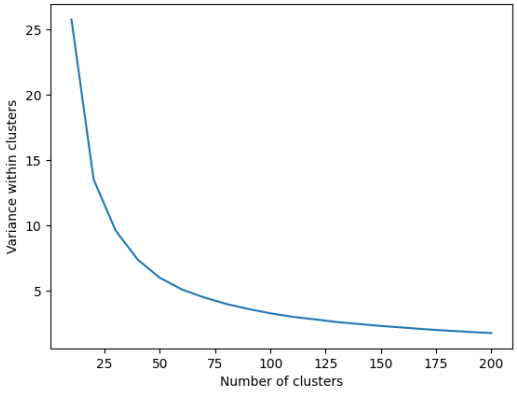


Figure 6.12: J_m of reduced-order Lorenz system when N_{cl} increases

Order of model

Figure 6.13 and Figure 6.14 shows the auto-correlation functions of the reduced-order Lorenz system using the CNM model with order of model L equals to 3 and 20 while keeping the number of clusters N_{cl} fixed at 20. The black curves in each sub-figures represent the uncontrolled Lorenz system, serving as a reference.

In the first plot, with $L = 3$, the red curve (CNM model) closely follows the black curve initially but deviates more significantly as τ increases. This indicates that the lower-order model captures the short-term dynamics well but struggles with longer-term predictions.

In the second plot, with $L = 20$, the red curve shows a much closer alignment with the black curve across the entire range of τ. The higher-order model maintains the accuracy of the short-term predictions while also significantly improving the long-term dynamics representation.

This comparison suggests that increasing the order of the model enhances the CNM model's ability to capture the repeated dynamical properties of the Lorenz system, providing a more accurate and reliable representation of both short-term and long-term dynamics. Thus, while a lower-order model may suffice for capturing immediate behavior, a higher-order model is necessary for a comprehensive and precise depiction of the system's dynamics.

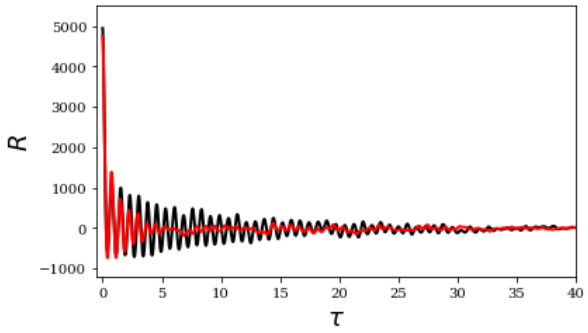


Figure 6.13: Auto-correlation for Lorenz system when $N_{CL} = 20$ and $L = 3$

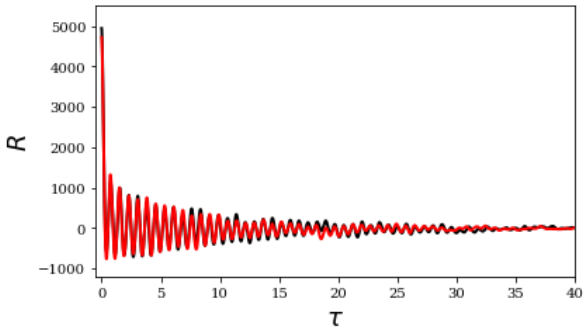


Figure 6.14: Auto-correlation for Lorenz system when $N_{CL} = 20$ and $L = 22$

Figure 6.15 and Figure 6.16 illustrates the statistic distributions of x, y and z of the reduced-order Lorenz system using the CNM model with order of model L equals to 3 and 22 while

keeping the number of clusters N_{cl} fixed at 20. The blue shaded areas serve as a reference for comparison, representing the distribution of the original Lorenz system. The results seems rather similar, indicating that when predicting the statistic distributions, increasing the model order alone does not significantly improve accuracy.

As a conclusion, increasing the order of model can be beneficial to capturing dynamical properties, but cannot increase the accuracy when predicting the possibility of the state.

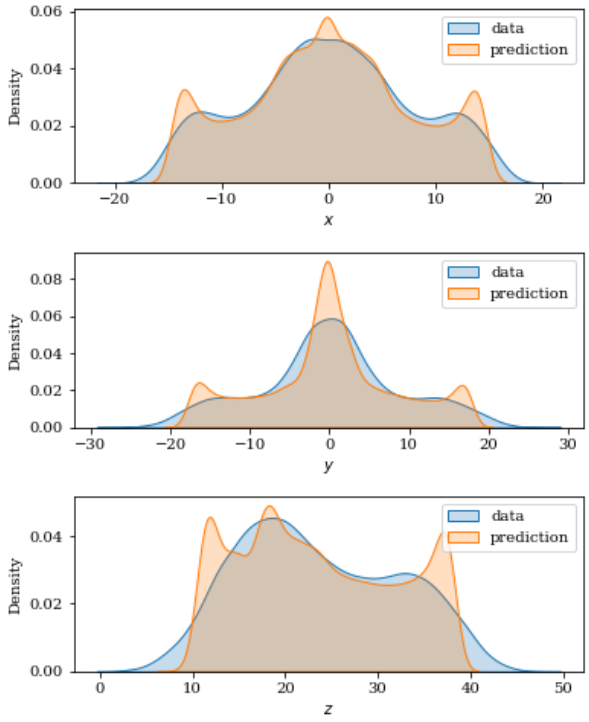


Figure 6.15: Distribution for Lorenz system when $N_{CL} = 20$ and $L = 3$

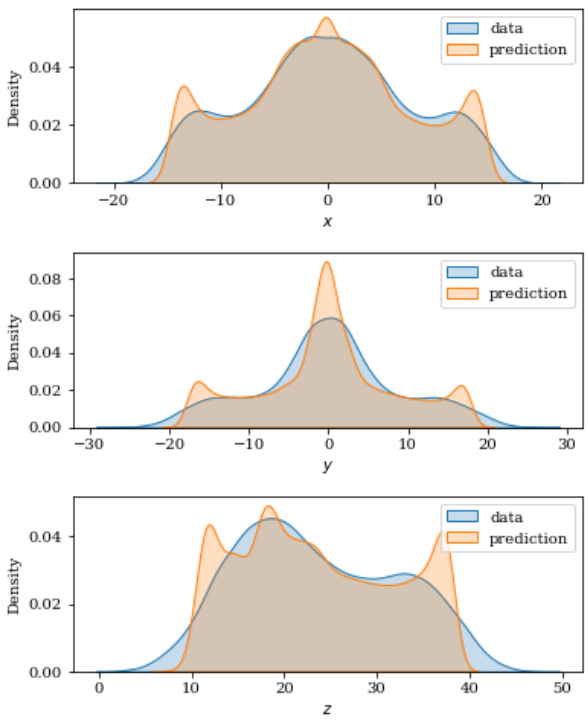


Figure 6.16: Distribution for Lorenz system when $N_{CL} = 20$ and $L = 22$

6.2. Truncated Charney-DeVore System

6.2.1. CMM Results of the CDV System

As discussed above, the same approach to determine the optimal number of clusters N_{cl} can be applied to the CDV system as well. The relation of J_m versus N_{cl} is shown in Figure 6.17. Following the same standard as Lorenz system, it can be observed that the reduction in J_m becomes much less pronounced after N_{cl} exceeds 50, indicating diminishing returns in model improvement. Therefore, the optimal number of clusters N_{cl} is selected at 50, which balances model complexity and accuracy.

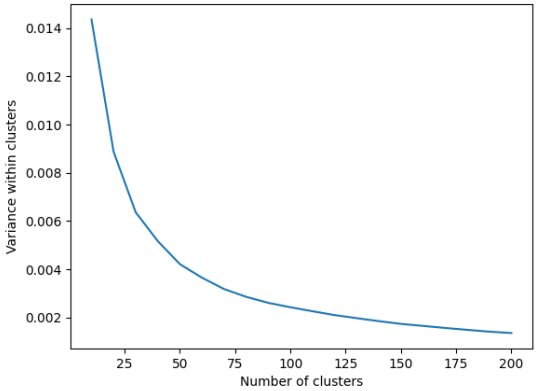


Figure 6.17: J_m of the reduced-order CDV system when N_{cl} increases

$N_{cl} = 50$ is selected in this section to check if the CMM can capture the features of the CDV system. A simplified network graph of the CDV system when applying the CMM method is shown in Figure 6.18. Each node represents a cluster in the system's phase space, and the directional edges indicate how the system transitions between these clusters. The self-loops on nodes suggest that the system frequently revisits the same cluster. This graph helps simplify and visualize the complex behavior of the CDV system, making it easier to understand how the system moves from one state to another.

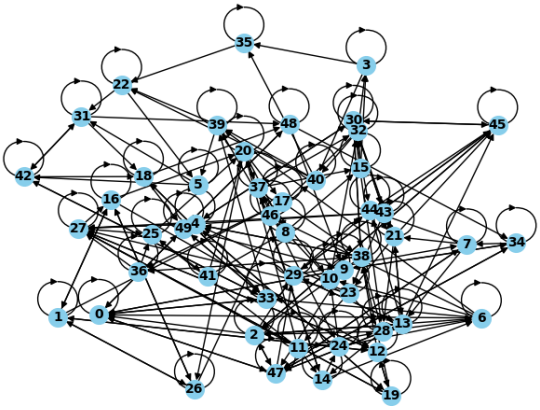


Figure 6.18: Network graph representation of the CDV system

The cluster transition and distance matrix are illustrated in Figure 6.19 and Figure 6.20, which have the same definitions describe in the section of the Lorenz system. Similar to the Lorenz system CTM of the reduced-order CDV system also shows strong diagonal dominance which suggests that the system tends to stay in the same cluster rather than transitioning to another cluster. The distance matrix shows that the clustering effectively distinguishes distinct regions in the CDV system's state space.

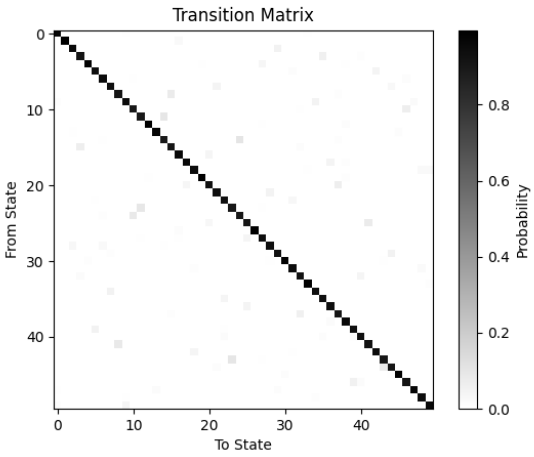


Figure 6.19: Transition matrix of the CDV system

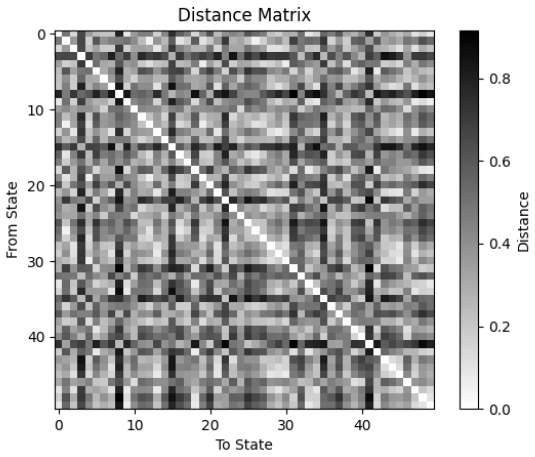


Figure 6.20: Distance matrix of the CDV system

A comparison between the distribution of the CDV system staying in each cluster and the distribution predicted by CMM model is shown in Figure 6.21. The red bars represent the distribution of samples in each cluster calculated from the original data, while the black bars represent the distribution predicted by the CTM. The close alignment between the red and black bars suggests that the CMM model's predictions are statistically accurate. However, as discussed in the context of the Lorenz system, the CMM model cannot predict the exact state of the system at a specific time and can only identify the centroids of each cluster as the system's state.

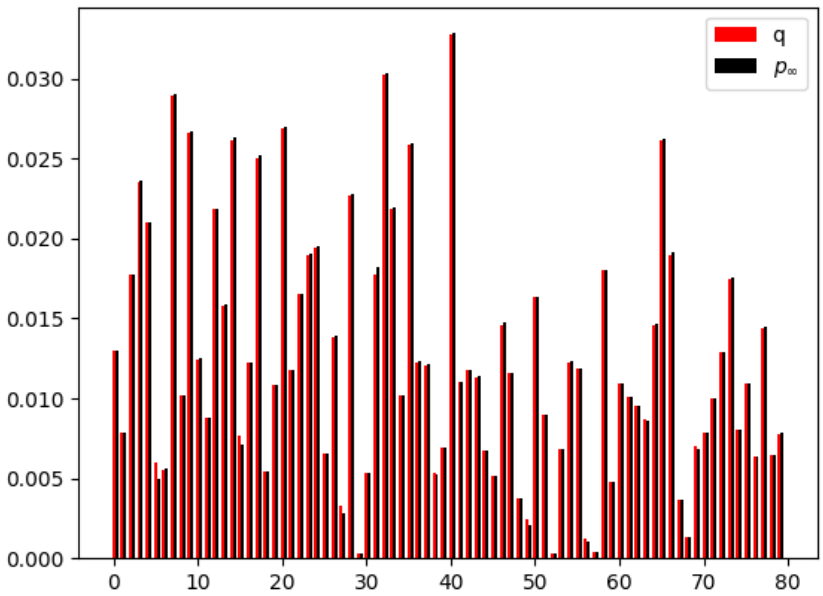
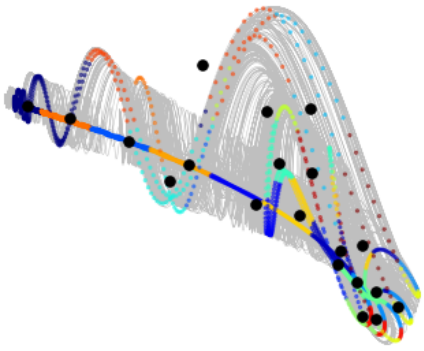
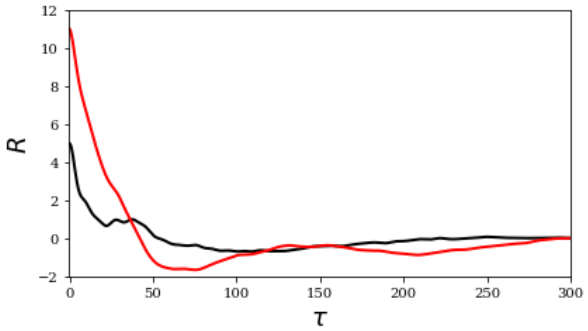


Figure 6.21: Accuracy of prediction for the CDV system with CMM

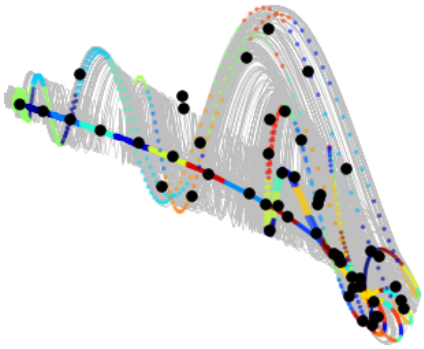
6.2.2. CNM Results of the CDV System



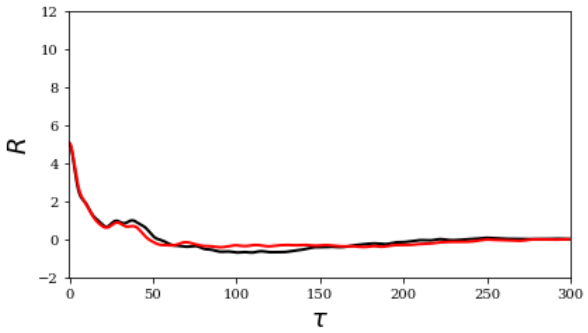
(a) $x_1 - x_4$ projection for the CDV system when $N_{cl} = 20$ and $L = 3$



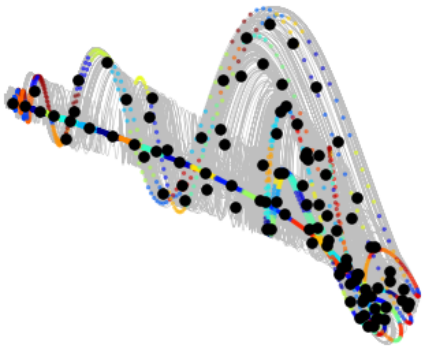
(b) Auto-correlation for the CDV system when $N_{cl} = 20$ and $L = 3$



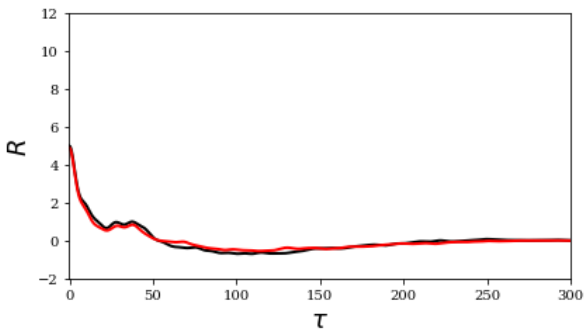
(c) $x_1 - x_4$ projection for the CDV system when $N_{cl} = 50$ and $L = 3$



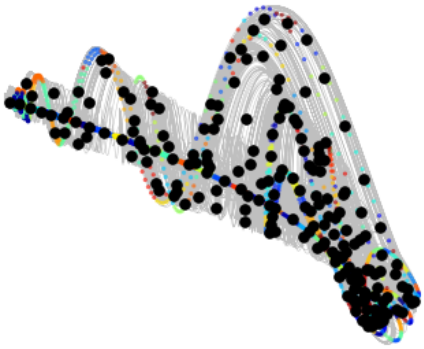
(d) Auto-correlation for the CDV system when $N_{cl} = 50$ and $L = 3$



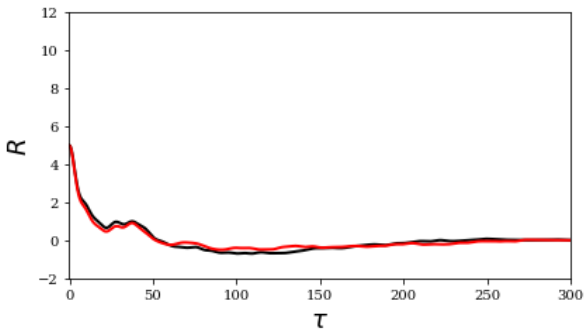
(e) $x_1 - x_4$ projection for the CDV system when $N_{cl} = 100$ and $L = 3$



(f) Auto-correlation for the CDV system when $N_{cl} = 100$ and $L = 3$



(g) $x_1 - x_4$ projection for the CDV system when $N_{cl} = 200$ and $L = 3$



(h) Auto-correlation for the CDV system when $N_{cl} = 200$ and $L = 3$

Figure 6.22: Results of phase space clustering and auto-correlation for CDV system

The results are projected on the $x_1 - x_4$ plane to visualized the high dimensional system. The clustering results are be found in Figure 6.22 with different numbers of clusters (20, 50, 100, and 200). Each cluster is represented by different colors and centroids are marked with black dots.

Figure 6.22b, Figure 6.22d, Figure 6.22f, and Figure 6.22h shows the corresponding auto-correlation functions, with the black curves representing the original CDV system and the red curves representing the predictions from the reduced-order model. While the auto-correlation function initially shows some improvement in accuracy with an increased number of clusters (from 20 to 50), further increases (100 and 200 clusters) do not significantly improve the alignment between the model and the original system. This suggests that there is a diminishing return in terms of dynamic accuracy when simply increasing the number of clusters. As a conclusion, for this complex dynamical system, using too few clusters will lead to inaccurate prediction. However, beyond a certain point, increasing the number of clusters does not necessarily lead to better model performances. Therefore, an optimal number of clusters is required to balances model complexity and predictive accuracy, which appears to be around 50 in this case. This value also aligns with the optimal N_{cl} that we obtained using the J_m plot.

The time evolution of x_1, x_2, x_3, x_4, x_5, and x_6 in the original CDV system and the CDV modelled with CNM with different numbers of clusters (20, 50, 100, and 200), while keeping the model order L fixed at 3. As the number of clusters increases, the time series generated by the reduced-order model show increasingly similarity to the original system. When N_{cl} is 20, many unnatural peaks appears in the time series of $\{x\}$. When N_{cl} reaches 50, the evolution of $\{x\}$ seems quite similar to the original CDV system. However, further increasing the number of clusters N_{cl} does not significantly improve the performance of the model. This result also aligns with the optimal number of clusters that we determined with J_m.

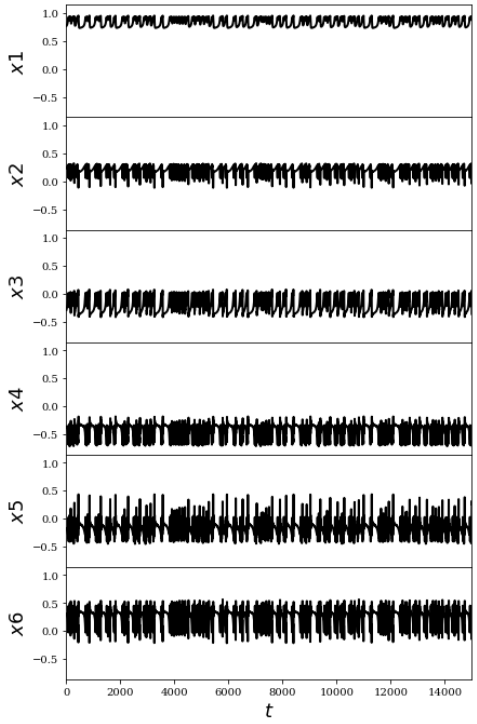


Figure 6.23: Time series for the CDV system without modeling

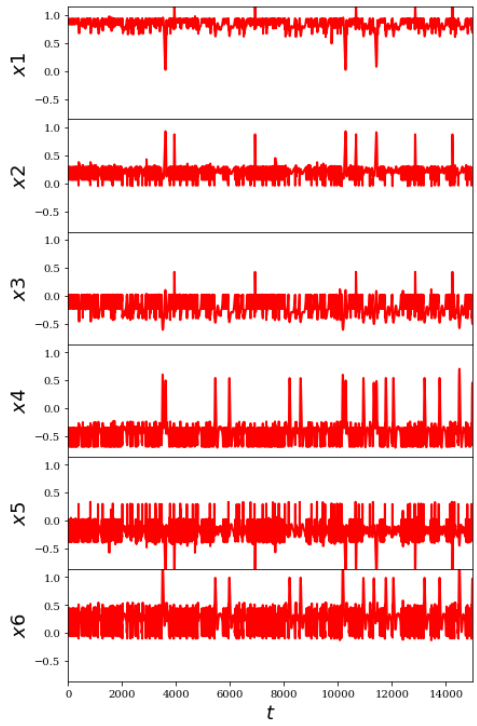


Figure 6.24: Time series for the CDV system when $N_{cl} = 20$ and $L = 3$

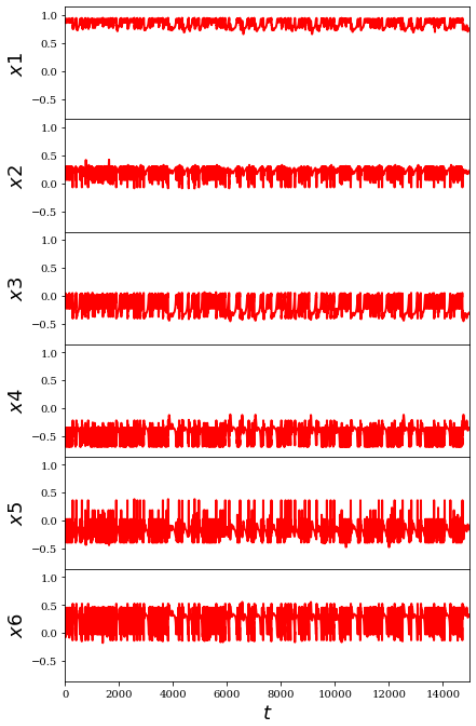


Figure 6.25: Time series for the CDV system when $N_{cl} = 50$ and $L = 3$

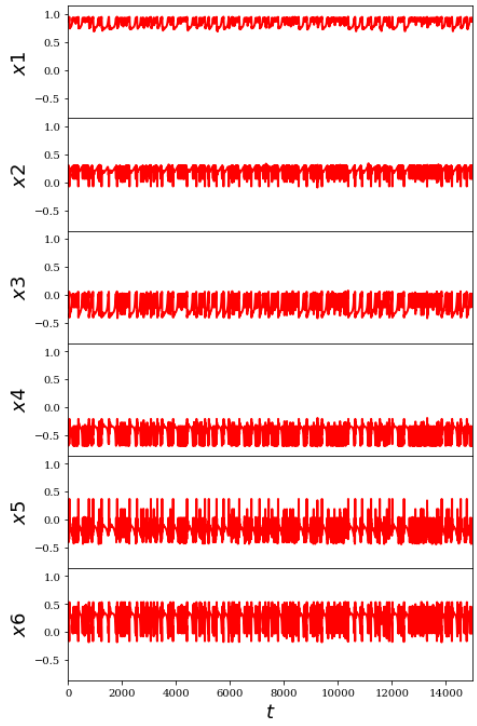


Figure 6.26: Time series for the CDV system when $N_{cl} = 100$ and $L = 3$

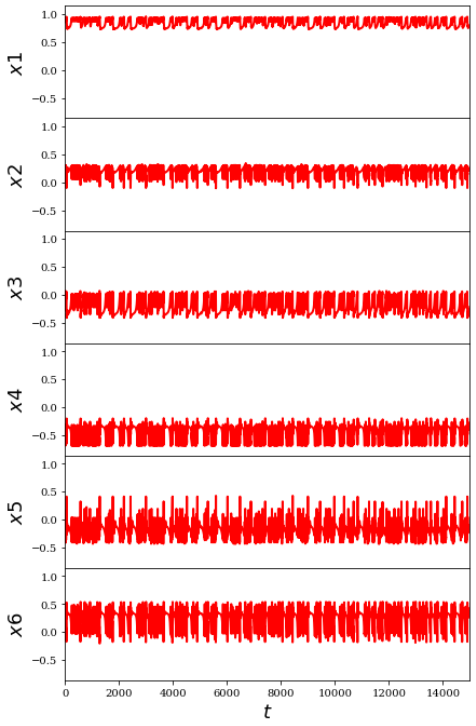


Figure 6.27: Time series for the CDV system when $N_{cl} = 200$ and $L = 3$

Figure 6.28 and Figure 6.29 illustrate the auto-correlation function of the modelled and the

original CDV system when N_{cl} is fixed at 50, and L changes (3 and 20). The corresponding time evolution of $/x/$ can be found in Figure 6.31 and Figure 6.32.

The auto-correlation plots show that when $L = 3$, the red curve (modelled system) deviates significantly from the black curve (original system) as time progresses, indicating that the model struggles to accurately capture the system's dynamics. However, with $L = 20$, the red curve aligns much more closely with the black curve, suggesting that the higher-order model can better replicate the temporal correlations of the original CDV system.

In terms of the time series, the plots show that with $L = 3$, the time series of the reduced-order CDV (red) fails to capture the tendency seen in the original system (black). Several unexpected peaks appears in time series of x_2 and x_4. When the model order is increased to $L = 20$, the time series generated by the model more closely matches the original system, showing better alignment in terms of the variability and overall behavior of the system. Though there is still some unexpected peaks, the amount reduces.

The results suggest that increasing the order of modeling, both results in auto-correlation function and time series are improved. As a conclusion, the order of modeling needs to be carefully determined, especially for high dimensional systems.

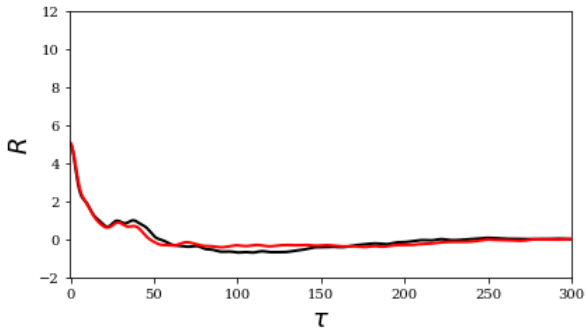


Figure 6.28: Auto-correlation for the CDV system when $N_{cl} = 50$ and $L = 3$

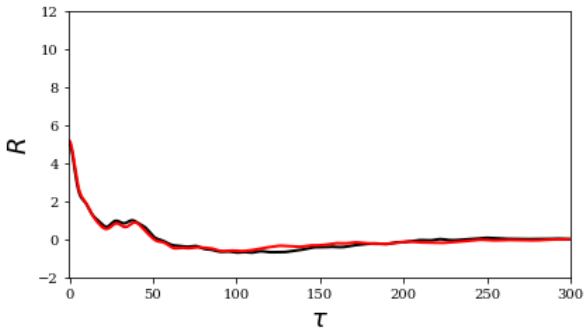


Figure 6.29: Auto-correlation for the CDV system when $N_{cl} = 50$ and $L = 20$

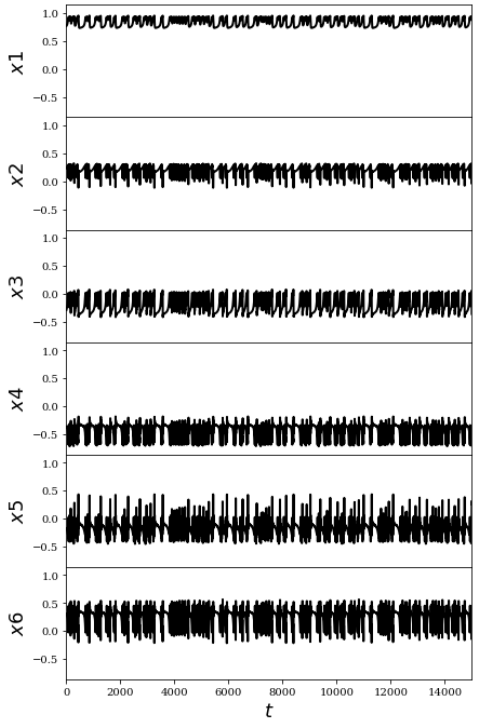


Figure 6.30: Time series for the CDV system without modeling

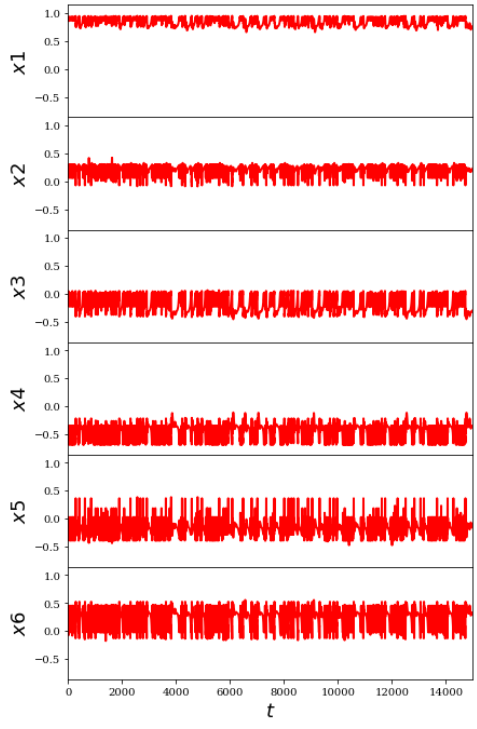


Figure 6.31: Time series for the CDV system when $N_{cl} = 50$ and $L = 3$

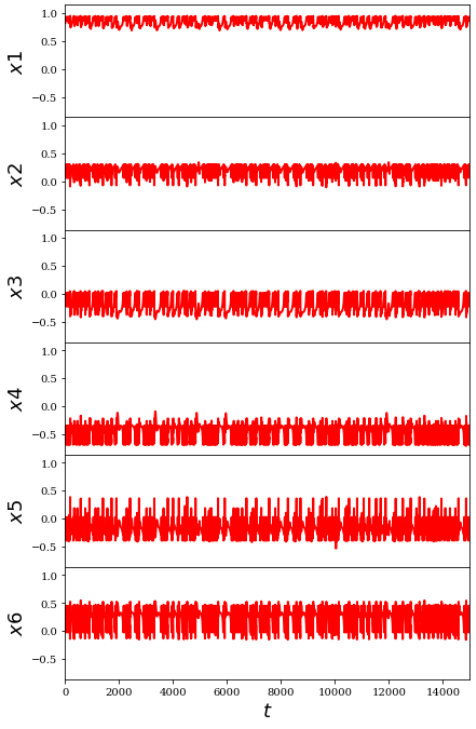


Figure 6.32: Auto-correlation for the CDV system when $N_{cl} = 50$ and $L = 20$

6.3. Moehlis-Faisst-Eckhart System

6.3.1. CMN Results of the MFE System

Based on the same approach, an optimal number of clusters N_{cl} is determined for the MFE system. The relation of J_m versus N_{cl} for MFE system is shown in Figure 6.33. Following the same standard, it can be observed that the reduction in J_m becomes much less pronounced after N_{cl} exceeds 80, indicating diminishing returns in model improvement. Therefore, the optimal number of clusters N_{cl} is selected at around 80, which balances model complexity and accuracy. This number of clusters N_{cl} is also used for further cluster-based control.

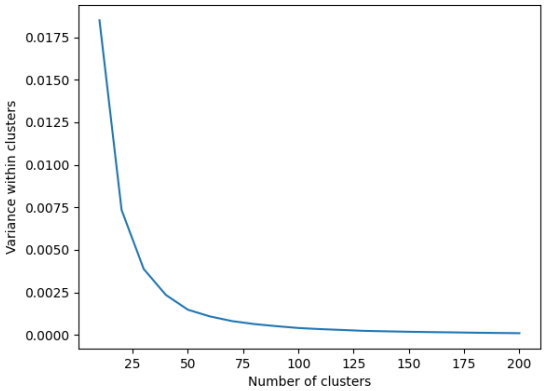


Figure 6.33: J_m of the reduced-order MFE system when $N_{cl}changes$

$N_{cl} = 80$ is selected in this section to evaluate whether the CMM method can capture the features of the MFE system. A simplified network graph of the MFE system, generated using the CMM method, is shown in Figure 6.34.

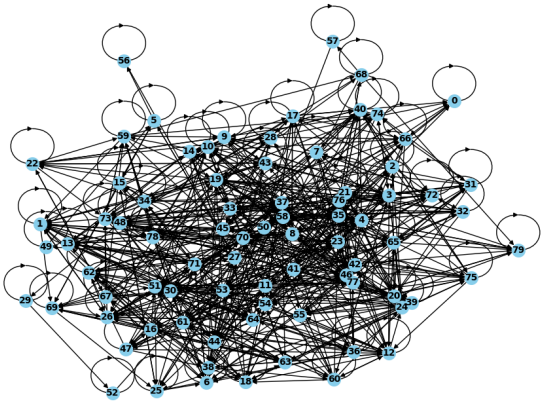


Figure 6.34: Network graph representation of the MFE system

The cluster transition and distance matrix are shown in Figure 6.35 and Figure 6.36. Due to strong diagonal dominance, logarithmic scale is applied to well illustrate the transition matrix. Similar conclusion as discussed in sections for the Lorenz system and the CDV system can be derived. The MFE system also has very high probability to stay in the same cluster. The distance matrix suggests that the clustering effectively distinguishes distinct region in the MFE system's phase space.

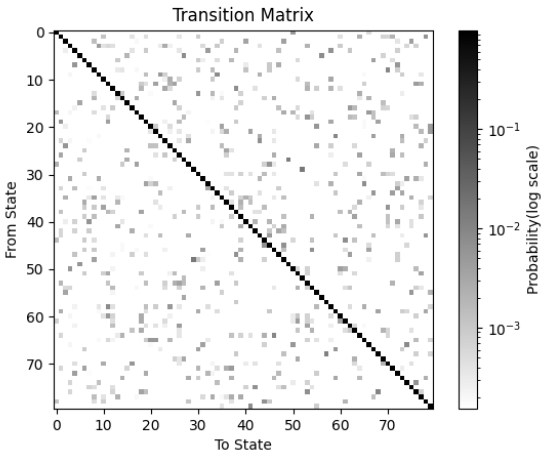


Figure 6.35: Transition matrix of the MFE system

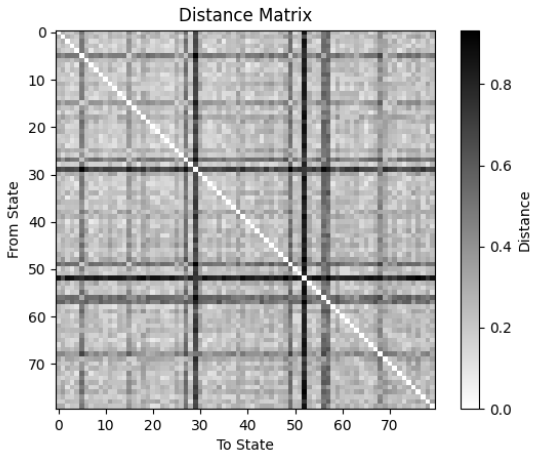


Figure 6.36: Distance matrix of the MFE system

Figure 6.37 presents a comparison between the actual distribution of the MFE system across clusters and the distribution predicted by the CMM model. The red bars show the distribution of samples within each cluster based on the original data, while the black bars represent the distribution predicted by the CMM model over a sufficiently large amount of time steps. The strong agreement between the red and black bars demonstrates that the CMM model is statistically accurate in its predictions. This outcome validates the clustering process and supports the implementation of subsequent control algorithms.

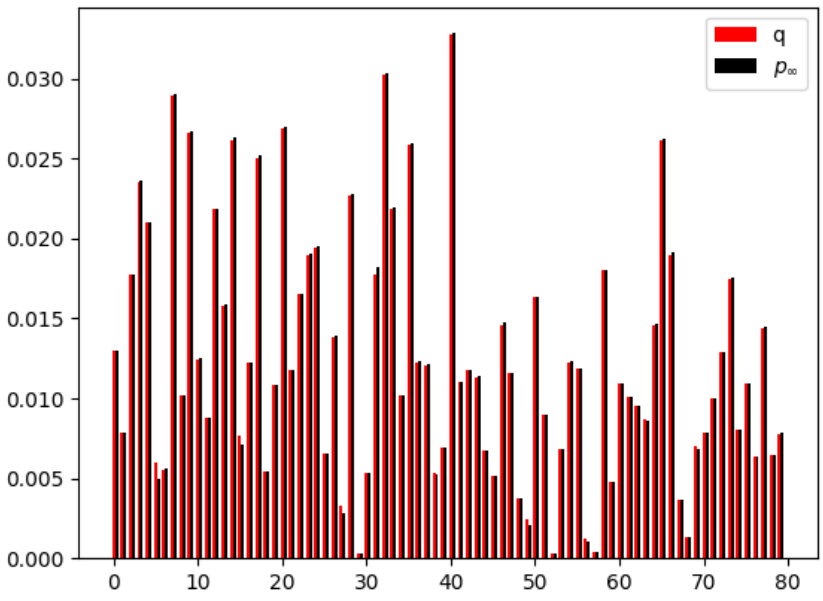
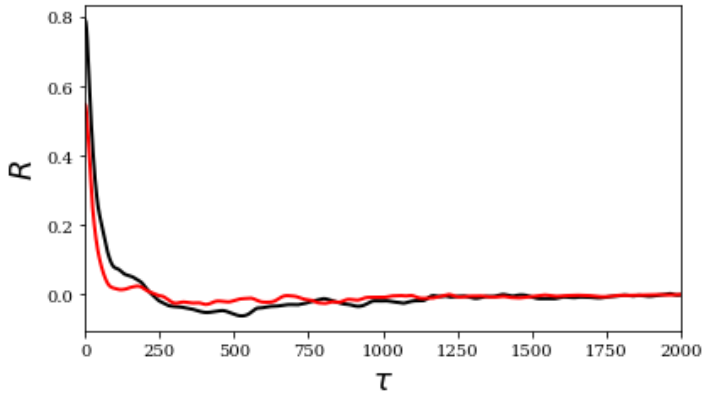


Figure 6.37: Accuracy of prediction for the MFE system with CMM

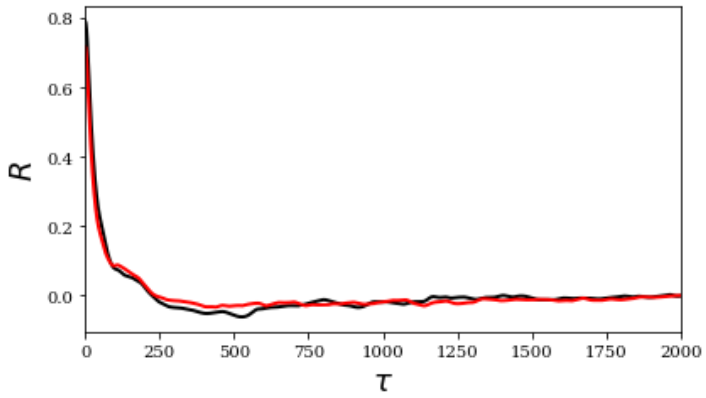
6.3.2. CNM Results of the MFE System

CNM method is also proved to be successful in a more complex system as the MFE system. And same trends for order of model and number of clusters can be found for the MFE system as well.

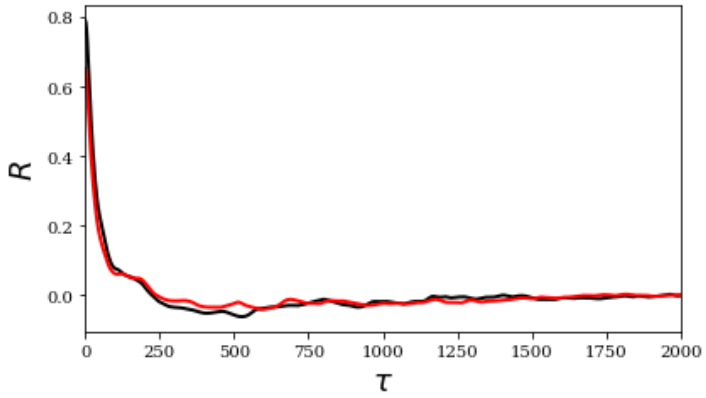
The auto-correlation function for fixed order of model and different number of clusters can be found in Figure 6.38. The results indicated that increasing the number of clusters can improve the model's ability to capture the dynamical properties, however, after exceeding a certain value, this improvement becomes negligible considering its cost.



(a) Auto-correlation for the MFE system when $N_{CL} = 40$ and $L = 3$



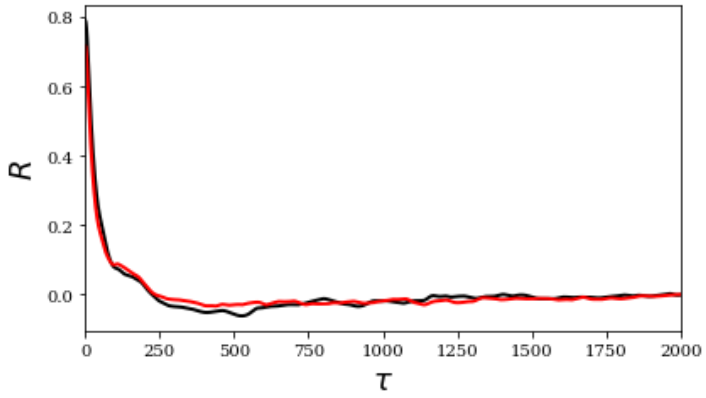
(b) Auto-correlation for the MFE system when $N_{CL} = 80$ and $L = 3$



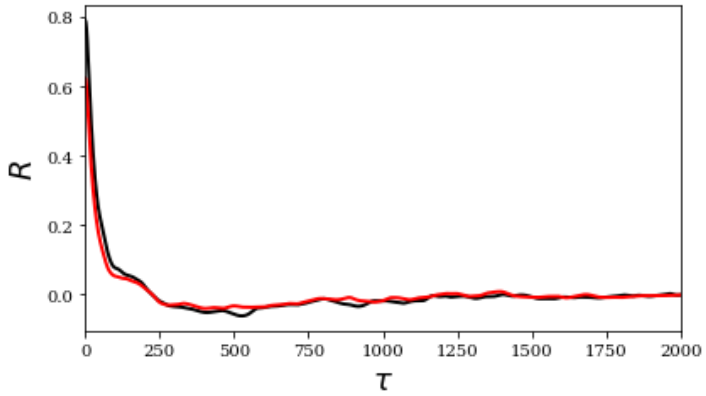
(c) Auto-correlation for the MFE system when $N_{CL} = 160$ and $L = 3$

Figure 6.38: Comparison of Auto-correlation function when N_{cl} varies for the MFE system

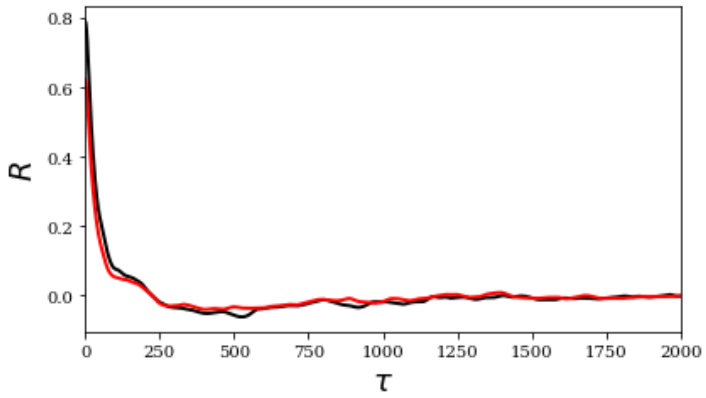
The auto-correlation function for fixed number of clusters and different order of model can be found in Figure 6.39. It can be seen that when increasing L from 3 to 20, the red curve becomes closed to the black curve, which indicating a better performance of the model. But when increasing L from 20 to 40, the result almost does not change, which indicating there is a limit.



(a) Auto-correlation for the MFE system when $N_{CL} = 80$ and $L = 3$



(b) Auto-correlation for the MFE system when $N_{CL} = 80$ and $L = 20$



(c) Auto-correlation for the MFE system when $N_{CL} = 160$ and $L = 40$

Figure 6.39: Comparison of Auto-correlation function when L varies for the MFE system

To well capture the dynamical properties of the MFE system, $N_{cl} = 80$ and $L = 20$ is selected. The time evolution of the nine variables of the original system and the modeled system are shown in Figure 6.40 and Figure 6.41. The prediction of each variables' ranges and trends

seems quite successful.

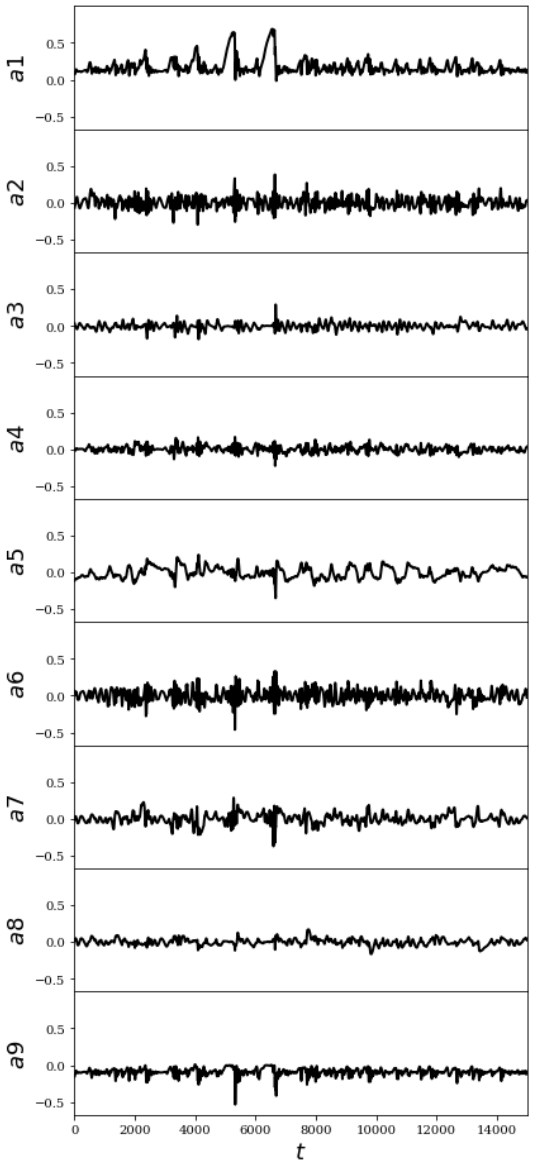


Figure 6.40: Time series for the MFE system without modeling

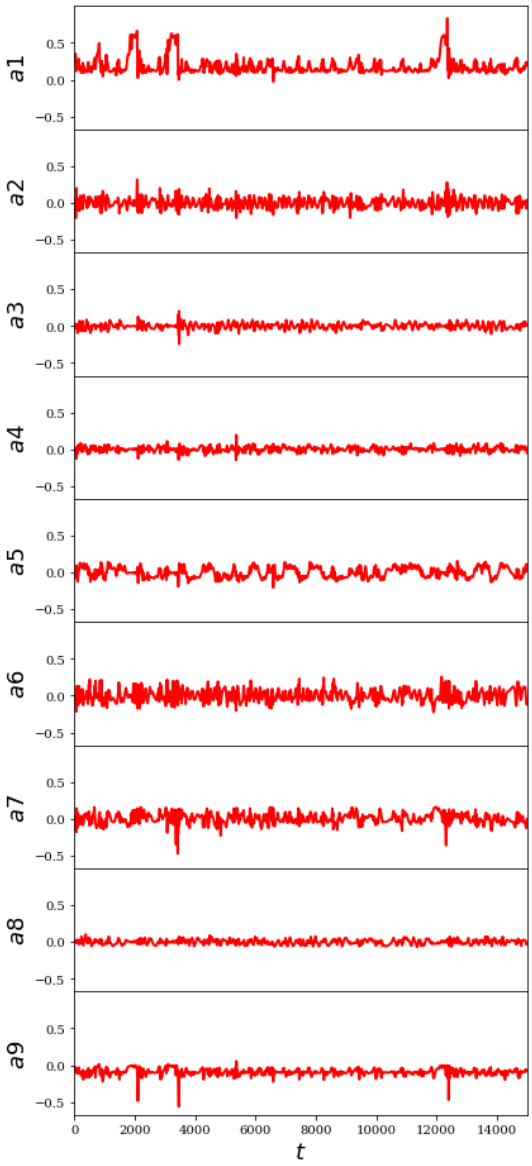


Figure 6.41: Time series for the MFE system when $N_{cl} = 80$ and $L = 20$

7

Control Results

In this chapter, the results of the control algorithm are examined for both the CDV and the MFE systems. Each section begins with an introduction to the control objectives and how the forcing term is introduced, followed by a discussion of the potential objective functions to optimize the control. Finally, the outcomes of the control implementation are detailed.

7.1. Truncated Charney-DeVore System

The CDV system is used as a benchmark case to test the control algorithm. To reduce the computational cost, $N_{cl} = 10$ is applied for the CDV system in this section. As discussed in Chapter 5, there are two regimes when the CDV system evolves. We decide the purpose of control to be maintaining the system in the zonal regime as much as possible, and get rid of slow evolution. To achieve this, a forcing term is added to the time derivative of x_1, and the magnitude is constrained within 5% of the range of $\dot{x}_1$ observed in the uncontrolled case. Consequently, the differential equations of the controlled CDV system becomes

$$\begin{aligned}
\dot{x_1} &= \gamma_1^* x_3 - C\left(x_1 - x_1^*\right) + b(b_k, \mathbf{x}) \\
\dot{x_2} &= -\left(\alpha_1 x_1 - \beta_1\right) x_3 - C x_2 - \delta_1 x_4 x_6 \\
\dot{x_3} &= \left(\alpha_1 x_1 - \beta_1\right) x_2 - \gamma_1 x_1 - C x_3 + \delta_1 x_4 x_5 \\
\dot{x_4} &= \gamma_2^* x_6 - C\left(x_4 - x_4^*\right) + \epsilon\left(x_2 x_6 - x_3 x_5\right) \\
\dot{x_5} &= -\left(\alpha_2 x_1 - \beta_2\right) x_6 - C x_5 - \delta_2 x_4 x_3 \\
\dot{x_6} &= \left(\alpha_2 x_1 - \beta_2\right) x_5 - \gamma_2 x_4 - C x_6 + \delta_2 x_4 x_2
\end{aligned} \tag{7.1}$$

The forcing term b is dependent on the location of the system $\mathbf{x}$ in the phase space, and a set of control parameters $\{b_k\}$ with each parameter corresponding to a specific cluster. These parameters are found by optimizing the objective function as discussed in Chapter 4.

Two kinds of objective function is used. The first aims to keep the mean value of $x_1^2 + x_4^2$ for all samples as close to the zonal states as possible. The objective function is defined as the distance between the mean value of $x_1^2 + x_4^2$ for all the samples and the centroids closest to the center of zonal regime. To identify the target centroid, the three centroids with the largest x_1 values are selected, and the one with the smallest x_4 value is designated as the objective centroid. Additionally, to achieve the control goal with minimal energy consumption, a term representing the magnitude of the forcing term must also be included. Both terms need to be normalized to balance their influence. The expression the the objective function is given by

$$\mathscr{J} = \mathscr{J}_{obj} + \mathscr{J}_e \tag{7.2}$$

Where $\mathscr{J}_{obj}$ is the term that is used to achieve the control objective, while $\mathscr{J}_e$ is related to minimizing the energy cost. And the expression of both is given by

$$\mathscr{J}_{obj} = \frac{1}{N_{obj}}[(\bar{x}_1 - x_{1,obj})^2 + (\bar{x}_4 - x_{4,obj})^2] \tag{7.3}$$

Where $\bar{x}_1$ and $\bar{x}_4$ are the mean value of x_1 and x_4 for current system, and $x_{1,obj}$) and $x_{4,obj}$) are the value of of x_1 and x_4 of the centroids closest to the center of zonal regime. N_{ob} represents the normalized factor for objective term.

$$\mathscr{J}_e = \frac{1}{NN_e}\sum_{i=0}^{N} b(b_k, \mathbf{x}(idt)) \tag{7.4}$$

Where N is the total number of samples in the system, and N_e represents the normalized factor for energy cost term.

With the well-defined objective function, $\{b_k\}$ can be obtained with the Nelder-Mead method. For each iteration in the optimizer, the system with a certain set of $\{b_k\}$ is simulated over a limited test time, and the corresponding objective function $\mathscr{J}$ is calculated for further optimization. Thus, this test time need to be choose carefully to ensure that the property of the system can be capture within such time. In this thesis, the test time when applying Nelder-Mead method is determined by Lyapunov time (LT) of the system, which can be calculated by the Lyapunov exponent [39] of the system. For control of the CDV system, $10LT$ is used to make sure most of the characteristics of the system are well captured.

The result when using the first objective function is shown in Figure 7.2. It can be seen that the samples have largely shifted, but the overall distribution still extends across a relatively wide area. The control managed to translate the mean value of $x_1^2 + x_4^2$ towards the lower right corner in the phase space but does not succeed in concentrating them within the zonal regime. This outcome suggests that while the control is effective in influencing the system's state, it is insufficient in driving the system into the desired zonal regime. The broader spread indicates that the control function is perhaps too simplistic, focusing on a global translation rather than a targeted compression into the zonal regime.

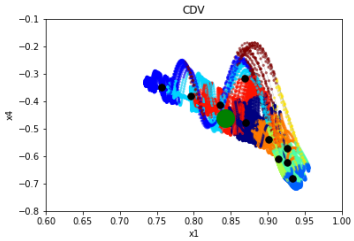


Figure 7.1: $x_1 - x_4$ projection for the original CDV system

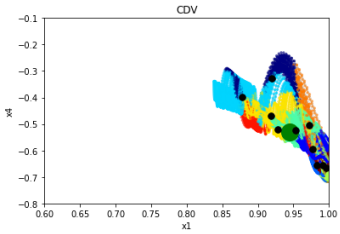


Figure 7.2: $x_1 - x_4$ projection for the controlled CDV system with control strategy 1

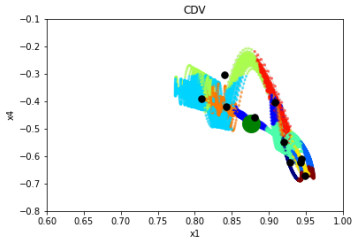


Figure 7.3: $x_1 - x_4$ projection for the controlled CDV system with control strategy 2

Second, the CDV system is controlled from a statistical perspective. Instead of solely focusing on the mean value of $x_1^2 + x_4^2$ for all the samples, the control strategy considers the distri-

bution of the samples on the $x_1 - x_4$ phase space. Thus, the objective location becomes a normal distribution with the mean equal to the objective centroids as discussed above. The objective function is then formulated as the Kullback–Leibler (K-L) divergence between the system's sample distribution and the target normal distribution near the zonal regime. The K-L divergence is given by

$$D_{KL}(P_1 \| P_2) = \sum_i P_1(i) \ln \frac{P_1(i)}{P_2(i)} \tag{7.5}$$

By definition, K-L divergence is naturally normalized. Thus, the expression of the objective term of objective function is given as

$$\mathscr{J}_{obj} = D_{KL}(P_{curr} \| P_{obj}) \tag{7.6}$$

where P_{curr} is the distribution of samples for the current system, and P_{obj} is the objective distribution, which is defined as a normal distribution with the mean value equals to the objective centroids and the variance equals to 0.01.

Similarly, a term representing the cost of control $\mathscr{J}_e$ is necessary as well, and is defined the same as Equation 7.4. With the well-formulated objective function, $\{b_k\}$ for each clusters can be efficiently optimized with Nelder-Mead method. The test time is the same as the first control strategy.

The result of the second objective function is shown in Figure 7.3. The result indicates a more effective control strategy, where the points have been compressed towards the lower right corner of the phase space. Although the sample distribution remains somewhat broad, it shows better performance compared to both the uncontrolled system and the system controlled with strategy 1. This demonstrates that the statistical control objective possess enormous potential in controlling the CDV system, as the points are more concentrated within the desired zonal regime. The successful compression suggests that this control function is better at guiding the system's states towards the target region, balancing the need to minimize energy cost while controlling the system effectively.

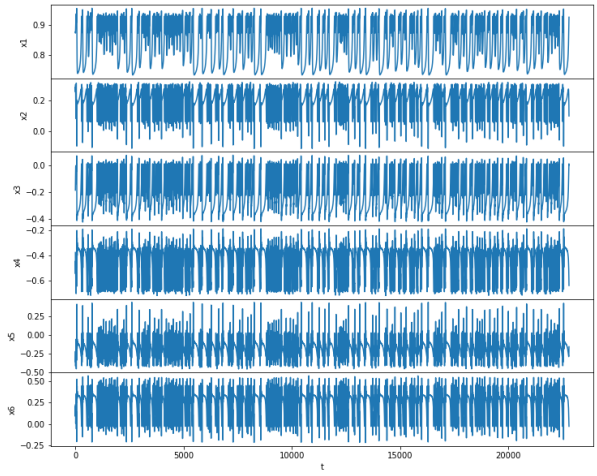


Figure 7.4: Time series for the original CDV system

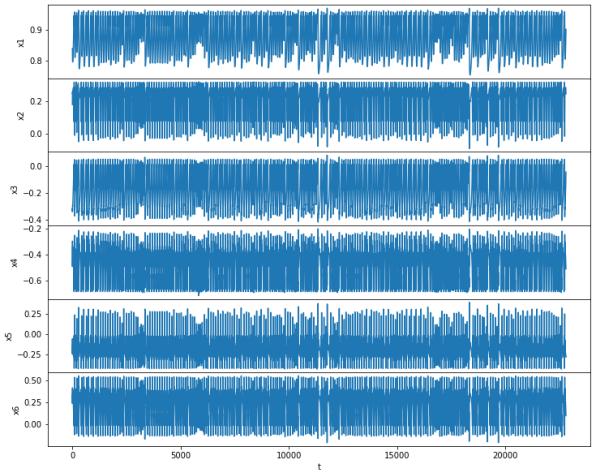


Figure 7.5: Time series for the controlled CDV system with control strategy 1

The comparison of the time evolution of the 6 variables of the original CDV system and the controlled CDV system with control strategy 2 is shown in Figure 7.4 and Figure 7.5 respectively. As has been discussed in Chapter 5, the features of blocked regime is a slow evolution of variables and a large decrease in x_1. From Figure 7.5, there is nearly no slow evolution observed. When the system enters a state with low x_1, it quickly recovers, exiting that state in a short time.

7.2. Moehlis-Faisst-Eckhart System

The MFE system is an ODE system with extreme events that occasionally occur. The control objective is to prevent extreme events from happening. As discussed in Chapter 6, to ensure the effectiveness of the control, $N_{cl} = 80$ is used for the MFE system in this section. Based on literature, one valid method to achieve this control objective is to increase the Reynolds number occasionally [40]. In the the controlled MFE system, Reynolds number is multiplied by a forcing term b, which is dependent on the location of the system $\mathbf{a}$ in the phase space, and a set of control parameters $\{b_k\}$ with each parameter corresponding to a specific cluster. The forcing term is restrained to 5, which means for $Re = 800$, the forced Reynolds number will not exceed 4000.

Two groups of objective functions are studied to examine the capability of the cluster-based control method.

The first is directly based on kinetic energy k, which is considered the most crucial indicator of extreme events. Two control strategies based on k are tested: one aims to minimize the mean value of k, while the other focuses on reducing the occurrence of extreme events. In this context, extreme events are defined as samples where k exceeds 0.1. Thus, the definition of the objective functions are

$$\mathscr{J}_1 = Mean(k) \tag{7.7}$$

and

$$\mathscr{J}_2 = \frac{\sum_{i=1}^{n} \mathbb{I}(k_i > 0.1)}{n} \tag{7.8}$$

With these well-formulated objective function, the clusters-specified control parameters $\{b_k\}$ can be obtained with the Nelder-Mead method. The test time of the Nelder-Mead method is determined by the Lyapunov time (LT) of the system as well. For control of the MFE system, $20LT$ is used to make sure that a sufficient number of extreme events occur in the uncontrolled system.

However, both control strategy appear to fail. A typical result can be found in Figure 7.6. This is a $k - \epsilon$ projection of the controlled and the uncontrolled MFE system using Equation 7.8 as objective function, and the yellow curve is representing the controlled system, while the blue curve is representing the uncontrolled system. It can be found that even though the control algorithm effectively reduced the samples with large k, it simultaneously causes a significant increase in the dissipation rate ϵ, which is unacceptable.

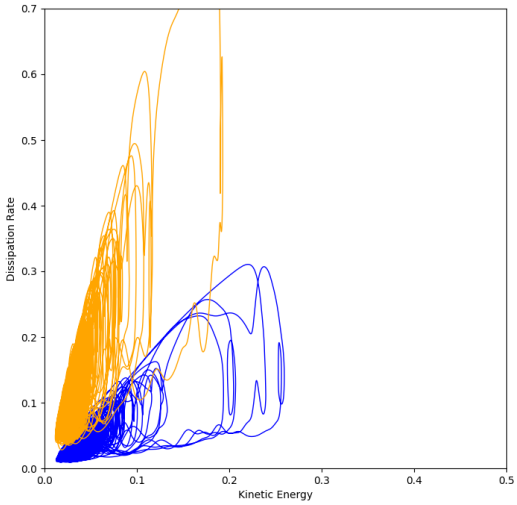


Figure 7.6: $k-\epsilon$ plot for kinetic energy control

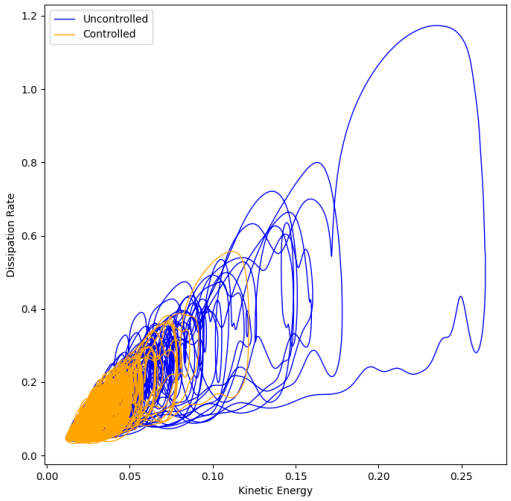


Figure 7.7: $k-\epsilon$ plot for dissipation rate control

The other groups of objective functions is based on mean dissipation rate. Since in a turbulence flow, the kinetic energy will finally dissipate, constraining the dissipation rate at a low level should, in theory, reduce the occurrence of extreme events. Based on this assumption, mean value of dissipation rate is used as the objective function, which is given by

$$\mathscr{J}_3 = Mean(\epsilon) \tag{7.9}$$

Then, using optimizing approach the clusters-specified control parameters $\{b_k\}$ can be obtained. The test time is also $20LT$.However, one drawback of this objective function is its computational cost. Calculating the mean dissipation rate requires the gradient of the velocity field. Thus, a rather time consuming matrix multiplication is required for all the samples within the test time for each iteration in the optimizer.

A result of the $k-\epsilon$ projection of the controlled and the uncontrolled MFE system can be found in Figure 7.7. It appears to be quite effective, significantly reducing extreme events. The samples from the system are now predominantly located in the low k and low ϵ region. Quantitative results are presented in Table 7.1. Compared to the original MFE system, the control algorithm reduce the mean kinetic energy by 20.7%. Additionally, it decreases the amount of extreme events by 97%.

	Mean kinetic energy k	**Percentage of extreme events occurrence**
Uncontrolled MFE system	0.0332	2.96%
Controlled MFE system	0.0263	0.29%

Table 7.1: Quantitative results for the MFE system control

The time evolution of k and ϵ for the MFE system before and after control are shown in Figure 7.8 and Figure 7.9. Both figures cover a time span that is 100 times longer than the test time used in the optimization. From these figures it is evident that both k and ϵ in the controlled system are maintained at a relatively low level compared to the uncontrolled system. Moreover, most of the large peaks in the figure, which indicates extreme events, are

eliminated. Conclusively, $\{b_k\}$ parameters, optimized over a short test time, demonstrates sufficient capability to control the system over a significantly extended time span.

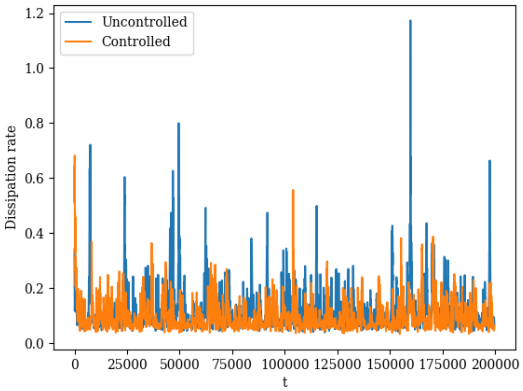


Figure 7.8: ϵ plot for dissipation rate control

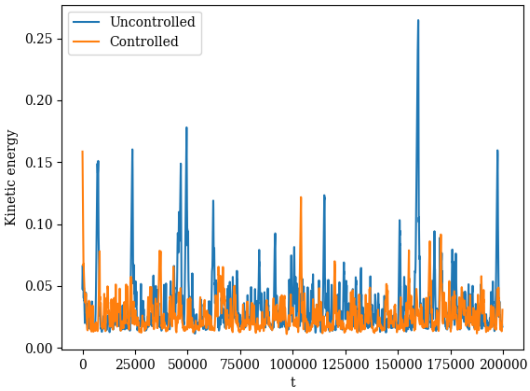


Figure 7.9: k plot for dissipation rate control

Figure 7.10 shows the probability density function (PDF) of the kinetic energy for the uncontrolled and controlled MFE system. The x-axis represents the kinetic energy(k), while the y-axis, on a logarithmic scale, shows the PDF values. An identical conclusion can be derived from these graphs. The controlled system effectively reduces the mean kinetic energy, as evidenced by the overall shift of the PDF curve to lower kinetic energy values. Especially, samples with large kinetic energy are eliminated.

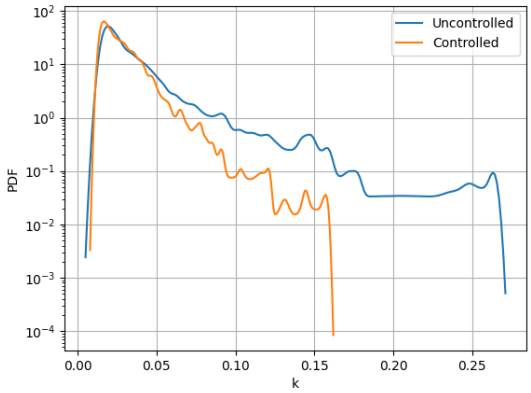


Figure 7.10: Probability density function of the uncontrolled and controlled MFE system

8

Conclusion and Recommendations

8.1. Conclusion

In this concluding chapter, reflections upon the key findings and outcomes from the cluster-based modeling and control method are presented. The primary goal of the research was to develop and validate efficient modeling and control strategies for chaotic dynamical systems, which are characterized by high-dimensional dynamics and chaotic behavior. The influence of key parameters on the cluster-based reduced-order modeling was well evaluated, from both static and dynamical perspective. And a quantitative approach to determine the optimal number of clusters N_{cl} was proposed, ensuring a balance between model accuracy and computational efficiency. Additionally, the cluster-based control algorithm was shown to be effective both in keeping the system in a certain state and prevent extreme event from happening. Different kinds of objective functions for each system were tested, and excellent control result was obtained, affirming the potential of cluster-based control algorithm for chaotic system control.

This thesis firstly explores the application of cluster-based Markov modeling (CMM) and cluster-based network modeling (CNM) techniques to chaotic systems, specifically focusing on the Lorenz, Truncated Charney-DeVore (CDV), and Moehlis-Faisst-Eckhart (MFE) systems. The results given in Chapter 6 indicates that good models were obtained by the CMM and CNM methods. Furthermore, the effect of number of clusters and model order were elucidated for the CNM. For all three systems, CMM method was proved to be effective in capturing the system's key features, as evidenced by the close alignment between the predicted and actual distributions in the phase space. The cluster transition matrix (CTM) and distance matrix confirmed the model's ability to represent the system's state transitions accurately, although the method showed limitations in dynamical prediction, particularly in forecasting specific states at specific times. For the Lorenz system and CDV system, CNM method provides more accurate time-series predictions, demonstrating its superior capability in capturing the chaotic dynamics of both systems. By fixing the order of model, the effect of the number of clusters on the performance of modeling was examined. It turns out that increasing the number of clusters did not necessarily enhance dynamical prediction accuracy, but can noticeably improve the CNM model's ability to make accurate statistical predictions. However, this improvement has a diminishing trend, highlighting the need for an optimal number of clusters. In this thesis, the mean variance of the samples in each cluster and the centroid of the cluster J_m was used to quantitatively determine the optimal number of clusters, preventing high computational cost. The optimal number of clusters for the Lorenz system, CDV system, and MFE system was determined to be 50, 50, and 80 respectively. While keeping the number of clusters the

same, the effect of the order of model L can be explored. The comparison of auto-correlation functions of different L suggests that increasing the order of the model enhances the CNM model's ability to capture dynamical properties of the system, providing a more accurate and reliable representation of both short-term and long-term dynamics. However, the comparison of distribution of state illustrated that statistical distribution of the reduced-order system seems rather similar, indicating that when predicting the statistical distributions, increasing the model order alone does not significantly improve accuracy. As a conclusion, increasing the order of model can be beneficial to capturing dynamical properties, but cannot increase the accuracy when predicting the state.

The results given in Chapter 7 validates the effectiveness of the cluster-based control algorithm which was tested on both the CDV system and the MFE system. The results demonstrated the potential of this approach in managing the complex dynamics of chaotic systems, particularly in maintaining desired states and preventing extreme events. The CDV system serves as a benchmark to evaluate the control algorithm's performance. The control strategy considers the distribution of the samples within the phase space and uses the Kullback-Leibler (K-L) divergence as the objective function. This method proves to be significantly effective, compressing the samples toward the target region and demonstrating a superior ability to maintain the system within the zonal regime. The comparison of time evolution between the original and controlled system further highlights the success of this approach, as the controlled system quickly recovered from states with low x_1, effectively avoiding slow evolution characteristic of the blocked regime.

The MFE system, characterized by the occasional occurrence of extreme events, presents a more complex challenge. The control purpose focuses on preventing these extreme events. The objective function, which focused on minimizing the mean dissipation rate, proves to be effective. This strategy not only reduces the occurrence of extreme events but also maintain both kinetic energy and dissipation rate at lower levels over an extended time span. Quantitative results show a substantial decrease in the mean kinetic energy and a significant reduction in the percentage of extreme events. The probability density function (PDF) further confirms these findings, with the controlled system exhibiting a clear shift towards lower kinetic energy values and the elimination of samples with high kinetic energy.

These findings highlight the potential of cluster-based control methods in practical applications, offering a powerful tool for managing the dynamics of chaotic systems. Especially, without the need for an analytic model, this control strategy can be of great significance for those systems whose governing equations is extremely hard to obtain. However, the results also indicate the importance of selecting appropriate control objectives.

8.2. Recommendations

For future research, several recommendations are given here.

1. In this thesis, only ordinary differential system are examined, and the system's degree of freedom doesn't exceed 9, allowing directly to apply clustering techniques to reduce the order of system. However, in reality, many of the chaotic dynamical systems are governed by partial differential equations with much more degrees of freedom. For these cases, a general approach to find the latent phase space to apply clustering is required. Proper orthogonal decomposition (POD) and autoencoders (AE) have already proved to be capable in finding a usable latent space for cluster-based modeling and control. Nevertheless, it is difficult to relate the latent space found by POD and AE with the original physical fields, leading to a dilemma to capture some evolution of coherent structures in

the original physical field. The approach that can generally find a latent phase space for cluster-based modeling and control for high-dimension system can be regarded as a good research direction, and future researchers can focus on the physical explanation of the latent space.

2. The influence of the model order L is well examined in this thesis. However, a way to determine the optimal model order is missing. Future researchers can study this further from two different perspectives. One is to consider the governing equations to check how far back in time can information influence the current state while the other can use machine learning techniques to determine the optimal L only with data.

References

[1] Osborne Reynolds. "IV. On the dynamical theory of incompressible viscous fluids and the determination of the criterion". In: *Philosophical transactions of the royal society of london.(a.)* 186 (1895), pp. 123–164.

[2] Ricardo Vinuesa and Steven L Brunton. "Emerging trends in machine learning for computational fluid dynamics". In: *Computing in Science & Engineering* 24.5 (2022), pp. 33–41.

[3] Steven H Strogatz. *Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering*. CRC press, 2018.

[4] FT Nieuwstadt, Jerry Westerweel, and B Boersma. *Introduction to theory and applications of turbulent flows*. Springer, 2016.

[5] Jerry Westerweel, Bendiks J Boersma, and Frans TM Nieuwstadt. *Turbulence: Introduction to Theory and Applications of Turbulent Flows*. Springer International Publishing, 2016.

[6] Henri Poincaré. *Les méthodes nouvelles de la mécanique céleste*. Vol. 2. Gauthier-Villars et fils, imprimeurs-libraires, 1893.

[7] Iain G Currie. *Fundamental mechanics of fluids*. CRC press, 2002.

[8] David D Nolte. "The tangled tale of phase space". In: *Physics today* 63.4 (2010), pp. 33–38.

[9] Edward N Lorenz. "Deterministic nonperiodic flow". In: *Journal of atmospheric sciences* 20.2 (1963), pp. 130–141.

[10] Philip Holmes. *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge university press, 2012.

[11] J Nathan Kutz et al. *Dynamic mode decomposition: data-driven modeling of complex systems*. SIAM, 2016.

[12] Thomas Peters. *Data-driven science and engineering: machine learning, dynamical systems, and control: by SL Brunton and JN Kutz, 2019, Cambridge, Cambridge University Press, 472 pp.,£ 49.99 (hardback), ISBN 9781108422093. Level: postgraduate. Scope: textbook*. Vol. 60. 4. Taylor & Francis, 2019.

[13] J Q Kou and W W Zhang. "Dynamic mode decomposition and its applications in fluid dynamics". In: *Acta Aerodynamica Sinica* 36.2 (2018), pp. 163–179.

[14] Jonathan H Tu. "Dynamic mode decomposition: Theory and applications". PhD thesis. Princeton University, 2013.

[15] Alberto Racca, Nguyen Anh Khoa Doan, and Luca Magri. "Predicting turbulent dynamics with the convolutional autoencoder echo state network". In: *Journal of Fluid Mechanics* 975 (2023), A2.

[16] Ehsan Farzamnik et al. "From snapshots to manifolds–a tale of shear flows". In: *Journal of Fluid Mechanics* 955 (2023), A34.

[17] Prem A Srinivasan et al. “Predictions of turbulent shear flows using deep neural networks”. In: *Physical Review Fluids* 4.5 (2019), p. 054603.

[18] Benjamin Peherstorfer and Karen Willcox. “Data-driven operator inference for nonintrusive projection-based model reduction”. In: *Computer Methods in Applied Mechanics and Engineering* 306 (2016), pp. 196–215.

[19] Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. “Discovering governing equations from data by sparse identification of nonlinear dynamical systems”. In: *Proceedings of the national academy of sciences* 113.15 (2016), pp. 3932–3937.

[20] Tong Wang, Huijun Gao, and Jianbin Qiu. “A combined adaptive neural network and nonlinear model predictive control for multirate networked industrial process control”. In: *IEEE Transactions on Neural Networks and Learning Systems* 27.2 (2015), pp. 416–425.

[21] FC Fu and JB Farison. “On the Volterra series functional evaluation of the response of non-linear discrete-time systems”. In: *International Journal of Control* 18.3 (1973), pp. 553–558.

[22] Chris Chatfield. *Time-series forecasting*. Chapman and Hall/CRC, 2000.

[23] Jer-Nan Juang. *Applied system identification*. Prentice-Hall, Inc., 1994.

[24] Eurika Kaiser et al. “Cluster-based reduced-order modelling of a mixing layer”. In: *Journal of Fluid Mechanics* 754 (2014), pp. 365–414.

[25] Hao Li et al. “Cluster-based network model”. In: *Journal of Fluid Mechanics* 906 (2021), A21.

[26] Chang Hou, Nan Deng, and Bernd R Noack. “Trajectory-optimized cluster-based network model for the sphere wake”. In: *Physics of Fluids* 34.8 (2022).

[27] Nan Deng et al. “Cluster-based hierarchical network model of the fluidic pinball–cartographing transient and post-transient, multi-frequency, multi-attractor behaviour”. In: *Journal of Fluid Mechanics* 934 (2022), A24.

[28] Aditya G Nair et al. “Cluster-based feedback control of turbulent post-stall separated flows”. In: *Journal of Fluid Mechanics* 875 (2019), pp. 345–375.

[29] Xin Wang et al. “Cluster-based control for net drag reduction of the fluidic pinball”. In: *Physics of Fluids* 35.2 (2023).

[30] Stuart Lloyd. “Least squares quantization in PCM”. In: *IEEE transactions on information theory* 28.2 (1982), pp. 129–137.

[31] Cyril Goutte et al. “On clustering fMRI time series”. In: *NeuroImage* 9.3 (1999), pp. 298–310.

[32] David MacKay. “An example inference task: Clustering”. In: *Information theory, inference and learning algorithms* 20 (2003), pp. 284–292.

[33] David Arthur, Sergei Vassilvitskii, et al. “k-means++: The advantages of careful seeding”. In: *Soda*. Vol. 7. 2007, pp. 1027–1035.

[34] Carl D Meyer. *Matrix analysis and applied linear algebra*. SIAM, 2023.

[35] Jule G Charney and John G DeVore. “Multiple flow equilibria in the atmosphere and blocking”. In: *Journal of the atmospheric sciences* 36.7 (1979), pp. 1205–1216.

[36] Jeff Moehlis, Holger Faisst, and Bruno Eckhardt. “A low-dimensional model for turbulent shear flows”. In: *New Journal of Physics* 6.1 (2004), p. 56.

[37] PJ Schmid et al. “Analysis and prediction of rare events in turbulent flows”. In: *Proceedings of the Summer Program. Center for Turbulence Research*. 2018.

[38] Madhura Joglekar, Ulrike Feudel, and James A Yorke. “Geometry of the edge of chaos in a low-dimensional turbulent shear flow model”. In: *Physical Review E* 91.5 (2015), p. 052903.

[39] Nguyen Anh Khoa Doan, Wolfgang Polifke, and Luca Magri. “Physics-informed echo state networks”. In: *Journal of Computational Science* 47 (2020), p. 101237.

[40] Alberto Racca and Luca Magri. “Data-driven prediction and control of extreme events in a chaotic flow”. In: *Physical Review Fluids* 7.10 (2022), p. 104402.

A

Nine modes of MFE system

The MFE system is based on Fourier modes and describes sinusoidal shear flow, in which the fluid between two free-slip walls experiences a sinusoidal body force. The original partial differential equation is

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u} \cdot \nabla)\mathbf{u} - \nabla p + \frac{1}{Re}\nabla^2 \mathbf{u} + \mathbf{F}(y) \tag{A.1}$$

with Reynolds number defined to be

$$Re = \frac{U_0 d}{2\nu} \tag{A.2}$$

and the non-dimensionalized body force is defined to be

$$\mathbf{F}(y) = \frac{\sqrt{2}\pi^2}{4Re} \sin(\pi y/2)\hat{\mathbf{e}}_x \tag{A.3}$$

The nine nodes of MFE system are Fourier modes that of the velocity in all three directions. The velocity field can be attained by

$$\mathbf{u} = \sum_{i=1}^{9} a_i u_i \tag{A.4}$$

Nine modes are presented below.

$$u_1 = \begin{pmatrix} \sqrt{2}\sin(\pi y/2) \\ 0 \\ 0 \end{pmatrix} \tag{A.5}$$

$$u_2 = \begin{pmatrix} \frac{4}{\sqrt{3}}\cos^2(\pi y/2)\cos(\gamma z) \\ 0 \\ 0 \end{pmatrix} \tag{A.6}$$

$$u_3 = \frac{2}{\sqrt{4\gamma^2 + \pi^2]}} \begin{pmatrix} 0 \\ 2\gamma \cos(\pi y/2) \cos(\gamma z) \\ \pi\gamma \sin(\pi y/2) \sin(\gamma z) \end{pmatrix} \tag{A.7}$$

$$u_4 = \begin{pmatrix} 0 \\ 0 \\ \frac{4}{\sqrt{3}} \cos^2(\pi y/2) \cos(\alpha x) \end{pmatrix} \tag{A.8}$$

$$u_5 = \begin{pmatrix} 0 \\ 0 \\ 2 \sin(\pi y/2) \sin(\alpha x) \end{pmatrix} \tag{A.9}$$

$$u_6 = \frac{4\sqrt{2}}{\sqrt{3(\alpha^2 + \gamma^2)}} \begin{pmatrix} -\gamma \cos(\alpha x) \cos^2(\pi y/2) \sin(\gamma z) \\ 0 \\ \alpha \sin(\alpha x) \cos^2(\pi y/2) \cos(\gamma z) \end{pmatrix} \tag{A.10}$$

$$u_7 = \frac{2\sqrt{2}}{\sqrt{\alpha^2 + \gamma^2}} \begin{pmatrix} \gamma \sin(\alpha x) \sin(\pi y/2) \sin(\gamma z) \\ 0 \\ \alpha \cos(\alpha x) \sin(\pi y/2) \cos(\gamma z) \end{pmatrix} \tag{A.11}$$

$$u_8 = N_8 \begin{pmatrix} \pi\alpha \sin(\alpha x) \sin(\pi y/2) \sin(\gamma z) \\ 2\left(\alpha^2 + \gamma^2\right) \cos(\alpha x) \cos(\pi y/2) \sin(\gamma z) \\ -\pi\gamma \cos(\alpha x) \sin(\pi y/2) \cos(\gamma z) \end{pmatrix} \tag{A.12}$$

where

$$N_8 = \frac{2\sqrt{2}}{\sqrt{(\alpha^2 + \gamma^2)(4\alpha^2 + 4\gamma^2 + \pi^2)}} \tag{A.13}$$

$$u_9 = \begin{pmatrix} \sqrt{2} \sin(3\pi y/2) \\ 0 \\ 0 \end{pmatrix} \tag{A.14}$$

B

Nelder-Mead Method

The Nelder-Mead method, or also known as simplex searching method, is an optimization algorithm used to find the minimum (or maximum) of an objective function in a multidimensional space. It's particularly useful for functions that are not differentiable or when gradient information is unavailable. The method is often applied in situations where the function is noisy, complex, or discontinuous.

Key Features:

- **Simplex Structure:** The Nelder-Mead method uses a geometric shape called a "simplex" which, in an n-dimensional space, consists of $n+1$ vertices for instance a triangle in 2D, or a tetrahedron in 3D. Each vertex represents a point in the search space.
- **Operations:** The method iteratively moves the simplex through the search space using operations like reflection, expansion, contraction, and shrinkage. These operations adjust the shape and position of the simplex to explore different areas of the search space.
- **No Derivatives Needed:** Unlike gradient-based methods, Nelder-Mead does not require the calculation of derivatives, making it suitable for functions that are non-smooth or noisy.

Steps of the Nelder-Mead Method:

1. **Initialization:** Start with an initial simplex formed by $n+1$ points in the n-dimensional space.
2. **Reflection:** Reflect the worst vertex across the centroid of the remaining vertices. If this new point is better, replace the worst point with it.
3. **Expansion:** If the reflection point is the best point so far, expand further in that direction.
4. **Contraction:** If the reflection point is not better, contract the simplex either towards the best point (outside contraction) or towards the centroid (inside contraction).
5. **Shrink:** If contraction fails to improve, shrink the simplex towards the best vertex.
6. **Termination:** The algorithm stops when the simplex converges, meaning that the vertices are close enough in space or the function values at the vertices are sufficiently similar.