

AMERICAN UNIVERSITY OF BEIRUT

CHARACTERIZING THE HYDRODYNAMICS
OF HIGH DENSITY CROWDS

by
ARWA AHMAD AWAD

A thesis
submitted in partial fulfillment of the requirements
for the degree of Master of Science
to the Department of Physics
of Faculty of Arts and Sciences
at the American University of Beirut

Beirut, Lebanon
January 2023

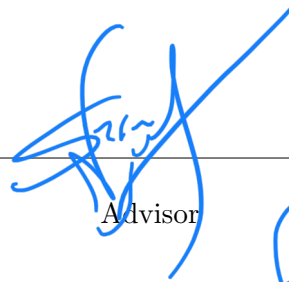
AMERICAN UNIVERSITY OF BEIRUT

CHARACTERIZING THE HYDRODYNAMICS
OF HIGH DENSITY CROWDS

by
ARWA AHMAD AWAD

Approved by:

Dr. Sara Najem, Assistant Professor
Physics



Advisor

Dr. Leonid Klushin, Professor
Physics



Member of Committee

Dr. Miguel Alfonso Mendez, Assistant Professor
von Karman Institute for Fluid Dynamics (VKI)



Member of Committee

Dr. Michaël Balabane, Professor
University of Paris- Institut Galilee - Math



Member of Committee

Date of thesis defense: January 18, 2022

AMERICAN UNIVERSITY OF BEIRUT

THESIS RELEASE FORM

Student Name: Awad Arwa Ahmad
Last First Middle

I authorize the American University of Beirut, to: (a) reproduce hard or electronic copies of my thesis; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes



As of the date of submission of my thesis

After 1 year from the date of submission of my thesis .

After 2 years from the date of submission of my thesis .

After 3 years from the date of submission of my thesis .

A handwritten signature in cursive script, appearing to read 'Arwa', is written over a horizontal line.

Signature

February 4, 2023

Date

ACKNOWLEDGEMENTS

With boundless gratitude, I extend my sincere appreciation to my advisor, Professor Sara Najem. Her unwavering guidance and support throughout my journey in completing this work have been nothing short of invaluable. With her direction and encouragement, I was able to embark on this journey with confidence, and ultimately reach its conclusion. Professor Najem's critical guidance was a beacon of hope throughout this endeavor, and I am forever grateful for her contribution.

I am also deeply thankful to my esteemed committee members, Professor Miguel Mendez, Leonid Klushin, and Michael Balabane, for their unwavering support and invaluable feedback throughout my journey. Their expertise and guidance have shaped my work.

Lastly, I would like to extend my eternal gratitude to my loving mother and father. From my childhood, they have been my constant source of support and encouragement, pushing me to take new steps and reach for the stars. In moments of doubt and uncertainty, they have been my rock, offering their unwavering love and support when no one else was there. May God bless them with good health and happiness always.

ABSTRACT

OF THE THESIS OF

Arwa Ahmad Awad for Master of Science
Major: Physics

Title: Characterizing The Hydrodynamics Of High Density Crowds

Collective behavior is a global phenomenon observed in a diverse of actively moving groups. It is the result of individual components interacting together or responding to external perturbations that propagate along the system's length scales. Humans have long been fascinated by the spectacular self organizing patterns generated by such collective behavior[1]. These patterns are displayed in a variety of examples that include flocks of birds , schools of fish [2], and human crowds. The latter display dynamically rich behavior in highly crowded events such as marathons [3], concerts [4] , and religious events [5]. In some cases, stop and go waves as well as crowd turbulence may form leading to serious disasters. Therefore, one of the scientific community's main goals is to characterize and predict such collective responses as a preventative measure for crowd hazards [3].

Two different approaches are used to characterize the collective behavior of dense human crowds. The first approach, which simulates crowd behavior by identifying local interactions among pedestrians, has been widely used by scientists. One of the most popular models proposed is the "Social Force Model" [6] introduced by Helbing and his collaborators. It has been successful in predicting the formation of irregular flows in crowd panic situations [7]. The second approach, known as the "Hydrodynamic Approach", bypasses the local interactions among agents and characterizes the entire motion as a flow [8]. Such approach has not been used abundantly in describing human collective behavior despite its success in explaining the emergent patterns of microscopic systems such as bacterial assemblies [9].

The derivation of a hydrodynamic model for a certain physical system is normally done through utilizing symmetry arguments and first principles [10]. The model is then validated through experimental observations. Such classical process has been successful in characterizing a wide range of systems in physics and engineering, but

not in complex systems' realm. To overcome these challenges, scientists have made use of the major advancements in video tracking technologies and sensors used in experiments to quantitatively characterize human crowd systems. The combination of such innovations with statistical learning techniques such as Non Linear Regression [5] and Neural Networks [11] has facilitated the inference of the system's governing partial differential equations (PDEs) solely from time series data collected at different spatial points. Recently, influential work such as PDE-Find [10] has proposed sparse regression techniques to infer the PDEs from spatio-temporal data. This approach determines the most prominent components in an equation via Penalized Linear Regression Methods and balances between accuracy and complexity. This technique has been used in [9] and has proven its ability to predict the emergent patterns formed in systems of interest.

In this thesis, we derive a hydrodynamic model for the collective behavior of runners in a marathon using the sparse regression approach proposed in [10]. The motion is made up of millions of runners heading toward the start line guided by staff members who are performing a regular cycle of walks and stops[3]. The motion is thus perfectly polarized with orientational fluctuations repressed. Furthermore, the runners' response to the staff's excitation causes the propagation of density and velocity waves spanning the entire system where the attenuation is small and the propagation is a longitudinal wave. Modeling the slow dynamics [8] of the crowd's flow is beneficial in providing recommendations for crowd management to avoid hazardous and momentous events in marathons.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	1
ABSTRACT	2
1 Introduction	7
1.1 Overview	7
1.2 Different Approaches for Modelling Crowd Events	7
2 Hydrodynamics of Active Matter	9
2.1 Introduction	9
2.2 Ordering Phenomena in Active Matter[8]	10
2.3 Toner and Tu Model	10
3 Sparse Regression	13
3.1 Linear Regression	13
3.2 Sparse Regression	15
3.2.1 LASSO Regression	15
3.2.2 Regularized Ridge Regression	15
3.2.3 Comparison Between LASSO and Ridge Regression	16
3.3 Inferring PDE Terms Using Sparse Regression (PDE-Find) [10]	16
3.3.1 Building The Candidate Library	17
3.3.2 Finding the sparse vector ξ using Sparse Regression Techniques	18
3.4 Inferring a system of ODE's using Sparse Regression [34]	19
4 Model Reduction and Data Denoising Techniques	22
4.1 Modal Reduction Techniques	22
4.1.1 Matrix Representation of High Dimensional Data	23
4.1.2 Proper Orthogonal Decomposition	24
4.1.3 Galerkin Projection	26
4.2 Signal Denoising Techniques	27
5 Results	31
5.1 Marathon System Description	32
5.2 Partial Differential Equation for Runners' Linear Waves	34
5.3 Model Reduction Approach	39
5.3.1 Proper Orthogonal Decomposition for the Motion of Runners	39

5.3.2	Inferring a system of ODEs for the temporal structures ψ_i . . .	41
5.3.3	Stability Analysis of the Obtained ODE	45
5.4	Comparison Between the Approaches	48
6	Conclusions and Future Work	50
	Bibliography	52

ILLUSTRATIONS

1.1	Concerts	8
1.2	Religious Events	8
1.3	Marathons	8
3.1	Difference Between Ridge and LASSO Regression, photo taken from [30]	17
4.1	Filter Representation	27
4.2	Different Filter Types	28
4.3	Filter Response To Frequencies	28
4.4	FIR Filter Block diagram [52]	29
4.5	Frequency and Impulse Response for a low pass FIR filter [53]	30
5.1	Summary of the Work done	31
5.2	Runners in Chicago Marathon	32
5.3	Velocity Fields at Two Instants	32
5.4	Longitudinal Velocity Kymograh	33
5.5	Kymograph for a single wave	33
5.6	Kymograph for the velocity wave simulated from PDE 5.14	38
5.7	ψ_2 before and after Densoing	40
5.8	Relative amplitude versus Mode number	41
5.9	Spatial Modes X	42
5.10	Temporal Modes ψ	43
5.11	Testing error as function of r and λ	45
5.12	Heat Map for the Inferred ODE	46
5.13	Comparison Between Prediction and Real Values for the temporal modes ψ_i	47
5.14	Mean Velocity	49

CHAPTER 1

INTRODUCTION

1.1 Overview

According to United Nations records, the human population reached 8 billion in November 2022 [12]. This increase has been accompanied by an increase in highly dense, crowd-gathering events where millions of people gather in relatively small spaces, such as concerts [4], marathons [3], and religious events [13]. These crowd events are characterized by self-organizing patterns similar to those observed in bird flocks [14] and fish schools [15].

Although the beautiful patterns formed during such events, serious disasters such as stampedes and tramlings can occur, resulting in a large number of deaths and injuries among pedestrians. More than 30 crowd panics have happened since 1945, causing serious injuries to around 3400 people and death to 1000 people in different locations around the world [16]. The 2006 Hajj Crush, which occurred on the Jamarat Bridge in Mecca, resulted in the death of 363 pilgrims. Therefore, one of the main goals of the scientific community is to characterize and predict such collective responses as a preventative measure against crowd hazards.

1.2 Different Approaches for Modelling Crowd Events

When studying the collective behavior of pedestrians in crowded events, various modeling techniques are used. These techniques are divided into two types. The first approach, known as the microscopic approach, studies the crowd's collective behavior by inferring element-wise interaction rules among neighboring pedestrians. An example of a microscopic model is the social force model developed by Helbing and his collaborators [6]. In this model, the motion of pedestrians is described by subjecting them to forces that depend on their personal environment and interactions with each other.

On the other hand, the second approach used is known as the macroscopic approach, which infers the macroscopic properties of the collective behavior as a whole, bypassing the local interactions among neighboring pedestrians. An example of a



Figure 1.1: Concerts



Figure 1.2: Religious Events



Figure 1.3: Marathons

macroscopic model is the hydrodynamic model, which models high-density crowds as fluid flow [3]. In this approach, the motion of the crowd is characterized by continuum fields such as the velocity field and density field, and the dynamics of these fields are inferred using conservation laws and symmetry arguments. For example, the dynamics of the density field can be obtained using the continuity equation.

CHAPTER 2

HYDRODYNAMICS OF ACTIVE MATTER

2.1 Introduction

Active matter systems are composed of components that use energy to move. These systems have many interesting non-equilibrium features, such as patterns that form due to the strong collective behavior of the components as they interact with each other or the environment [17]. Examples of active matter systems include bird flocks, fish schools, and human crowds. However, it is currently not possible to create a theoretical description that accurately captures the general characteristics of these systems while considering the many degrees of freedom present [8].

Both hydrodynamic and agent-based models have been used in the literature to study active matter systems. Agent-based models provide a minimal approach that focuses more on order and fluctuation than on forces and mechanics. Vicsek and his collaborators used this approach in [18] to study the phase transition of bird flocks from disordered to ordered states based on the noise strength and density. In this study, each active component is modeled as a point particle with a constant speed (v_0) and an orientation vector ($\hat{\mathbf{p}}$). The direction of ($\hat{\mathbf{p}}$) changes based on noisy local rules that force every point particle to align with its neighbors at every time instant [8].

Contrary to agent-based models, hydrodynamic models are continuum models that describe the macroscopic properties of large-scale behavior by inferring the dynamics of a few fields only [19]. There are two approaches to obtain a hydrodynamic model for an active matter system.

- infer the hydrodynamic equation of the continuum fields using symmetry and conservation laws (as done in [20] to derive the "General Theory of Flocking" and in [21] to study the long-wavelength behavior of active membranes)
- use non-equilibrium statistical physics tools to coarse-grain the microscopic model of the active system to obtain the system's large-scale equations (as

done in [22] and [23].

Systems at different scales can have similar macroscopic properties even though their microscopic properties are very different. This means that active matter systems can be divided into groups called universality classes based on their conservation laws and symmetry properties, which have similar macroscopic properties and hydrodynamic equations [8].

2.2 Ordering Phenomena in Active Matter [8]

Active matter systems can order themselves either in a polar phase or a nematic phase. Polar ordering [24], [25], which is described by a polarization vector (\mathbf{p}), occurs when a system is made up of elongated, self-propelled units that have a head-tail asymmetry and are aligned in the same direction. In contrast, nematic ordering [26], [27], which is described by an alignment tensor (\mathbf{Q}), occurs in systems where elongated, self-propelled units have random head-tail orientations and are parallel, or in systems where the self-propelled units are head-tail symmetric.

2.3 Toner and Tu Model

The Toner and Tu Model is a model used to understand the polar ordering behavior of bird flocks. It is a continuum model based on the Vicsek model [18], and was originally developed by Toner and Tu [28] using symmetry arguments. Recently, it was rederived by Ihle [8] through the process of coarse-graining the microscopic Vicsek model. In this section, we will briefly explore the hydrodynamic equations for density ρ and polarization \mathbf{p} in the Toner and Tu Model.

In systems where birds exchange energy with their environment, the number density of active matter, $\rho(\mathbf{x}, t)$, is the only conserved quantity. The orientational order of the system is described by the polarization vector, $\mathbf{p}(\mathbf{x}, t)$. This vector and the number density $\rho(\mathbf{x}, t)$ can be expressed in terms of the position, $\mathbf{x}_n(t)$, and unit velocity vector, $\hat{\mathbf{v}}_n(t)$, of each active particle. The unit vector $\hat{\mathbf{v}}_n(t)$ indicates the orientation of the velocity vector for each particle at a given time.

The expression of $\rho(\mathbf{x}, t)$ and $\mathbf{p}(\mathbf{x}, t)$ as function of $\mathbf{x}_n(t)$ and $\hat{\mathbf{v}}_n(t)$ is:

$$\rho(\mathbf{x}, t) = \sum_n \delta(\mathbf{x} - \mathbf{x}_n(t)) \quad (2.1)$$

$$\mathbf{p}(\mathbf{x}, t) = \frac{1}{\rho(\mathbf{x}, t)} \sum_n \hat{\mathbf{v}}_n(t) \delta(\mathbf{x} - \mathbf{x}_n(t)) \quad (2.2)$$

In order to distinguish between non-equilibrium terms and terms that arise from

a free energy function, \mathbf{F} , the hydrodynamic equations for the continuum fields $\mathbf{p}(\mathbf{x}, t)$ and $\rho(\mathbf{x}, t)$ in [8] are written as:

$$\frac{\partial \rho}{\partial t} + v_0 \nabla \cdot (\rho \mathbf{p}) = 0 \quad (2.3)$$

$$\frac{\partial \mathbf{p}}{\partial t} + \lambda (\mathbf{p} \cdot \nabla) \mathbf{p} = -\frac{1}{\gamma} \frac{\delta \mathbf{F}}{\delta \mathbf{p}} + \mathbf{f} \quad (2.4)$$

Equation 2.3 is the mass conservation equation, where $v_0 \mathbf{p}$ represents the velocity field \mathbf{v} . The second term on the left-hand side of equation 2.4 is controlled by the parameter λ , which has the dimension of speed. Unlike conventional fluids, which are Galilean invariant due to their momentum conservation, the parameter λ is not universal and depends on the microscopic properties of the system, with $\lambda \neq v_0$ (v_0 being the average velocity of the active system). The last term \mathbf{f} on the right-hand side of equation 2.4 represents the fluctuations of the system and is independent of ρ and \mathbf{p} . These fluctuations are assumed to be white Gaussian noise with zero mean and correlations

$$\langle f_\gamma(\mathbf{x}, t) f_\beta(\mathbf{x}', t') \rangle = A \delta_{\gamma\beta} \delta(\mathbf{x} - \mathbf{x}') \delta(t - t') \quad (2.5)$$

The expression of the free energy function \mathbf{F} used in equation 2.4 is given by [8]:

$$\mathbf{F} = \int_{\mathbf{x}} \frac{\hat{\alpha}(\rho)}{2} |\mathbf{p}|^2 + \frac{\hat{\beta}}{2} |\mathbf{p}|^4 + \frac{\hat{K}}{2} (\partial_a \mathbf{p}_b) (\partial_a \mathbf{p}_b) + \frac{w}{2} |\mathbf{p}|^2 \nabla \cdot \mathbf{p} - b_1 \nabla \cdot \mathbf{p} \left(\frac{\delta \rho}{\rho_0} \right) + \frac{A}{2} \left(\frac{\delta \rho}{\rho_0} \right)^2 \quad (2.6)$$

$\delta \rho$ is the density fluctuation around its average value ρ_0 .

The parameter $\hat{\alpha}$ depends on both local density ρ and noise strength [29] while $\hat{\beta}$ is positive quantity to ensure stability. When $\hat{\alpha}$ parameter tends to zero, the mean field order disorder transitions controlled by the first two terms on the right hand side of equation 2.6 take place. On the other hand, the spatially inhomogeneous deformation of the order parameter's energy cost is described by the third term in the free energy, which is controlled by the positive parameter \hat{K} (Frank constant). The fifth and fourth terms in 2.6 allow density ρ and polarization magnitude $|\mathbf{p}|^2$ gradients to appear in equation 2.4 through integration by parts. These gradients contribute in an aligning field for the polarization vector \mathbf{p} .

Through combining equation 2.6 and equation 2.4, the hydrodynamic equation for the polarization vector \mathbf{p} becomes:

$$\frac{\partial \mathbf{p}}{\partial t} + \lambda(\mathbf{p} \cdot \nabla)\mathbf{p} = -\frac{1}{\gamma}(\hat{\alpha} + \hat{\beta}|\mathbf{p}|^2)\mathbf{p} + \frac{\hat{K}}{\gamma}\nabla^2\mathbf{p} - \frac{w}{\gamma}\nabla\left(\frac{\rho}{\rho_0}\right) + \frac{b_1}{2\gamma}\nabla|\mathbf{p}|^2 - \lambda_1\mathbf{p}(\nabla \cdot \mathbf{p}) + \mathbf{f} \quad (2.7)$$

Equation 2.7 can be compared to the Navier-Stokes equations because the polarization vector \mathbf{p} is proportional to the velocity vector. The first two terms on the right-hand side of equation 2.7 depict frictional forces, while the third and fourth terms represent the gradient of a pressure term. This pressure term is a function of the density and the magnitude of the polarization vector, and its gradient contributes to the aligning field for the polarization vector.

CHAPTER 3

SPARSE REGRESSION

3.1 Linear Regression

One of the most widely used machine learning models is regression. Regression has several applications, including forecasting, finding the cause-and-effect relationship between variables, and time series modeling [30]. The goal of regression is to learn a function that predicts the value of a dependent variable based on several observed variables, known as independent variables or features. Regression functions are learned from available data and can be used to predict the output of new cases.

Linear regression is a statistical method used to understand the relationship between two continuous variables: an independent variable, x , and a dependent variable, y . It aims to find a linear relationship between these variables, such that a change in x can be used to predict a change in y . Linear regression was first developed in the pre-computer era, but it remains useful today because of its simplicity and interpretability. The relationship between the variables is described by a linear equation, which can be used to predict the value of y based on a given value of x . The ability to interpret the effect of input features on the output is a key advantage of linear regression, making it useful for understanding the relationships between different variables in a dataset [30].

The linear model is characterized by a vector of coefficients β of length equal to the number of predictors p , along with a y-intercept β_0 . Given the input vector $X = (X_1 \ X_2 \ \cdots \ X_p) \in \mathbf{R}^p$ and the output variable $Y \in \mathbf{R}$, the model is of the form:

$$Y = \beta_0 + \sum_{i=1}^p X_i \beta_i \tag{3.1}$$

The above equation can be simplified by adding 1 to the input vector X to become $X = (1 \ X_1 \ X_2 \ \cdots \ X_p)$, along with adding β_0 to the vector β . Thus, equation

3.1 becomes:

$$Y = (1 \quad X_1 \quad X_2 \quad \cdots \quad X_p) \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \cdots \\ \beta_p \end{pmatrix} = X\beta \quad (3.2)$$

Evaluating 3.2 on n data points consisting of a dependent variable Y_i and an independent variable X_i yields:

$$\begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} 1 & X_{11} & X_{12} & X_{13} & \cdots & X_{1p} \\ 1 & X_{21} & X_{22} & X_{23} & \cdots & X_{2p} \\ 1 & X_{31} & X_{32} & X_{33} & \cdots & X_{3p} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & X_{n1} & X_{n2} & X_{n3} & \cdots & X_{np} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \beta_3 \\ \vdots \\ \beta_p \end{bmatrix} \quad (3.3)$$

$$\mathbf{Y} = \mathbf{X}\beta \quad (3.4)$$

The linear regression coefficients can be estimated using the Least Square Method, which minimizes the residual sum of squares (RSS) using available data points. The RSS is given by:

$$RSS(\beta) = \sum_{i=1}^n (Y_i - \beta_0 + \sum_{j=1}^p X_{ij}\beta_j)^2 = \|\mathbf{Y} - \mathbf{X}\beta\|_2^2 \quad (3.5)$$

After estimating the coefficients of the linear regression model (β), it is important to check whether the model is prone to overfitting. Overfitting occurs when the model is able to perfectly predict Y for the data used in the fitting process but fails to accurately predict Y for new data points.

To check if a linear regression model is suffering from overfitting, the following steps can be taken:

1. Split the available data into a training set, which will be used for fitting the model, and a testing set, which will be used to evaluate the final error at the end of the training.
2. Calculate the training error using the following formula:

$$\text{Error}_{train} = \|\mathbf{X}_{train}\beta - \mathbf{Y}_{train}\|_2 \quad (3.6)$$

3. Calculate the testing error using the following formula:

$$\text{Error}_{test} = \|\mathbf{X}_{test}\beta - \mathbf{Y}_{test}\|_2 \quad (3.7)$$

4. A model that is suffering from overfitting will have a low training error but a large testing error.

3.2 Sparse Regression

The least square method is commonly used to find the coefficients of a linear regression model, but it can produce unsatisfactory results when there are a large number of predictors (independent variables). In these cases, it is often preferable to find an interpretable model with high prediction accuracy and a small number of predictors that have the greatest impact on the dependent variable [30].

Obtaining a parsimonious model that balances between accuracy and interpretability can be achieved using sparse regression techniques, which aim to identify the most important predictors while minimizing the number of non-zero coefficients. Some examples of these techniques include LASSO regression [31] and ridge regression [32]. These methods are particularly useful when dealing with large numbers of predictors, as they can help identify the most impactful variables while maintaining good prediction accuracy.

3.2.1 *LASSO Regression*

The least absolute shrinkage and selection operator (LASSO) is a statistical modeling method that was introduced by Tibshirani in 1996 [31]. It is known for its ability to balance complexity and interpretability in statistical models. LASSO is a l_1 -penalized regularization method that produces a sparse coefficient vector β with non-zero entries only for the most relevant predictors. This is achieved by adding an l_1 penalty to the least squares problem, which helps to prevent overfitting.

The Lagrangian formulation of LASSO formulation can be written as:

$$\beta = \operatorname{argmin}_{\beta} \|X\beta - Y\|_2 + \lambda \|\beta\|_1 \quad (3.8)$$

The parameter λ controls the number of non-zero entries in the coefficient vector β where larger values of λ cause more coefficients to be set to zero [30]. In order to find the optimal value of λ , the procedure generally involves building a model for each value of λ in a list, testing the performance of each model on the testing data, and selecting the value of λ that results in the minimum testing error.

3.2.2 *Regularized Ridge Regression*

Ridge Regression is a regularization technique that shrinks the coefficients of the linear regression model to prevent overfitting. It does this by adding an l_2 penalty term to the least squares objective function. This penalty term shrinks the coefficients towards zero, but does not set them to zero like LASSO Regression does.

The amount of shrinkage is controlled by a parameter λ [30], which is chosen from a list of possible values. The optimal value of λ is generally determined by building a model for each value of λ , testing the performance of each model on the testing data,

and then selecting the value of λ that achieves the minimum testing error. Ridge Regression was developed in 1970 by Arthur E. Horel and Robert W. Kennard [32].

The Lagrangian formulation of Ridge Regression is given by:

$$\beta = \operatorname{argmin}_{\beta} \|X\beta - Y\|_2 + \lambda \|\beta\|_2 \quad (3.9)$$

Ridge Regression is often used when dealing with highly correlated variables in a linear regression model. In such cases, the coefficients of the model can be characterized by high variance. This means that a large positive coefficient on one variable can be cancelled out by a large negative coefficient on a correlated variable. To address this issue, Ridge Regression shrinks the coefficients towards zero, which helps reduce variance and prevent overfitting.

3.2.3 Comparison Between LASSO and Ridge Regression

The difference between LASSO and Ridge Regression can be illustrated with Figure 3.1, which shows the LASSO framework on the left and the Ridge Regression framework on the right in the case of two features. The elliptical contours depicted in the figure represent the residual sum of squares, centered at the full least square estimate. The l_2 constraint imposed by Ridge Regression is represented by the disk region, while the l_1 constraint imposed by LASSO is represented by the diamond region. Both models aim to find the first intersection point between the constraint region and the elliptical contours. In the LASSO framework, if the intersection occurs at one of the corners of the diamond, one of the parameters will be set to zero, while in Ridge Regression, both parameters are non-zero as the disk has no edges. While Ridge Regression reduces the magnitude of the coefficients, this doesn't necessarily mean that some coefficients will be set to zero. On the other hand, LASSO performs both feature selection and parameter shrinkage simultaneously, only keeping the most important terms in the model.

3.3 Inferring PDE Terms Using Sparse Regression (PDE-Find) [10]

To infer the most relevant terms in a partial differential equation (PDE), one can utilize sparse regression techniques on a large library of candidate terms that include linear, nonlinear, and spatial derivatives. This process is carried out using spatiotemporal data, which can be collected through either a Lagrangian approach using moving sensors or an Eulerian approach using fixed spatial sensors.

In order to infer the PDE, the following steps are followed:

1. Construct the candidate library Θ .
2. Inferring the most informative terms in the candidate library using sparse regression algorithms.

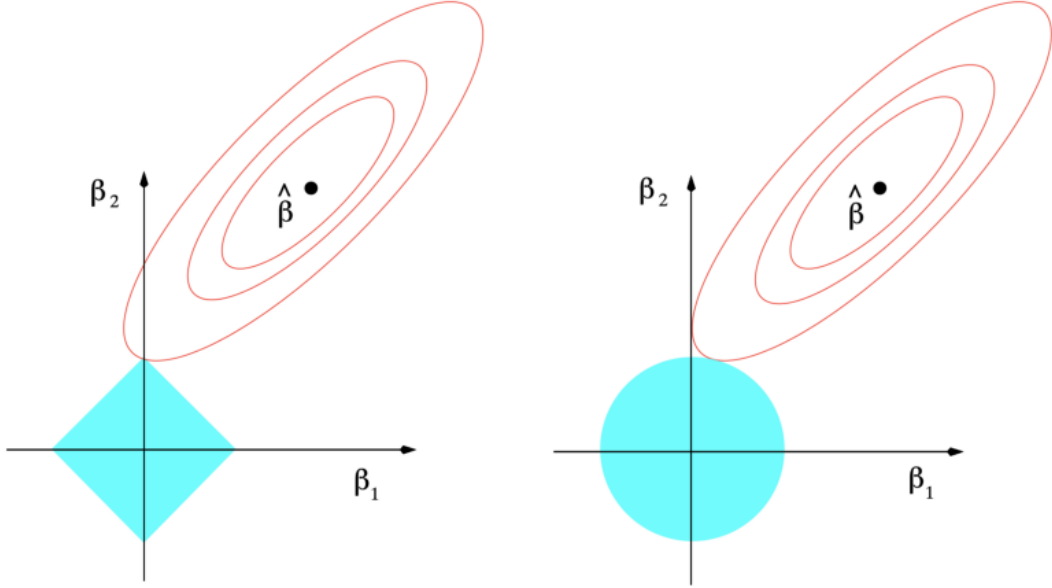


Figure 3.1: Difference Between Ridge and LASSO Regression, photo taken from [30]

3.3.1 Building The Candidate Library

The general form of a partial differential equation describing the evolution of function $v(x, t)$ is defined as:

$$v_t = F(v, v_x, v_{xx}, v^2, \dots, x) \quad (3.10)$$

The partial differentiation with respect to space and time is denoted by the subscripts while the right-hand side $F(\cdot)$ is a non linear function of v and its derivative with respect to both space and time. The function F is assumed to be a sparse function made up of few terms only .

The time series data measured at fixed spatial points is collected in a matrix V . The matrix $V \in \mathbb{C}^{nm}$ represents the data measured at m spatial points and n time points.

$$V = \begin{pmatrix} v_{11} & v_{12} & v_{13} & \cdots & v_{1m} \\ v_{21} & v_{22} & v_{23} & \cdots & v_{2m} \\ v_{31} & v_{32} & v_{33} & \cdots & v_{3m} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ v_{n1} & v_{n2} & v_{n3} & \cdots & v_{nm} \end{pmatrix} \quad (3.11)$$

Each row represents the value of v for a fixed time overall spatial points m while each column represents the value of v on a fixed spatial point over all time points.

After collecting all the data points in matrix V , matrix $\Theta \in \mathbb{C}^{nm \times D}$ is created which

contains D candidate terms that may be suitable for the function F evaluated at all spatial and temporal points. These candidate terms can include linear, non-linear, and partial derivative terms, and can be computed using numerical methods such as the finite difference method and polynomial interpolation. A possible way to describe the matrix Θ is:

$$\Theta = [1 \quad V \quad V^2 \quad V^3 \quad \dots \quad V_x \quad VV_x \quad \dots \quad V_{xx} \quad \dots] \quad (3.12)$$

Each column in Θ represents the values of a certain candidate term evaluated at all spatial and temporal points.

$$\Theta = \begin{pmatrix} 1 & v(x_0, t_0) & v^2(x_0, t_0) & \dots & v_x(x_0, t_0) & \dots & v_{xx}(x_0, t_0) & \dots \\ 1 & v(x_1, t_0) & v^2(x_1, t_0) & \dots & v_x(x_1, t_0) & \dots & v_{xx}(x_1, t_0) & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & v(x_m, t_0) & v^2(x_m, t_0) & \dots & v_x(x_m, t_0) & \dots & v_{xx}(x_m, t_0) & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & v(x_m, t_n) & v^2(x_m, t_n) & \dots & v_x(x_m, t_n) & \dots & v_{xx}(x_m, t_n) & \dots \end{pmatrix} \quad (3.13)$$

To obtain the time derivative of the data measured at fixed spatial points, numerical methods such as finite difference or polynomial interpolation can be used. The time derivative can then be evaluated at all spatial and temporal points and collected in a column vector V_t .

$$V_t = \begin{pmatrix} v_t(x_0, t_0) \\ v_t(x_1, t_0) \\ \vdots \\ v_t(x_m, t_0) \\ \vdots \\ v_t(x_m, t_n) \end{pmatrix} \quad (3.14)$$

After computing V_t , the time derivative of the data, and the candidate library matrix Θ , the discretized version of Equation 3.10 can be written as:

$$V_t = \Theta \xi \quad (3.15)$$

Where ξ is a sparse vector of size D (number of candidate terms). Each entry in ξ corresponds to a candidate term, and only the relevant terms will have non-zero entries in ξ . In other words, ξ represents the coefficients of the relevant terms that are used to approximate the time derivative of the data.

3.3.2 Finding the sparse vector ξ using Sparse Regression Techniques

Least square method can be used to calculate ξ , however most of ξ 's entries will be non zeros which leads to a partial differential equation that includes all the

Algorithm 1 STLSQ Algorithm(Θ, V_t, λ, N)

```
# Apply Least Square Method
 $\xi = \operatorname{argmin}_{\xi} \|V_t - \Theta\xi\|_2^2$ 
# Select  $\xi$  coefficient smaller than  $\lambda$ 
trimedcoeffs = { j :  $|\xi_j| \leq \lambda$  }
Apply Hard Thresholding
 $\xi[\text{trimedcoeffs}] = 0$ 
# Redo the above steps with fewer coefficients
 $\xi[\text{trimedcoeffs}] = \text{STLSQ}(\Theta, V_t, \lambda, N-1)$ 
return  $\xi$ 
```

terms in the candidate library. Therefore, LASSO regularization can be used instead of a combinatorial approach for obtaining a parsimonious vector ξ . However, the columns of the candidate library Θ are highly correlated, and LASSO performs poorly in such cases. Other methods have been proposed for a sparse representation of ξ such as Elastic Net Algorithm [33] that combines both LASSO and Ridge Regression Techniques.

Recently, the Sequential Threshold and Least Square Algorithm was proposed by Brunton et al in [34] and has been used in many papers to infer partial differential equations for physical systems. For example, this algorithm was used in [35] to learn the hydrodynamic equation for active matter from microscopic data obtained from particle simulations and experiments.

The Sequential Threshold and Least Square Algorithm (STLSQ) is a method for estimating a sparse vector ξ . It does this by first finding the least square estimate of ξ (i.e $\xi = \operatorname{argmin}_{\xi} \|Vt - \Theta\xi\|_2^2$) and then iteratively setting the coefficients of ξ below a specified thresholding parameter λ to zero. STLSQ takes as input a candidate library Θ , a time derivative matrix Vt , a thresholding hyper parameter λ , and the maximum number of iterations N , and returns a sparse vector ξ as output. The steps of the STLSQ algorithm are shown in Algorithm 1.

To obtain the sparsest partial differential equation (PDE) that balances complexity and interpretability, the STLSQ algorithm should be applied over a range of values for the thresholding parameter λ ($[\lambda_{\min}, \lambda_{\max}]$). This will result in a set of PDEs with increasing complexities. Each of these PDEs should then be validated based on its ability to accurately describe the phenomenology of the input data.

3.4 Inferring a system of ODE's using Sparse Regression [34]

The STLSQ algorithm can also be used to infer a system of sparse ordinary differential equations (ODEs) from time series data. The steps for doing so are similar

to those for inferring PDEs, with minor modifications.

The form of the differential equation describing the evolution of a dynamical system is:

$$\frac{d}{dt}\mathbf{x}(t) = \mathbf{f}(\mathbf{x}(t)) \quad (3.16)$$

Where $\mathbf{x}(t) \in \mathbf{R}^n$ represents the state of the system at time t , and $\mathbf{f}(\mathbf{x}(t)) \in \mathbf{R}^n$ is a sparse, nonlinear function of $\mathbf{x}(t)$ that imposes constraints on the evolution of the system.

In order to infer f , we can first sample $\mathbf{x}(t)$ at several time instants t_1, t_2, \dots, t_m , and then collect the values of $\mathbf{x}(t)$ at these instants in a matrix \mathbf{X} . The form of \mathbf{X} is:

$$\mathbf{X} = \begin{pmatrix} x_1(t_1) & x_2(t_1) & x_3(t_1) & \cdots & x_n(t_1) \\ x_1(t_2) & x_2(t_2) & x_3(t_2) & \cdots & x_n(t_2) \\ x_1(t_3) & x_2(t_3) & x_3(t_3) & \cdots & x_n(t_3) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_1(t_m) & x_2(t_m) & x_3(t_m) & \cdots & x_n(t_m) \end{pmatrix} \quad (3.17)$$

Each row in \mathbf{X} represents the values of the vector $\mathbf{x}(t)$ at a specific time instant, and each column in \mathbf{X} represents the values of a particular component of $\mathbf{x}(t)$ at all time instants from t_1 to t_m .

Similarly, the derivative of the state vector $\mathbf{x}(t)$, $\dot{\mathbf{x}}(t)$, must be sampled at t_1, t_2, \dots, t_m . $\dot{\mathbf{x}}(t)$ can be measured experimentally or approximated using numerical differentiation techniques, and the resulting values can be collected in a matrix $\dot{\mathbf{X}}$.

$$\dot{\mathbf{X}} = \begin{pmatrix} \dot{x}_1(t_1) & \dot{x}_2(t_1) & \dot{x}_3(t_1) & \cdots & \dot{x}_n(t_1) \\ \dot{x}_1(t_2) & \dot{x}_2(t_2) & \dot{x}_3(t_2) & \cdots & \dot{x}_n(t_2) \\ \dot{x}_1(t_3) & \dot{x}_2(t_3) & \dot{x}_3(t_3) & \cdots & \dot{x}_n(t_3) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \dot{x}_1(t_m) & \dot{x}_2(t_m) & \dot{x}_3(t_m) & \cdots & \dot{x}_n(t_m) \end{pmatrix} \quad (3.18)$$

Next, similar to the procedure for inferring PDEs, a candidate library matrix Θ is constructed using both \mathbf{X} . This matrix contains all the linear and nonlinear terms that $\mathbf{f}(\mathbf{x}(t))$ may be a function of. There is flexibility in choosing the candidate terms, as only a few of them will be active in each row after applying sparse regression.

Here is an example of a candidate library matrix Θ that includes both polynomial and trigonometric functions:

$$\theta(\mathbf{X}) = [1 \quad \mathbf{X} \quad \mathbf{X}^{P_2} \quad \mathbf{X}^{P_3} \quad \dots \quad \cos(\mathbf{X}) \quad \sin(\mathbf{X}) \quad \dots]$$

In the equation 3.19, X^{P_2} and X^{P_3} denote higher-order polynomials. For example, X^{P_2} , which represents quadratic linearities, can be expressed as:

$$X^{P_2} = \begin{bmatrix} x_1^2(t_1) & x_1(t_1)x_2(t_1) & x_1(t_1)x_2(t_1) & x_2^2(t_1) & \dots & x_3^2(t_1) & \dots & x_n^2(t_1) \\ x_1^2(t_2) & x_1(t_2)x_2(t_2) & x_1(t_2)x_2(t_2) & x_2^2(t_2) & \dots & x_3^2(t_2) & \dots & x_n^2(t_2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_1^2(t_n) & x_1(t_n)x_2(t_n) & x_1(t_n)x_2(t_n) & x_2^2(t_n) & \dots & x_3^2(t_n) & \dots & x_n^2(t_n) \end{bmatrix} \quad (3.19)$$

Once we have collected both \mathbf{X} and $\dot{\mathbf{X}}$ and constructed the candidate library matrix $\Theta(\mathbf{X})$, we can write Equation 3.16 as:

$$\dot{\mathbf{X}} = \Theta \Xi \quad (3.20)$$

where Ξ is a matrix made up of sparse columns ξ_i that determine the active nonlinear terms in the right-hand side of each row of Equation 3.16 ($\dot{x}k = fk(x)$). These columns can be estimated using the STLSQ algorithm.

Once Ξ has been determined, each row of Equation 3.16 can be expressed as:

$$\dot{x}k = \Theta(x^T)\xi k \quad (3.21)$$

where $\Theta(x^T)$ is a symbolic function of x and not a data matrix.

To determine the optimal thresholding hyperparameter λ for inferring ODEs, the time series data can be split into training data (80 percent of the available data) and testing data (20 percent of the total available data). STLSQ can then be used to estimate Ξ using the training data for a range of values of λ in the interval $[\lambda_{\min}, \lambda_{\max}]$. The optimal value of λ will be the one that leads to the highest accuracy on the testing data.

CHAPTER 4

MODEL REDUCTION AND DATA DENOISING TECHNIQUES

4.1 Modal Reduction Techniques

Dimensionality reduction is a technique used to transform high-dimensional data from physical experiments or numerical simulations into a lower-dimensional representation that captures the essential patterns of a complex system. This is done to address the challenge of the curse of dimensionality, which refers to the difficulty of analyzing systems with a high number of dimensions. The intrinsic dimensionality of a system [36] is the minimum number of dimensions needed to describe its observed patterns [37], and dimensionality reduction techniques such as factor analysis [38], classical scaling [39], and proper orthogonal decomposition (POD) [40] are used to identify and extract this intrinsic dimensionality.

Proper Orthogonal Decomposition, also known as POD, is a common technique used in various fields such as image processing [41], flow control [42], and data reduction for dimensionality reduction [43]. POD is a method for representing an infinite dimensional system with a limited number of basis functions, known as “Basis Vectors” or “Orthogonal Functions” [40]. It is highly effective and widely used due to its optimality [44]. POD can also be used in conjunction with Galerkin Projection to derive low-order differential equations from high dimensional equations [45].

The first section will discuss a general matrix factorization technique that is commonly used for reducing high dimensional data in various fields. In the second section, the Proper Orthogonal Decomposition (POD) method will be presented, which is used to calculate the basis vectors of high dimensional data. The third section will cover the use of the Galerkin projection method, which can be used in combination with POD to derive low order differential equations from high dimensional equations.

4.1.1 Matrix Representation of High Dimensional Data

Consider a real variable $F(x, t)$ that depends on both space x and time t . To sample F , we take n snapshots at uniform time points $(t_0, \dots, t_i, \dots, t_n)$ where $t_i = i\Delta t$. Each snapshot consists of F at m uniform spatial points $(x_0, \dots, x_j, \dots, x_m)$ where $x_j = j\Delta x$. All snapshots are then collected in a matrix $\mathbf{X} \in \mathbf{R}^{(n_fm) \times n}$ after reshaping each snapshot into a column vector $X_i \in \mathbf{R}^{n_fm \times 1}$, where n_f is the number of components of F . If $n_f = 1$, the matrix \mathbf{X} has the following shape:

$$\mathbf{X} = \begin{bmatrix} X_1(x_1) & X_2(x_1) & X_3(x_1) & \cdots & X_i(x_1) & \cdots & X_n(x_1) \\ X_1(x_2) & X_2(x_2) & X_3(x_2) & \cdots & X_i(x_2) & \cdots & X_n(x_2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_1(x_m) & X_2(x_m) & X_3(x_m) & \cdots & X_i(x_m) & \cdots & X_n(x_m) \end{bmatrix} \quad (4.1)$$

Data decomposition techniques aim to represent a matrix as a sum of “modes”, which are rank one contributions characterized by a spatial component $\phi_r \in \mathbf{R}^{(n_fm) \times 1}$, a temporal component $\psi_r \in \mathbf{R}^{n \times 1}$, and an energy contribution scalar σ_r . The matrix \mathbf{X} can then be written as:

$$\mathbf{X} = \sum_{i=1}^{M=\min(n_fm, n)} \sigma_r \phi_r \psi_r^T \quad (4.2)$$

For the amplitudes σ_r to accurately represent the energy contribution of each mode, the spatial basis ψ_r and temporal basis ϕ_r should have unit norm [46]. Additionally, each spatial component ϕ_i should be orthogonal to all other spatial basis, and the same applies for the temporal basis ψ_i . Therefore, the temporal and spatial basis must satisfy the following conditions:

$$\|\phi_r\|_2^2 = \frac{1}{n_fm} \sum_{k=1}^{n_fm} \phi_r[k] \phi_r[k] = \frac{1}{n_fm} \langle \phi_r, \phi_r \rangle = \frac{1}{n_fm} \phi_r^T \phi_r = 1 \quad (4.3)$$

$$\langle \phi_i, \phi_j \rangle = 0 \quad (4.4)$$

$$\|\psi_r\|_2^2 = \frac{1}{n} \sum_{k=1}^n \psi_r[k] \psi_r[k] = \frac{1}{n} \langle \psi_r, \psi_r \rangle = \frac{1}{n} \psi_r^T \psi_r = 1 \quad (4.5)$$

$$\langle \psi_k, \psi_l \rangle = 0 \quad (4.6)$$

The orthonormal spatial basis can be arranged in a matrix $\boldsymbol{\phi} = [\phi_1 \ \phi_2 \ \cdots \ \phi_M]$, the orthonormal temporal basis can be arranged as well in matrix $\boldsymbol{\psi} = [\psi_1 \ \psi_2 \ \cdots \ \psi_M]$, and the amplitudes σ_r can be arranged in a diagonal matrix $\Sigma = \text{diag}(\sigma_1 \ \sigma_2 \ \cdots \ \sigma_M)$. As a result, \mathbf{X} can be written as:

$$\mathbf{X} = \boldsymbol{\phi} \Sigma \boldsymbol{\psi}^T \quad (4.7)$$

The energy content of matrix \mathbf{X} can be defined using the classical Frobenius norm [46] where:

$$\|\mathbf{X}\|_F^2 = \text{Tr}(K) = \text{Tr}(D^\dagger D) = \text{Tr}(\boldsymbol{\psi}\boldsymbol{\Sigma}\boldsymbol{\phi}^\dagger\boldsymbol{\phi}\boldsymbol{\Sigma}\boldsymbol{\psi}^\dagger) = \sum_{i=1}^M \lambda_i \quad (4.8)$$

The factorization shown in Equation 4.7 can be obtained either by computing $\boldsymbol{\phi}$ or $\boldsymbol{\psi}$, as both $\boldsymbol{\phi}$ and $\boldsymbol{\psi}$ have normalized columns and $\boldsymbol{\Sigma}$ is a diagonal matrix [46]. For instance, after computing $\boldsymbol{\psi}$, $\boldsymbol{\phi}$ and $\boldsymbol{\Sigma}$ can be obtained through a series of matrix inversion and column normalization. The procedure for computing $\boldsymbol{\phi}$ and $\boldsymbol{\Sigma}$ given $\boldsymbol{\psi}$ is outlined in Algorithm 2.

Algorithm 2 Computing Algorithm $\boldsymbol{\Sigma}$ and $\boldsymbol{\phi}$ given $\boldsymbol{\psi}$

```
# Compute C
C =  $\boldsymbol{\phi}\boldsymbol{\Sigma} = \mathbf{X}(\boldsymbol{\psi}^T)^{-1}$ 
# Compute the diagonal elements  $\sigma_i$  of  $\boldsymbol{\Sigma}$ 
 $\sigma_i = \|C_i\|_2$  Where  $C_i$  is  $i^{\text{th}}$  column of C
# Compute  $\boldsymbol{\phi}$ 
 $\boldsymbol{\phi} = C\boldsymbol{\Sigma}^{-1}$ 
```

There are several methods for computing the spatial and temporal basis $\boldsymbol{\psi}$, such as DFT (Discrete Fourier Transform) and Proper Orthogonal Decompositions (POD). In the following section, the POD technique for calculating the temporal basis $\boldsymbol{\psi}$ is described.

4.1.2 Proper Orthogonal Decomposition

Proper Orthogonal Decomposition (POD), introduced by Loeve in [47], is a method for finding a low-dimensional representation of a high-dimensional system by identifying its most energetic modes. These modes are made up of orthonormal spatial and temporal basis functions that are determined purely from the data matrix \mathbf{X} , unlike methods such as the Discrete Fourier Transform (DFT) that assume a prior form for the basis functions, such as Fourier modes or spherical harmonics.

An approximation of the matrix \mathbf{X} can be created by considering the first M' modes, which capture the highest amount of energy in the data contained in \mathbf{X}_{app} .

$$\mathbf{X}_{\text{app}} = \sum_{i=1}^{M'} \sigma_i \boldsymbol{\phi}_i \boldsymbol{\psi}_i^T = \boldsymbol{\phi}_{M'} \boldsymbol{\Sigma}_{M'} \boldsymbol{\psi}_{M'}^T$$

Here $\boldsymbol{\phi}_{M'} \in \mathbf{R}^{n_f m \times M'}$, $\boldsymbol{\psi}_{M'} \in \mathbf{R}^{(n) \times M'}$, and $\boldsymbol{\Sigma}_{M'} \in \mathbf{R}^{M' \times M'}$.

The orthonormal basis functions are calculated by minimizing the sum of squares between \mathbf{X} and its approximated form, \mathbf{X}_{app} , under the assumption that the first M' energetic modes provide the optimal description. The sum of squares between \mathbf{X} and \mathbf{X}_{app} is given by:

$$\text{SSE} = \|\mathbf{X} - \sum_{i=1}^{M'} \sigma_i \phi_i \psi_i^T\|_2^2 \quad (4.9)$$

The orthonormal temporal basis functions are the vectors that solve the following constrained minimization problem:

$$\min_{(\psi_1, \psi_2, \dots, \psi_{M'})} \|\mathbf{X} - \sum_{i=1}^{M'} \sigma_i \phi_i \psi_i^T\|_2^2 \quad (4.10)$$

Subject to

$$\langle \psi_i, \psi_j \rangle = \delta_{ij} \quad (4.11)$$

The orthonormal temporal basis functions ψ_i are the eigenvectors of the temporal correlation matrix $k = \mathbf{X}^\dagger \mathbf{X}$ [48]. The eigendecomposition of K can be written as:

$$K = \boldsymbol{\psi} \Lambda \boldsymbol{\psi}^T = \sum_{i=1}^M \psi_i \lambda_i \psi_i^T \quad (4.12)$$

By combining equation 4.7 and equation 4.12, the following relations can be deduced:

$$\mathbf{K} = \boldsymbol{\psi} \boldsymbol{\Sigma} \boldsymbol{\phi}^\dagger \boldsymbol{\phi} \boldsymbol{\Sigma} \boldsymbol{\psi}^\dagger \rightarrow \Lambda = \boldsymbol{\Sigma} \boldsymbol{\phi}^\dagger \boldsymbol{\phi} \boldsymbol{\Sigma}$$

The spatial basis $\boldsymbol{\phi}$ are orthonormal ($\boldsymbol{\phi}^T \boldsymbol{\phi} = \mathbf{I}$) thus:

$$\Lambda = \boldsymbol{\Sigma}^2 \rightarrow \sigma_i = \sqrt{\lambda_r}$$

The optimality of the POD is guaranteed by the Eckart-Young-Mirsky theorem [49], which states that the singular value of mode $\sigma_{M'+1}$ represents the sum of squared errors (L_2 norm) resulting from an approximation using the first M' modes. This can be written mathematically as:

$$\text{SSE} = \|\mathbf{X} - \sum_{i=1}^{M'} \sigma_i \phi_i \psi_i^T\|_2^2 = \sigma_{M'+1}^2 \quad (4.13)$$

Both the temporal basis $\boldsymbol{\psi}$ and spatial basis $\boldsymbol{\phi}$ computed by POD can also be obtained through the Singular Value Decomposition (SVD) of matrix \mathbf{X} .

4.1.3 Galerkin Projection

Complex systems are often described by partial differential equations with infinite degrees of freedom. To study these systems numerically, it is necessary to approximate them using a dynamical system with a finite number of dimensions [50]. This can be done using techniques such as proper orthogonal decomposition (POD) and Galerkin projection. POD is used to identify the most energetic “modes” of the complex system, and then Galerkin projection is used to derive low order differential equations for each of these energetic modes [45]. The following outlines the main steps for these projections.

The dynamics of a complex system can be described by the following partial differential equation:

$$\dot{x}(t) = f(x(t)) \quad (4.14)$$

where $f(x(t))$ is a nonlinear function of $x(t)$ and its derivative with respect to space.

To approximate the solution to this equation, we define a residual function $R(x, \dot{x})$ as follows:

$$R(x, \dot{x}) = \dot{x}(t) - f(x(t)) \quad (4.15)$$

Using the first n energetic modes, which consist of a spatial component ϕ_i , a temporal component ψ_i , and an amplitude σ_i , we can approximate $x(t)$ as follows:

$$x_n(t) = \sum_{i=1}^n \chi_i \psi_i(t) = \boldsymbol{\chi}_n \boldsymbol{\psi}_n^T \quad (4.16)$$

where $\chi_i = \sigma_i \phi_i$, $\boldsymbol{\chi}_n = [\sigma_1 \phi_1 \quad \sigma_2 \phi_2 \quad \cdots \quad \sigma_n \phi_n] = \boldsymbol{\phi}_n \boldsymbol{\Sigma}_n$, $\boldsymbol{\phi}_n = [\phi_1 \quad \phi_2 \quad \cdots \quad \phi_n]$, $\boldsymbol{\Sigma} = \text{diag} [\sigma_1 \quad \sigma_2 \quad \cdots \quad \sigma_n]$, and $\boldsymbol{\psi}_n = [\psi_1 \quad \psi_2 \quad \cdots \quad \psi_n]$.

By replacing $x(t)$ with $x_n(t)$ in equation 4.14, we obtain:

$$\sum_{i=1}^n \chi_i \dot{\psi}_i(t) = f\left(\sum_{i=1}^n \chi_i \psi_i(t)\right) \quad (4.17)$$

$$\boldsymbol{\chi}_n \dot{\boldsymbol{\psi}}^T = f(\boldsymbol{\chi}_n \boldsymbol{\psi}^T) \quad (4.18)$$

According to the principle of Galerkin projection, the projection of $R(x, \dot{x})$ onto the space spanned by the spatial vectors ϕ is equal to zero. This can be written as:

$$\langle R(x, \dot{x}), \phi_i \rangle = 0 \quad \forall j = 1, \cdots, n \quad (4.19)$$

Substituting $x_n(t)$ for $x(t)$ in equation 4.19 gives us:

$$\langle \chi_n \dot{\psi}^T, \phi_i \rangle = \langle f(\chi_n \psi^T), \phi_i \rangle \quad \forall j = 1, \dots, n \quad (4.20)$$

$$\phi_n^T \phi_n \Sigma_n \dot{\psi}_n^T = \phi_n^T f(\phi_n \Sigma_n \psi_n^T) \quad (4.21)$$

Therefore, the reduced order model of the partial differential equation can be obtained from equation 4.21 as follows:

$$\dot{\psi}_n^T = \Sigma_n^{-1} \phi_n^T f(\phi_n \Sigma_n \psi_n^T) \quad (4.22)$$

4.2 Signal Denoising Techniques

Signals that represent the evolution of a physical quantity over time often contain noise due to limitations in the experimental setup. To ensure accurate analysis of these signals, it is important to remove this noise before proceeding. There are various techniques that can be used for this purpose, such as wavelet transform [51] and digital filtering [52]. This section will focus specifically on digital filtering techniques.

Digital filters use a coefficient vector, represented by w , to take a distorted signal, y , as input and output a clean signal, x . This process is illustrated in schematic 4.1. The coefficient vector, w , can be expressed in both the time and frequency domains. In the time domain, it is referred to as the ‘‘Impulse Response’’, and in the frequency domain, it is referred to as the ‘‘Frequency Response’’.



Figure 4.1: Filter Representation

Filters can be classified into four categories based on the range of frequencies they allow, called the ‘‘pass band’’, and the range of frequencies they block, called the ‘‘stop band’’. These four types of filters are low-pass filters, high-pass filters, band-pass filters, and band-stop filters, as shown in Figure 4.2. Among these types, low-pass filters are frequently used because they remove unwanted noise by filtering out everything except for the main signal.

The transition between the ‘‘pass band’’ and ‘‘stop band’’ of a digital filter is determined by a specific cutoff frequency f_c . In reality, however, it is not possible to

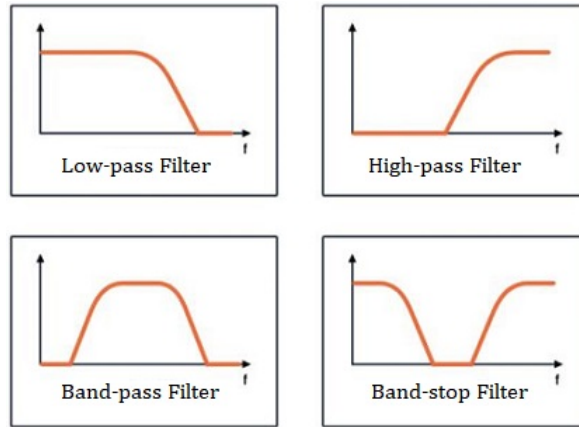


Figure 4.2: Different Filter Types

achieve an instantaneous transition between these two regions. To account for this, the "transition width" is introduced to describe how quickly the filter should transition between the pass band and stop band. This transition width can be observed in the red line of figure 4.3, which shows a realistic filter response with ripples and a finite transition width. In contrast, the green line in the same figure represents an ideal filter response with an abrupt transition and a zero transition width.

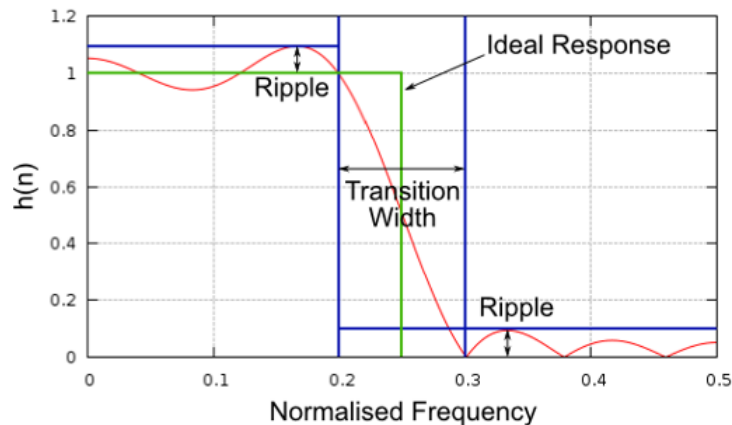


Figure 4.3: Filter Response To Frequencies

Moreover, filters can be classified based on the algorithmic implementation of their impulse response as either finite duration impulse response (FIR) or infinite duration impulse response (IIR). While IIR filters utilize both feed-forward and feedback terms in their implementation, FIR filters use only feed-forward terms. In this section, we will focus on FIR filters as they are more practical and easier to compute than IIR filters [52]. The relationship between the noisy signal y and the clean signal

y' in FIR filters is given by:

$$x(m) = \sum_{k=0}^{p-1} w_k y(m-k) = w^T y \quad (4.23)$$

Where $w = [w_0 \ w_1 \ w_2 \ \cdots \ w_{P-1}]$ and $y^T = [y(m) \ y(m-1) \ \cdots \ y(m-p-1)]$

This means that the output signal y' is obtained by convolving the coefficient vector w with the input signal y [52]. The FIR block diagram is shown in Figure 4.4, where the triangles represent the coefficients of the filter w_i and the squares represent the delay of the input signal y by one time step.

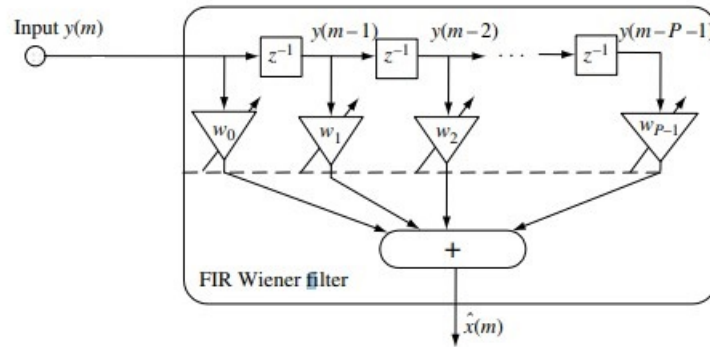


Figure 4.4: FIR Filter Block diagram [52]

In an FIR filter, the coefficient vector w is determined by minimizing the error $e = x - \hat{x}$, which is the difference between the output signal y and the target signal \hat{x} . The impulse response and frequency response of an FIR low pass filter are shown in Figure 4.5.

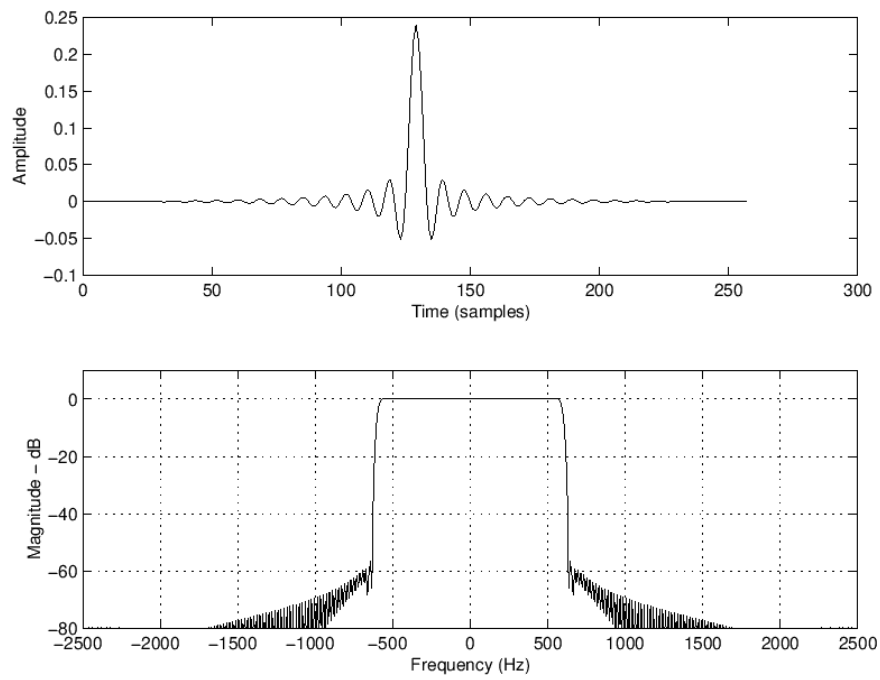


Figure 4.5: Frequency and Impulse Response for a low pass FIR filter [53]

CHAPTER 5

RESULTS

In this study, we use hydrodynamic, sparse regression, and model reduction techniques to model the collective behavior of runners in the Chicago Marathon. Our approach is summarized in Figure 5.1, with the first section of this chapter discussing the general characteristics of the marathon and the second section outlining the steps for deriving a partial differential equation for this system. The final section of this chapter explains the method for obtaining a low-order system of differential equations for this system.

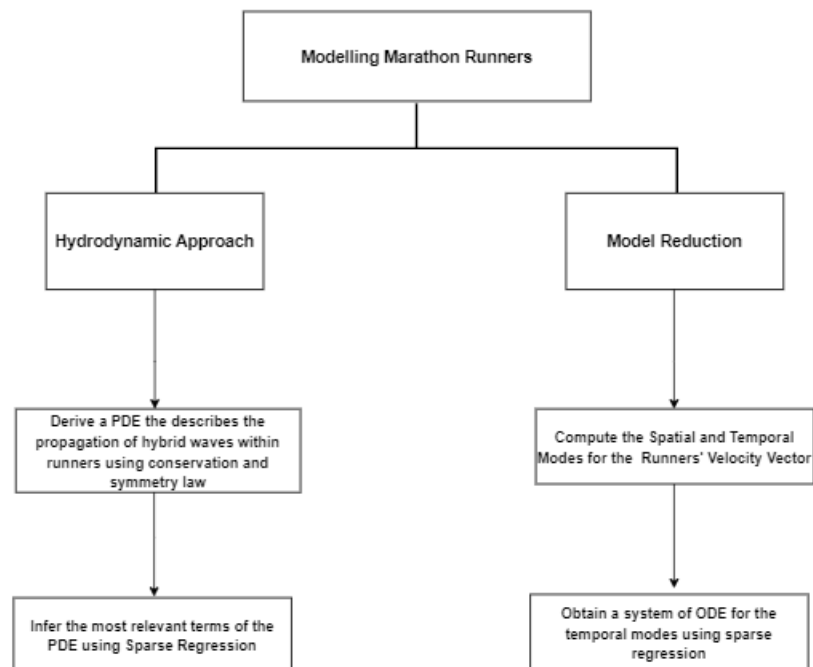


Figure 5.1: Summary of the Work done



Figure 5.2: Runners in Chicago Marathon

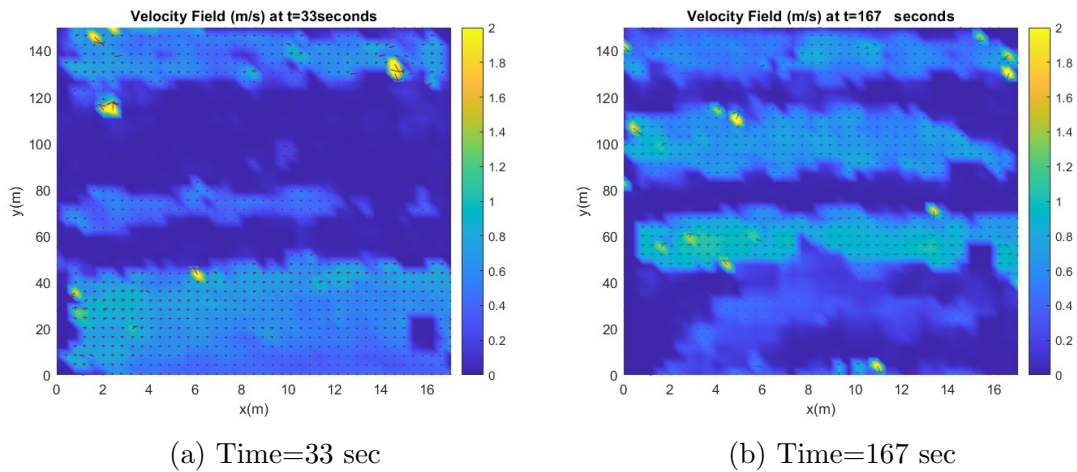


Figure 5.3: Velocity Fields at Two Instants

5.1 Marathon System Description

In the Chicago Marathon (Figure 5.2), thousands of runners are guided to the starting line by staff members who perform a series of stop-and-go waves that excite the runners. As a result, hybrid density and velocity waves propagate through the crowd with little attenuation (Video 1). The motion of the runners is treated as a continuum and characterized by velocity fields $\mathbf{v}(\mathbf{x}, t)$, polarization fields $\mathbf{p}(\mathbf{x}, t)$, and density fields $\rho(\mathbf{x}, t)$. Figures 5.3a and 5.3b show the velocity fields at times $t = 33$ seconds and $t = 167$ seconds, respectively.

In the kymograph shown in Figure 5.4, the longitudinal velocity component v_y is averaged over the transverse direction x at every time instant t . This kymograph clearly shows the presence of several longitudinal waves propagating upstream. A kymograph of a single one of these waves is shown in Figure 5.5.

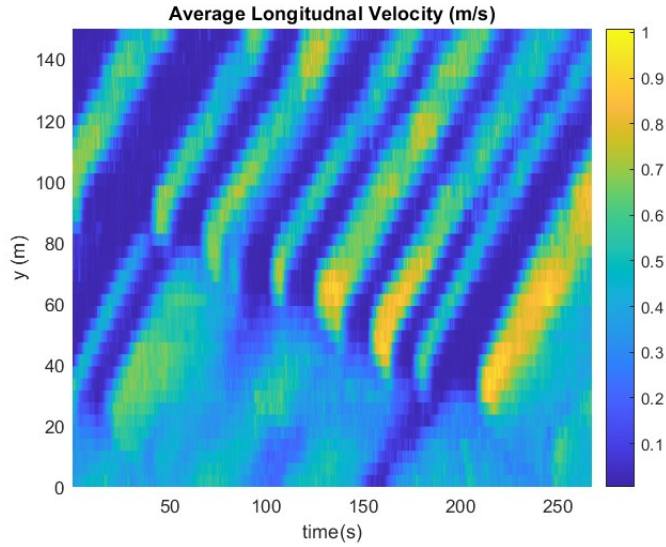


Figure 5.4: Longitudinal Velocity Kymograph

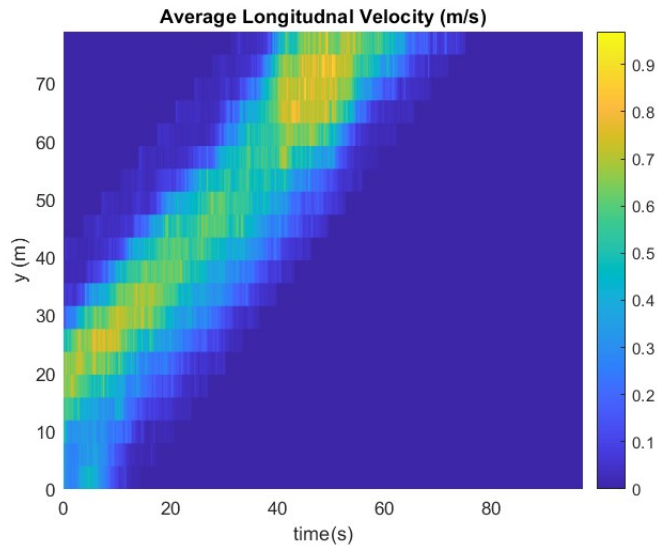


Figure 5.5: Kymograph for a single wave

The propagating density and velocity hybrid waves in the Chicago Marathon are linear waves, as their shape is the same as the imposed boundary excitations [3]. These linear waves can be characterized by their speed, u , and diffusion constant,

D , which we aim to determine by inferring a hydrodynamic equation for the velocity field.

5.2 Partial Differential Equation for Runners' Linear Waves

The hydrodynamic fields needed to describe the large-scale collective behavior of runners in the Chicago Marathon include the density field $\rho(\mathbf{x}, t)$, velocity field $\mathbf{v}(\mathbf{x}, t)$, and polarization vector $\mathbf{p}(\mathbf{x}, t)$, which represents the average direction of the runners at time t and position $\mathbf{x} = (x, y)$. The polarization vector can be expressed as:

$$\mathbf{p} = (\cos \psi(\mathbf{x}, t) \sin \psi(\mathbf{x}, t)) \quad (5.1)$$

Where $\psi(\mathbf{x}, t)$ is the angle between the orientation vector \mathbf{p} and the transverse direction x . The relationship between the velocity vector $\mathbf{v}(\mathbf{x}, t)$ and the orientational vector $\mathbf{p}(\mathbf{x}, t)$ is:

$$\mathbf{v}(\mathbf{x}, t) = v(\mathbf{x}, t)\mathbf{p}(\mathbf{x}, t) \quad (5.2)$$

Where $v(\mathbf{x}, t)$ is the speed of the runners at time t and position \mathbf{x} .

The density field $\rho(\mathbf{x}, t)$ changes over time according to the continuity equation:

$$\frac{\partial}{\partial t}\rho = -\nabla \cdot (\rho\mathbf{v}) \quad (5.3)$$

Linear momentum is not conserved in the motion of runners, unlike in conventional fluids, due to the friction force between the runners and the ground. This friction force causes the exchange of momentum between the runners and the ground, resulting in the non-conservation of linear momentum.

Therefore, the partial differential equation that described the evolution of momentum per unit mass (i.e $\rho\mathbf{v}$) is given by:

$$\frac{\partial}{\partial t}(\rho\mathbf{v}) + \mathbf{v} \cdot \nabla(\rho\mathbf{v}) = \nabla \cdot \sigma - \nabla(P) + f(\mathbf{v}, \rho, \mathbf{p}) \quad (5.4)$$

The term on the left side represents the time derivative of the momentum per unit mass and the convective derivative, which describes how the momentum changes as the runners move. The term on the right side includes the divergence of the stress tensor, which represents the forces exerted on the runners by their neighbors, the force due to pressure ($-\nabla(P)$), and the frictional force ($f(\mathbf{v}, \rho, \mathbf{p})$) between the runners and the ground. This equation takes into account the forces acting on the runners and how they affect the momentum of the system

The stress tensor σ is given by :

$$\sigma = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yz} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{yz} & \sigma_{zy} & \sigma_{zz} \end{bmatrix} \quad (5.5)$$

In the stress tensor matrix, σ_{ij} represents the ratio of the i^{th} force components on the j^{th} surface to the surface area in the vanishing area limit. It is defined as:

$$\sigma_{ij} = \lim_{dA \rightarrow 0} \frac{F_i^{ej}}{dA} \quad (5.6)$$

The frictional force f , the stress tensor σ , and the pressure term P are unknown quantities in the modeling of the collective behavior of runners in a marathon. To obtain the hydrodynamic equation for the velocity field, we adopt a method similar to that used by Toner and Tu in their work [20]. This method involves constructing possible terms that describe the evolution of the velocity field $\mathbf{v}(\mathbf{x}, t)$ using elements from a set S :

$$S = \nabla, \mathbf{v}, \rho, \psi, \mathbf{p} \quad (5.7)$$

It is important to note that the terms obtained from this set must be vectors.

The partial derivative of $\mathbf{v}(\mathbf{x}, t)$ can be written as:

$$\begin{aligned} \partial_t \mathbf{v} = & c_1 \mathbf{v} + c_2 \rho \mathbf{v} + c_3 \mathbf{p} + c_4 \rho \mathbf{p} + c_5 \nabla(\rho) + c_6 \nabla|\mathbf{v}|^2 + c_7 \nabla(\nabla \cdot \mathbf{v}) \\ & + c_8 (\nabla \cdot \mathbf{v}) \mathbf{v} + c_9 (\mathbf{v} \cdot \nabla) \rho \mathbf{v} + c_{10} \nabla^2 \mathbf{v} + c_{11} \nabla(\nabla \cdot (\rho \mathbf{v})) \\ & + c_{12} (\nabla \cdot (\rho \mathbf{v})) (\rho \mathbf{v}) + c_{13} ((\rho \mathbf{v}) \cdot \nabla) \rho \mathbf{v} + c_{14} \nabla^2 \rho \mathbf{v} + c_{15} \nabla(\nabla \cdot \mathbf{p}) \\ & + c_{16} (\nabla \cdot \mathbf{p}) \mathbf{v} + c_{17} (\mathbf{p} \cdot \nabla) \mathbf{p} + c_{18} \nabla^2 \mathbf{p} + c_{19} \nabla(\nabla \cdot (\rho \mathbf{p})) + c_{20} (\nabla \cdot (\rho \mathbf{p})) \rho \mathbf{p} \\ & + c_{21} (\rho \mathbf{p} \cdot \nabla) \rho \mathbf{p} + c_{22} \nabla^2 \rho \mathbf{p} + \dots \end{aligned} \quad (5.8)$$

Since the runners are heading toward the longitudinal direction, ψ can be written as:

$$\psi(\mathbf{x}, t) = \frac{\pi}{2} + \mathcal{O}(\nabla) \quad (5.9)$$

After plugging in equations 5.9, 5.1, and 5.2 into equation 5.8 and considering the linear terms only, the longitudinal component (y-component) of equation 5.8 becomes:

$$\frac{\partial}{\partial t} v = a_1 v + a_2 \rho + a_3 \frac{\partial \rho}{\partial y} + a_4 \frac{\partial v}{\partial y} + a_5 \frac{\partial^2 v}{\partial x^2} + a_6 \frac{\partial^2 v}{\partial y^2} + a_7 \frac{\partial^2 \rho}{\partial x^2} + a_8 \frac{\partial^2 \rho}{\partial y^2} \quad (5.10)$$

The terms on the right-hand side of the equation represent various forces acting on the runners. The first two terms, $a_1v + a_2\rho = -\gamma(v - \alpha\rho)$, represent the damping force between the ground and runners. The third and fourth terms, $a_3\frac{\partial\rho}{\partial y} + a_4\frac{\partial v}{\partial y} = \beta\frac{\partial\rho}{\partial y} + u\frac{\partial v}{\partial y}$, represent the gradient of anisotropic pressure that depends on both density and speed. The fifth and sixth terms, $a_5\frac{\partial^2 v}{\partial x^2} + a_6\frac{\partial^2 v}{\partial y^2}$, are shear and bulk viscous forces present in conventional fluids. Finally, the seventh and eighth terms represent dissipative forces due to density gradients.

After finding equation 5.10, we can use sparse regression to determine the hydrodynamic coefficients (a_1, a_2, \dots, a_8) in this equation. The process for doing this involves the following steps:

- The hydrodynamic fields $v(\mathbf{x}, t)$ and $\rho(\mathbf{x}, t)$ are evaluated at **nm** space-time points (n space points and m time points)
- The partial derivative of v with respect to time is evaluated (using finite difference method) at all space-time points, and then collected in Matrix V_t

$$V_t = \begin{pmatrix} \frac{\partial}{\partial t}v(x_0, t_0) \\ \frac{\partial}{\partial t}v(x_1, t_0) \\ \frac{\partial}{\partial t}v(x_2, t_0) \\ \vdots \\ \frac{\partial}{\partial t}v(x_n, t_0) \\ \vdots \\ \frac{\partial}{\partial t}v(x_n, t_m) \end{pmatrix} \quad (5.11)$$

- All the linear terms present are evaluated at all the spatiotemporal points and collected in Θ

$$\Theta = \begin{pmatrix} 1 & v(x_0, t_0) & \cdots & \frac{\partial^2}{\partial y^2}v(x_0, t_0) & \frac{\partial^2}{\partial x^2}\rho(x_0, t_0) & \frac{\partial^2}{\partial y^2}\rho(x_0, t_0) \\ 1 & v(x_1, t_0) & \cdots & \frac{\partial^2}{\partial y^2}v(x_1, t_0) & \frac{\partial^2}{\partial x^2}\rho(x_1, t_0) & \frac{\partial^2}{\partial y^2}\rho(x_1, t_0) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & v(x_n, t_m) & \cdots & \frac{\partial^2}{\partial y^2}v(x_n, t_m) & \frac{\partial^2}{\partial x^2}\rho(x_n, t_m) & \frac{\partial^2}{\partial y^2}\rho(x_n, t_m) \end{pmatrix} \quad (5.12)$$

- Evaluating equation 5.10 at all the spatiotemporal points results in a system of linear equations of the form

$$V_t = \Theta\xi \quad (5.13)$$

In this case, ξ is a sparse vector that contains all the hydrodynamic coefficients c_i and is obtained using the STLSQ algorithm described in Chapter 3. The parameter λ in the STLSQ algorithm is varied over a range of values from $[\lambda_{\min}, \lambda_{\max}]$. This results in a set of PDEs with an increasing number of terms being inferred (as shown in Table 5.1). The best PDE is chosen as the one that best fits the phenomenology

of the data.

Table 5.1: PDEs Obtained with their coefficients

Term	PDE 1	PDE 2 ←	PDE 3	PDE 4	PDE 5
$\frac{\partial v}{\partial y}$	-1.05	-1.053	-1.069	-1.0688	-1.069
$\frac{\partial^2 v}{\partial y^2}$	-	0.812	1.22	1.291	1.283
v	-	-	0.0193	0.0229	0.0233
ρ	-	-	-0.0024	-0.003	-0.0031
$\frac{\partial^2 v}{\partial x^2}$	-	-	0.004	0.00415	0.00419
$\frac{\partial \rho}{\partial y}$	-	-	-	-	-0.0044
$\frac{\partial^2 \rho}{\partial y^2}$	-	-	0.124	0.132	0.13127
$\frac{\partial^2 \rho}{\partial x^2}$	-	-	-	-0.005	-0.0005

After using a spectral PDE solver, Dedalus [54], to simulate the obtained PDEs, we found that PDE 2 in Table 5.1 was the best match for the phenomenological behavior of the original data.

$$\frac{\partial v}{\partial t} = -u \frac{\partial v}{\partial y} + D \frac{\partial^2 v}{\partial y^2} \quad (5.14)$$

where $u(\mathbf{m/s})$ is the hybrid wave speed and $D(\mathbf{m/s}^2)$ is its corresponding diffusion constant.

Figure 5.6 shows the kymograph for the simulated wave using equation 5.14. The kymograph in Figure 5.6 shows the simulated wave using equation 5.14. To determine the uncertainty of the coefficients in this equation, we use the bootstrapping method. This involves repeatedly sampling the data with replacement and applying a least square regression to the terms of equation 5.14 on each sample. This generates a set of estimates for the coefficients, which can be used to construct probability density functions for both u and D . The standard deviations of these functions are then used to quantify the uncertainty of these two quantities. The bootstrapping method is a robust way to estimate the uncertainty of a statistic because it takes into account the entire distribution of the data, rather than just assuming a normal distribution as in traditional statistical methods.

As a result, the final values for u and D are:

$$u = 1.230 \pm 0.019(\mathbf{m/s}) \quad (5.15)$$

$$D = 0.80 \pm 0.14(\mathbf{m}^2/\mathbf{s}) \quad (5.16)$$

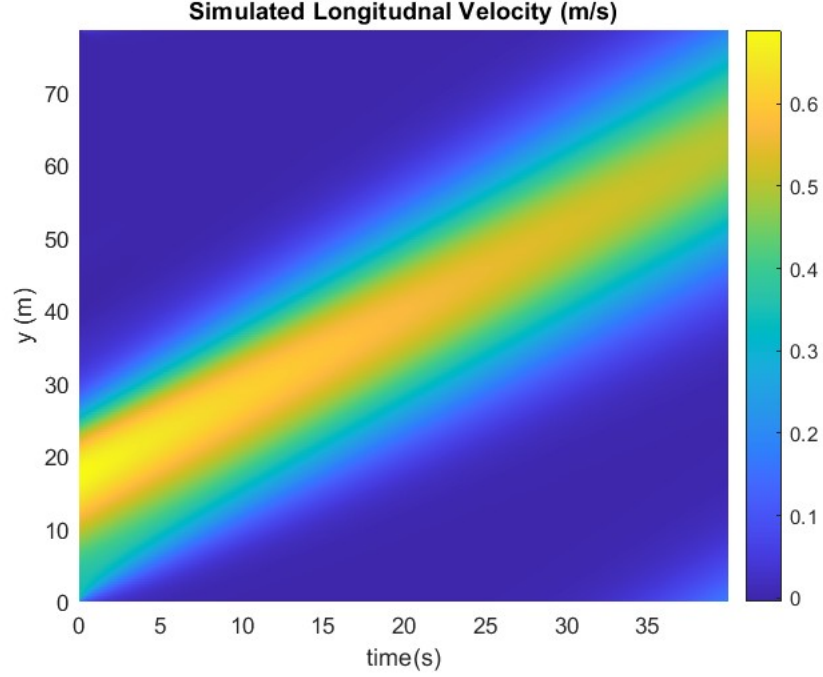


Figure 5.6: Kymograph for the velocity wave simulated from PDE 5.14

The wave speed u and diffusion constant D were also estimated in [3] by calculating the correlation $C_v(t, q)$ of the speed v in the vertical direction as a function of time t and wave vector q . The correlation was assumed to have the following form:

$$C_v(t, q) = \exp(-Dq^2t) \cos(uqt) \quad (5.17)$$

After fitting this equation to the data of the runners' speed v , the values obtained for u and D were

$$u' = 1.2 \pm 0.3(\mathbf{m/s}) \quad (5.18)$$

$$D' = 1 \pm 0.5(\mathbf{m}^2/\mathbf{s}) \quad (5.19)$$

The experimental intervals for the u, u', D , and D' (calculated within two standard deviations) are:

$$\text{Interval}_u = [1.192, 1.268] \quad (5.20)$$

$$\text{Interval}_{u'} = [0.6, 1.8] \quad (5.21)$$

$$\text{Interval}_D = [0.52, 1.08] \quad (5.22)$$

$$\text{Interval}_{D'} = [0, 2] \quad (5.23)$$

It is clear that the intervals for u and u' overlap, as do the intervals for D and D' .

5.3 Model Reduction Approach

After inferring the partial differential equation of the marathon runners' motion, our next goal is to obtain a system of low order differential equations for such motion. The procedure for this approach is the following:

- Calculate the main modes of the runners' motion using Proper orthogonal decomposition (POD).
- Infer a system of ordinary differential equations that describes the evolution of the mode's temporal components as function of time.

5.3.1 Proper Orthogonal Decomposition for the Motion of Runners

Before performing POD, the velocity vector measurement on every position (x, y) at each time instant is arranged in a vector \mathbf{q} of dimension $d = 2n_x n_y$ of the form :

$$\mathbf{q} = \begin{bmatrix} \mathbf{v}_x(x_1, y_1) \\ \mathbf{v}_x(x_2, y_1) \\ \vdots \\ \mathbf{v}_x(x_{n_x}, y_{n_y}) \\ \vdots \\ \mathbf{v}_y(x_1, y_1) \\ \mathbf{v}_y(x_2, y_1) \\ \vdots \\ \mathbf{v}_y(x_{n_x}, y_{n_y}) \end{bmatrix} \quad (5.24)$$

(5.25)

(5.26)

Then, the snapshots at all time instants t_1, \dots, t_n are collected into matrix D .

$$D = \begin{pmatrix} \mathbf{v}_x(x_1, y_1, t_1) & \mathbf{v}_x(x_1, y_1, t_2) & \cdots & \mathbf{v}_x(x_1, y_1, t_n) \\ \mathbf{v}_x(x_2, y_1, t_1) & \mathbf{v}_x(x_2, y_1, t_2) & \cdots & \mathbf{v}_x(x_2, y_1, t_n) \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{v}_x(x_{n_x}, y_{n_y}, t_1) & \mathbf{v}_x(x_{n_x}, y_{n_y}, t_2) & \cdots & \mathbf{v}_x(x_{n_x}, y_{n_y}, t_n) \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{v}_y(x_1, y_1, t_1) & \mathbf{v}_y(x_1, y_1, t_2) & \cdots & \mathbf{v}_y(x_1, y_1, t_n) \\ \mathbf{v}_y(x_2, y_1, t_1) & \mathbf{v}_y(x_2, y_1, t_2) & \cdots & \mathbf{v}_y(x_2, y_1, t_n) \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{v}_y(x_{n_x}, y_{n_y}, t_1) & \mathbf{v}_y(x_{n_x}, y_{n_y}, t_2) & \cdots & \mathbf{v}_y(x_{n_x}, y_{n_y}, t_n) \end{pmatrix} \quad (5.27)$$

Once we have collected the runners' velocity data in matrix D , we can apply Proper orthogonal decomposition (POD) to this matrix to compute the main modes of the runners' motion. These modes are composed of three components: singular value σ_i that determines the importance of each mode in describing the overall motion of runners, spatial component X_i that depends on the spatial coordinates only and temporal component ψ_i that depends on time only.

In Figure 5.8, we plot the decrease in the relative amplitude of mode i ($\frac{\sigma_i}{\sigma_1}$) as a function of the mode number. This plot shows how the relative importance of each mode decreases as the mode number increases.

Based on the plot in Figure 5.8, we can see that $\frac{\sigma_i}{\sigma_0}$ drops from 1 to 0.2 after just 10 modes. This suggests that the runners velocity vector $\mathbf{v}(x_i, y_i, t_k)$ can be accurately approximated using only the first r modes. Therefore, the velocity field at position (x_i, y_i) and instant t_k can be expressed as:

$$\mathbf{v}(x_i, y_i, t_k) = \bar{\mathbf{v}}(x_i, y_i) + \sum_{j=1}^r \sigma_j X_j(x_i, y_i) \psi_j(t_k) \quad (5.28)$$

Where $\bar{\mathbf{v}}(x_i, y_i)$ is the averaged velocity vector over time.

The spatial components of the first six modes are depicted in Figure 5.9. To prepare the temporal modes obtained from POD for analysis, we first apply a low pass FIR filter to denoise them. Figure 5.7 shows the temporal component of Mode 2 before and after denoising, and Figure 5.10 shows the temporal components of the first six modes after denoising.

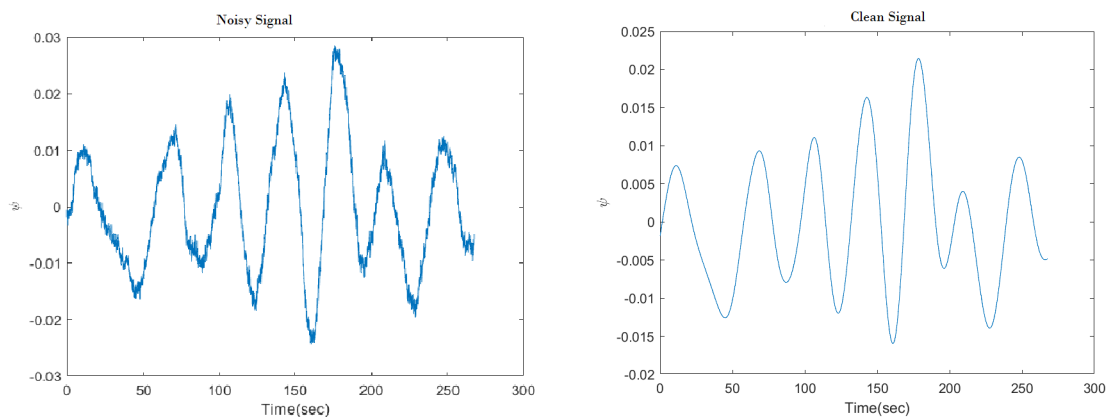


Figure 5.7: ψ_2 before and after Denoising

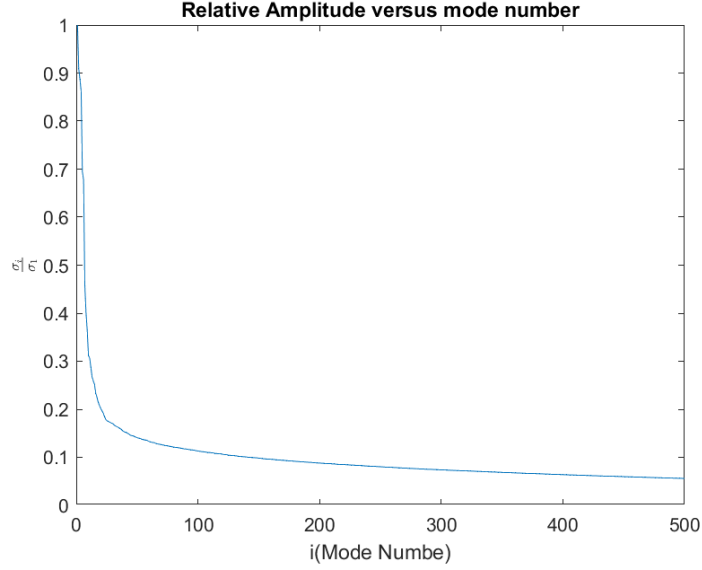


Figure 5.8: Relative amplitude versus Mode number

5.3.2 Inferring a system of ODEs for the temporal structures ψ_i

After calculating the POD modes for the runner's motion, our goal is to infer a system of ordinary differential equations that describes how the temporal modes ψ_i evolve over time. This system of equations will have the form:

$$\dot{\psi}_i(t) = \sum_{j=1}^r c_{ij} \psi_j + \sum_{i,j=1}^r d_{ijk} \psi_j \psi_k \quad (5.29)$$

The system of ODEs in equation 5.29 describes the dynamics of the first r temporal modes in terms of linear and nonlinear functions of these modes.

To obtain the coefficients c_{ij} and d_{ijk} in equation 5.29, we use the STLSQ algorithm. Similar to what we did in the previous section, we first collect the first r temporal modes and their derivatives with respect to time in matrices X and X' , respectively. Each column in X_i (X'_i) represents the evolution of ψ_i (ψ'_i) at all time instants (t_0, \dots, t_n) . Then, we evaluate all the candidate terms in equation 5.32 at all time instants and collect them in matrix Θ (Equation 5.30).

$$\Theta = \begin{pmatrix} \psi_1(t_1) & \cdots & \psi_r(t_1) & \psi_1^2(t_1) & \cdots & \psi_r \psi_1(t_1) & \cdots & \psi_r^2(t_1) \\ \psi_1(t_2) & \cdots & \psi_r(t_2) & \psi_1^2(t_2) & \cdots & \psi_r \psi_1(t_2) & \cdots & \psi_r^2(t_2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \psi_1(t_n) & \cdots & \psi_r(t_n) & \psi_1^2(t_n) & \cdots & \psi_r \psi_1(t_n) & \cdots & \psi_r^2(t_n) \end{pmatrix} \quad (5.30)$$

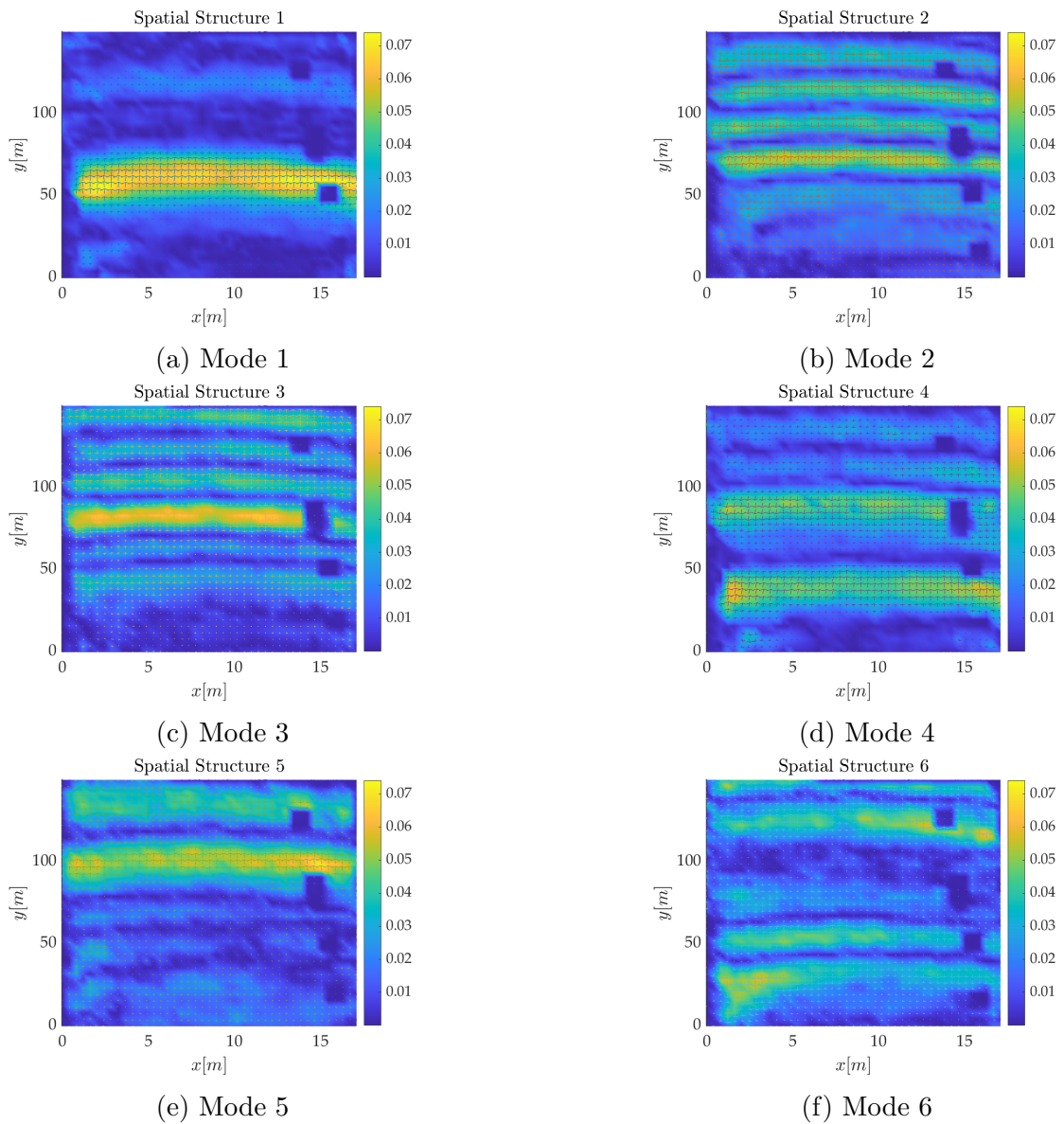
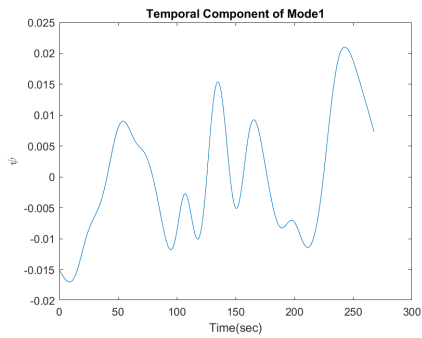
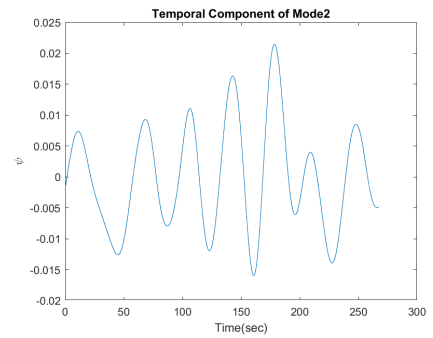


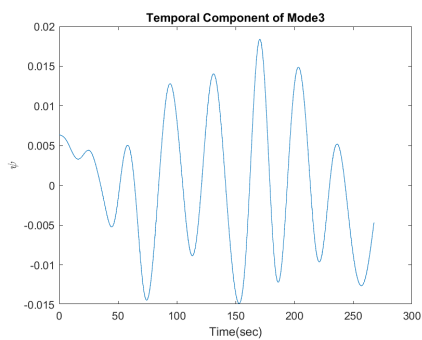
Figure 5.9: Spatial Modes X



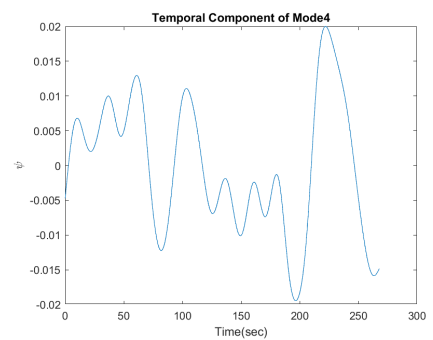
(a) Temporal Mode 1



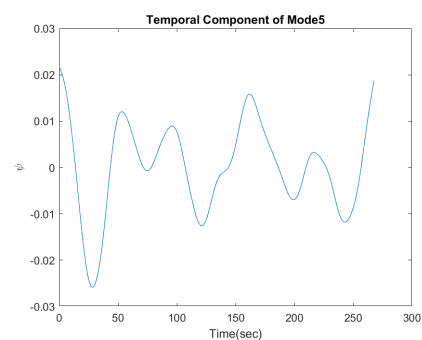
(b) Temporal Mode 2



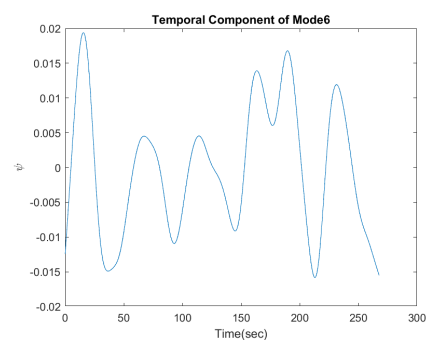
(c) Temporal Mode 3



(d) Temporal Mode 4



(e) Temporal Mode 5



(f) Temporal Mode 6

Figure 5.10: Temporal Modes ψ

The hyperparameters λ and r need to be carefully chosen to ensure that the system of ODEs accurately describes the dynamics of the runners' motion. λ controls the complexity of the ODEs, while r determines the number of temporal modes included in the system.

To find the optimal values for λ and r , we first divide the time series data into a training set (80 percent of the data) and a testing set (20 percent of the data). We then use Algorithm 3 to determine the values that give the best results.

Algorithm 3 Hyper parameter tuning for r and λ

For r in $[1, \dots, r_{\max}]$:

For λ in $[\lambda_{\min}, \dots, \lambda_{\max}]$:

- Compute Ξ using STLSQ algorithm using the training data
- Simulate the obtained ODE in testing data interval
- Calculate the normalized Frobenius that reflects the difference between the testing data and the simulated testing data. The Frobenius error is given by:

$$\text{Normalized Frobenius Error} = \frac{\sum_{j=1}^r (\sum_{i=1}^{N_t} (\psi_{\text{test}j}(t_i) - \psi_{\text{simulated}j}(t_i))^2 \sigma_j^2)}{\sum_{j=1}^r (\sum_{i=1}^{N_t} (\psi_{\text{test}j}(t_i))^2 \sigma_j^2)} \quad (5.31)$$

- The pair $(r_{\min}, \lambda_{\min})$ that have the minimum Frobenius error are the best values for r and λ
-

After following the steps in Algorithm 3, we can plot the Frobenius testing error as a function of the number of modes r and the thresholding parameter λ . This plot, shown in Figure 5.11, allows us to determine the optimal values for these hyperparameters. In this case, the minimum Frobenius testing error is achieved when $\lambda = 0.038$ and $r = 6$.

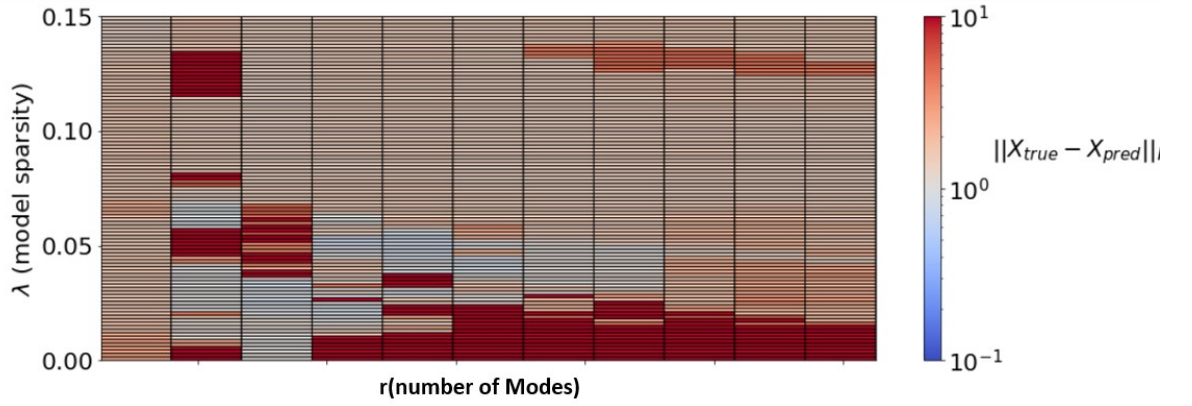


Figure 5.11: Testing error as function of r and λ

Thus, the obtained system of ODE for the temporal Modes for the optimal values of r and λ is:

$$\begin{cases} \dot{\psi}_1 = -0.06\psi_2 + 0.057\psi_4 - 0.904\psi_3\psi_5 - 0.159\psi_5\psi_5 \\ \dot{\psi}_2 = 0.07\psi_1 + 0.143\psi_3 \\ \dot{\psi}_3 = -0.14\psi_2 - 0.054\psi_4 + 0.059\psi_6 - 0.298\psi_2\psi_3 \\ \dot{\psi}_4 = -0.046\psi_1 + 0.063\psi_3 + 0.042\psi_5 - 0.043\psi_6 + 0.355\psi_2\psi_3 - 0.809\psi_3\psi_5 - 0.196\psi_6\psi_6 \\ \dot{\psi}_5 = -0.047\psi_4 - 0.099\psi_6 \\ \dot{\psi}_6 = -0.046\psi_3 + 0.053\psi_4 + 0.073\psi_5 \end{cases} \quad (5.32)$$

Figure 5.12 shows a heat map of the terms included in each ODE in equation 5.32. Additionally, Figure 5.13 compares the simulated data to the real testing data.

5.3.3 Stability Analysis of the Obtained ODE

After deriving a system of ODEs, we aim to find the fixed points of these equations and test their stability. To do this, we set the right-hand side of equation 5.32 to zero.

$$\begin{cases} -0.06\psi_2 + 0.057\psi_4 - 0.904\psi_3\psi_5 - 0.159\psi_5\psi_5 = 0 \\ 0.07\psi_1 + 0.143\psi_3 = 0 \\ -0.14\psi_2 - 0.054\psi_4 + 0.059\psi_6 - 0.298\psi_2\psi_3 = 0 \\ -0.046\psi_1 + 0.063\psi_3 + 0.042\psi_5 - 0.043\psi_6 + 0.355\psi_2\psi_3 - 0.809\psi_3\psi_5 - 0.196\psi_6\psi_6 = 0 \\ -0.047\psi_4 - 0.099\psi_6 = 0 \\ -0.046\psi_3 + 0.053\psi_4 + 0.073\psi_5 = 0 \end{cases} \quad (5.33)$$

The solution to the system of equations in 5.33 is a fixed point where all six temporal

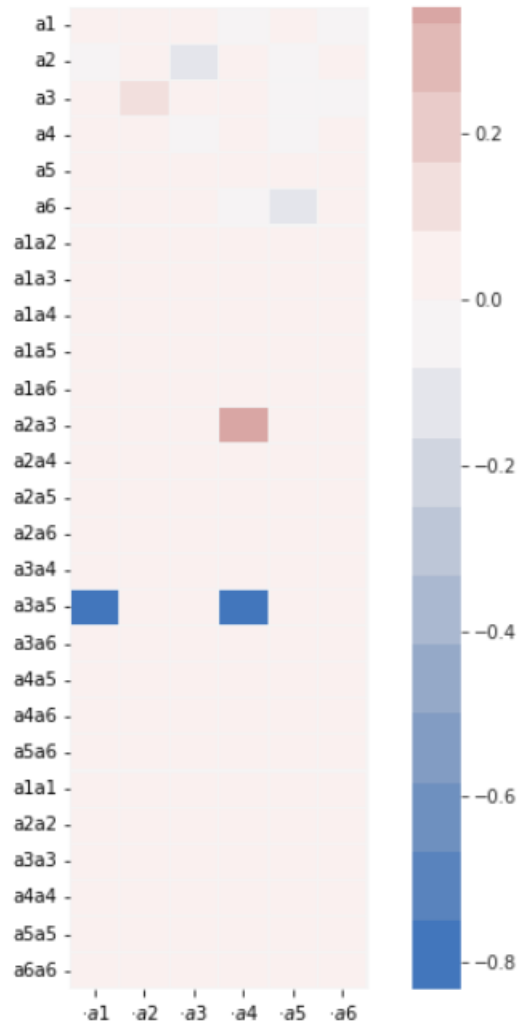


Figure 5.12: Heat Map for the Inferred ODE

modes are equal to zero.

$$\psi_{\text{fixed}} = \begin{cases} \psi_1 = 0 \\ \psi_2 = 0 \\ \psi_3 = 0 \\ \psi_4 = 0 \\ \psi_5 = 0 \\ \psi_6 = 0 \end{cases} \quad (5.34)$$

To analyze the stability of the fixed point, we compute the Jacobian of equation 5.34 and find its eigenvalues and eigenvectors. This allows us to determine whether the fixed point is stable or unstable.

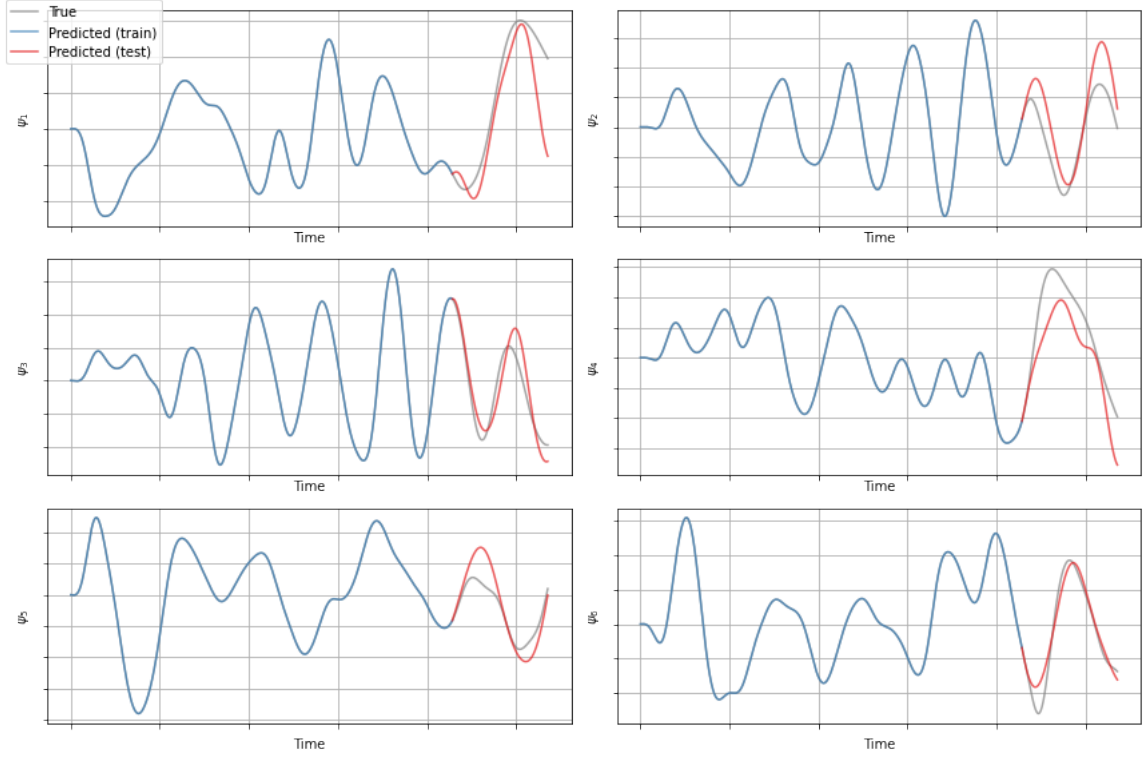


Figure 5.13: Comparison Between Prediction and Real Values for the temporal modes ψ_i

The jacobian of equation 5.33 is defined as:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial \dot{\psi}_1}{\partial \psi_1} & \frac{\partial \dot{\psi}_1}{\partial \psi_2} & \dots & \frac{\partial \dot{\psi}_1}{\partial \psi_6} \\ \frac{\partial \dot{\psi}_2}{\partial \psi_1} & \frac{\partial \dot{\psi}_2}{\partial \psi_2} & \dots & \frac{\partial \dot{\psi}_2}{\partial \psi_6} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial \dot{\psi}_6}{\partial \psi_1} & \frac{\partial \dot{\psi}_6}{\partial \psi_2} & \dots & \frac{\partial \dot{\psi}_6}{\partial \psi_6} \end{bmatrix} \quad (5.35)$$

Evaluating 5.35 at the fixed point (Equation 5.34) yields

$$\mathbf{J} = \begin{bmatrix} 0 & -0.06 & 0 & 0.057 & 0 & 0 \\ 0.07 & 0 & 0.143 & 0 & 0 & 0 \\ 0 & -0.14 & 0 & 0.054 & 0 & 0.059 \\ -0.046 & 0 & 0.063 & 0 & 0.042 & -0.043 \\ 0 & 0 & 0 & -0.047 & 0 & -0.099 \\ 0 & 0 & -0.046 & 0.053 & 0.073 & 0 \end{bmatrix} \quad (5.36)$$

The eigenvalue analysis of Matrix 5.36 gives us six eigenvalues $(\lambda_1, \lambda_2, \dots, \lambda_6)$,

which can be real or complex numbers. If the real parts of all six eigenvalues are negative, then the fixed point in equation 5.32 is stable. Otherwise, it is an unstable fixed point.

The six eigenvalues, along with their corresponding eigenvectors, are as follows:

$$\begin{cases} \lambda_1 = -0.00968298 + 0.15736839j \\ \lambda_2 = -0.00968298 - 0.15736839j \\ \lambda_3 = -0.00210766 + 0.04644548j \\ \lambda_4 = -0.00210766 - 0.04644548j \\ \lambda_5 = 0.01179065 + 0.10463857j \\ \lambda_6 = 0.01179065 - 0.10463857j \end{cases} \quad (5.37)$$

Moreover, the eigen vector ψ_i^* corresponding to each eigenvalue λ_i :

$$\begin{pmatrix} \psi_1^* \\ \psi_2^* \\ \psi_3^* \\ \psi_4^* \\ \psi_5^* \\ \psi_6^* \end{pmatrix} = \begin{pmatrix} -0.027 + 0.18j & 0.65 & -0.03 + 0.6j & 0.18 - 0.1j & 0.079 - 0.13j & 0.28 - 0.08j \\ -0.027 - 0.18j & 0.65 & -0.030 - 0.6j & 0.18 + 0.104j & 0.079 + 0.13j & -0.28 + 0.09j \\ -0.14 - 0.4j & 0.18 + 0.09j & 0.035 + 0.26j & 0.5 & -0.5 - 0.015j & -0.27 + 0.24j \\ -0.14 + 0.4j & 0.18 - 0.09j & 0.035 - 0.26j & 0.5 & -0.5 + 0.014j & -0.27 - 0.24j \\ 0.11 + 0.11j & 0.54 & -0.011 + 0.3j & 0.4 + 0.23j & -0.0021 + 0.5j & 0.3 - 0.16j \\ 0.11 - 0.11j & 0.54 & -0.011 - 0.3j & 0.4 - 0.23j & -0.002 - 0.5j & 0.3 + 0.16j \end{pmatrix} \begin{pmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \\ \psi_5 \\ \psi_6 \end{pmatrix} \quad (5.38)$$

As a result, the temporal basis ψ_i can be written as a function of the eigenvectors of the jacobian ψ_i^* .

$$\begin{pmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \\ \psi_5 \\ \psi_6 \end{pmatrix} = \begin{pmatrix} -0.19 - 0.39j & -0.19 + 0.39j & -0.15 + 0.8j & -0.15 - 0.8j & 0.4 + 0.04j & 0.4 - 0.04j \\ 0.6 - 0.16j & 0.6 + 0.16j & -0.030 - 0.6j & 0.18 + 0.104j & 0.079 + 0.13j & -0.28 + 0.09j \\ -0.14 - 0.4j & 0.18 + 0.09j & 0.035 + 0.26j & 0.5 & -0.5 - 0.015j & -0.27 + 0.24j \\ -0.14 + 0.4j & 0.18 - 0.09j & 0.035 - 0.26j & 0.5 & -0.5 + 0.014j & -0.27 - 0.24j \\ 0.11 + 0.11j & 0.54 & -0.011 + 0.3j & 0.4 + 0.23j & -0.0021 + 0.5j & 0.3 - 0.16j \\ 0.11 - 0.11j & 0.54 & -0.011 - 0.3j & 0.4 - 0.23j & -0.002 - 0.5j & 0.3 + 0.16j \end{pmatrix} \begin{pmatrix} \psi_1^* \\ \psi_2^* \\ \psi_3^* \\ \psi_4^* \\ \psi_5^* \\ \psi_6^* \end{pmatrix} \quad (5.39)$$

The results of our eigenvalue calculation indicate an unstable fixed point, with two eigenvalues having positive real components that are significantly larger in magnitude than the four negative ones. This instability is to be expected as, in a system where staff continuously excite runners, a scenario (Figure 5.14) where there are no excitations and runners are simply heading towards the starting point is not a stable configuration. The velocity field in this scenario is equal to the average velocity vector over time, represented by $v(x_i, y_i, t_k) = \hat{v}(x_i, y_i)$.

5.4 Comparison Between the Approaches

In this study, two methods were employed to model the upstream propagation of velocity and density waves among marathon runners. The first approach utilized

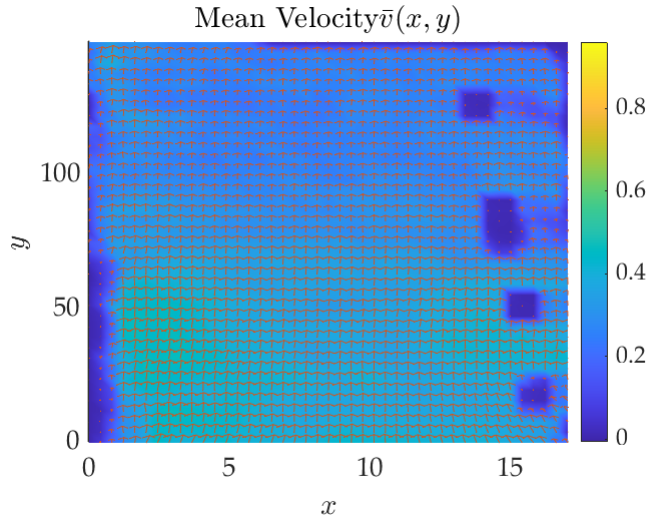


Figure 5.14: Mean Velocity

sparse regression and hydrodynamics to derive a partial differential equation (PDE) for these waves, while the second approach employed data reduction techniques to formulate an ordinary differential equation (ODE) for the first six temporal modes. To validate and compare these two methods, the authors will proceed as follows:

- The Galerkin projection can be applied to the PDE obtained through sparse regression to derive an ODE for the temporal modes. This ODE can then be compared with the ODE obtained through data reduction techniques.
- The density field was modeled using the Continuity equation. However, this equation was not derived from the marathon data. To validate this approach, the velocity obtained from solving the PDE can be used to calculate the density at each time instance, and this calculated density can be compared with the actual density values.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

In this thesis, we infer a partial differential equation for the hybrid velocity and density wave that propagates upstream within runners in a marathon, using video data. This approach successfully determines the wave speed, v , and its diffusion constant, D , which are in agreement with those obtained by Bain et al in [3]. In contrast to modern machine learning approaches such as neural networks [55] and model-free methods [56], [57], our approach provides a physics-informed model that takes into account symmetry and conservation laws. This allows scientists to better understand the dynamics of the system by identifying the relevant terms in the partial differential equation, rather than just developing a model that accurately predicts the dynamics without giving insight into how the system evolves or what factors control this evolution.

In addition to inferring a PDE for the runners' collective motion, we also described the evolution of the first r temporal modes using a system of ordinary differential equations. We proposed a method for determining r , the number of temporal modes involved, by dividing the data into training and testing sets and choosing the values of (r, λ) that result in the highest accuracy on the testing set rather than using the ratio of singular values $(\frac{\sigma_r}{\sigma_0})$.

Our modeling approach provides important insights into crowd management at marathon events. The partial differential equation we obtained is independent of the horizontal axis (x-direction), which implies that exciting or stopping the runners from the side boundaries is irrelevant. Using the wave speed we obtained v , we can estimate the time it will take for staff members to stop or excite the runners at a specific position y based on the initial conditions of the staff. Additionally, we can infer that changing the direction of the running crowd can only be achieved by giving orientational direction through the staff members directly, rather than through signals that are easily accessible to each individual runner.

In future work, there are several directions we could take to further improve our understanding of crowd management. One possibility is to apply our tool to sys-

tems with increased degrees of freedom in both directions to assess its effectiveness under more complex conditions. By studying the performance of the partial differential equation we developed in situations where runners have more flexibility in their movements, we can gain a better understanding of how to handle crowds in a range of scenarios. It will also be interesting to examine the usefulness of using staff members to direct crowds in these situations where there are more degrees of freedom.

To ensure the robustness of our model, it would be useful to test it on data from a variety of different marathon events and large gatherings. This would allow us to see how well the model generalizes to different settings and understand any factors that may influence its performance. In addition to testing our model on data from real-world events, we could use computer simulations to study the behavior of simulated crowds in a range of scenarios. This would allow us to investigate situations that may be difficult or impossible to study in the real world.

BIBLIOGRAPHY

- [1] T. Vicsek and A. Zafeiris, “Collective motion,” *Physics Reports*, vol. 517, no. 3, pp. 71–140, 2012, ISSN: 0370-1573. DOI: <https://doi.org/10.1016/j.physrep.2012.03.004>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0370157312000968>.
- [2] D. S. Calovi, U. Lopez, S. Ngo, C. Sire, H. Chaté, and G. Theraulaz, “Swarming, schooling, milling: Phase diagram of a data-driven fish school model,” *New journal of Physics*, vol. 16, no. 1, p. 015 026, 2014.
- [3] N. Bain and D. Bartolo, “Dynamic response and hydrodynamics of polarized crowds,” *Science*, vol. 363, no. 6422, pp. 46–49, 2019.
- [4] J. L. Silverberg, M. Bierbaum, J. P. Sethna, and I. Cohen, “Collective motion of humans in mosh and circle pits at heavy metal concerts,” *Phys. Rev. Lett.*, vol. 110, p. 228 701, 22 May 2013. DOI: [10.1103/PhysRevLett.110.228701](https://doi.org/10.1103/PhysRevLett.110.228701). [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevLett.110.228701>.
- [5] H. Voss, M. J. Bünner, and M. Abel, “Identification of continuous, spatiotemporal systems,” *Phys. Rev. E*, vol. 57, pp. 2820–2823, 3 Mar. 1998. DOI: [10.1103/PhysRevE.57.2820](https://doi.org/10.1103/PhysRevE.57.2820). [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevE.57.2820>.
- [6] D. Helbing and P. Molnar, “Social force model for pedestrian dynamics,” *Physical review E*, vol. 51, no. 5, p. 4282, 1995.
- [7] D. Helbing, I. Farkas, and T. Vicsek, “Simulating dynamical features of escape panic,” *Nature*, vol. 407, no. 6803, pp. 487–490, 2000.
- [8] M. C. Marchetti, J. F. Joanny, S. Ramaswamy, *et al.*, “Hydrodynamics of soft active matter,” *Rev. Mod. Phys.*, vol. 85, pp. 1143–1189, 3 Jul. 2013. DOI: [10.1103/RevModPhys.85.1143](https://doi.org/10.1103/RevModPhys.85.1143). [Online]. Available: <https://link.aps.org/doi/10.1103/RevModPhys.85.1143>.
- [9] H. Wioland, F. G. Woodhouse, J. Dunkel, J. O. Kessler, and R. E. Goldstein, “Confinement stabilizes a bacterial suspension into a spiral vortex,” *Phys. Rev. Lett.*, vol. 110, p. 268 102, 26 Jun. 2013. DOI: [10.1103/PhysRevLett.110.268102](https://doi.org/10.1103/PhysRevLett.110.268102). [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevLett.110.268102>.

- [10] S. H. Rudy, S. L. Brunton, J. L. Proctor, and J. N. Kutz, “Data-driven discovery of partial differential equations,” *Science Advances*, vol. 3, no. 4, e1602614, 2017. DOI: [10.1126/sciadv.1602614](https://doi.org/10.1126/sciadv.1602614). eprint: <https://www.science.org/doi/pdf/10.1126/sciadv.1602614>. [Online]. Available: <https://www.science.org/doi/abs/10.1126/sciadv.1602614>.
- [11] Z. Long, Y. Lu, X. Ma, and B. Dong, “PDE-net: Learning PDEs from data,” in *Proceedings of the 35th International Conference on Machine Learning*, J. Dy and A. Krause, Eds., ser. Proceedings of Machine Learning Research, vol. 80, PMLR, Oct. 2018, pp. 3208–3216. [Online]. Available: <https://proceedings.mlr.press/v80/long18a.html>.
- [12] U. N. D. of Economic and P. D. Social Affairs, “World population prospects 2022: Summary of results,” 2022.
- [13] D. Helbing, A. Johansson, and H. Z. Al-Abideen, “Dynamics of crowd disasters: An empirical study,” *Physical review E*, vol. 75, no. 4, p. 046 109, 2007.
- [14] M. Ballerini, N. Cabibbo, R. Candelier, *et al.*, “Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study,” *Proceedings of the National Academy of Sciences*, vol. 105, no. 4, pp. 1232–1237, 2008. DOI: [10.1073/pnas.0711437105](https://doi.org/10.1073/pnas.0711437105). [Online]. Available: <https://www.pnas.org/doi/abs/10.1073/pnas.0711437105>.
- [15] D. Pavlov, A. Kasumyan, *et al.*, “Patterns and mechanisms of schooling behavior in fish: A review,” *Journal of Ichthyology*, vol. 40, no. 2, S163, 2000.
- [16] D. Helbing and A. Johansson, “Pedestrian, crowd, and evacuation dynamics,” *arXiv preprint arXiv:1309.1609*, 2013.
- [17] F. Schweitzer and J. Farmer, *Brownian Agents and Active Particles: Collective Dynamics in the Natural and Social Sciences*, ser. Springer Series in Synergetics. Springer, 2007, ISBN: 9783540738442. [Online]. Available: <https://books.google.com.lb/books?id=Son9nY3FcVEC>.
- [18] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, “Novel type of phase transition in a system of self-driven particles,” *Phys. Rev. Lett.*, vol. 75, pp. 1226–1229, 6 Aug. 1995. DOI: [10.1103/PhysRevLett.75.1226](https://doi.org/10.1103/PhysRevLett.75.1226). [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevLett.75.1226>.
- [19] P. M. Chaikin and T. C. Lubensky, *Principles of Condensed Matter Physics*. Cambridge University Press, 1995. DOI: [10.1017/CB09780511813467](https://doi.org/10.1017/CB09780511813467).
- [20] J. Toner, Y. Tu, and S. Ramaswamy, “Hydrodynamics and phases of flocks,” *Annals of Physics*, vol. 318, no. 1, pp. 170–244, 2005, Special Issue, ISSN: 0003-4916. DOI: <https://doi.org/10.1016/j.aop.2005.04.011>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0003491605000540>.
- [21] S. Ramaswamy and M. Rao, “Active-filament hydrodynamics: Instabilities, boundary conditions and rheology,” *New Journal of Physics*, vol. 9, no. 11, p. 423, Nov. 2007. DOI: [10.1088/1367-2630/9/11/423](https://doi.org/10.1088/1367-2630/9/11/423). [Online]. Available: <https://dx.doi.org/10.1088/1367-2630/9/11/423>.

- [22] A. Ahmadi, M. C. Marchetti, and T. B. Liverpool, “Hydrodynamics of isotropic and liquid crystalline active polymer solutions,” *Phys. Rev. E*, vol. 74, p. 061 913, 6 Dec. 2006. DOI: [10.1103/PhysRevE.74.061913](https://doi.org/10.1103/PhysRevE.74.061913). [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevE.74.061913>.
- [23] K. Kruse and F. Jülicher, “Self-organization and mechanical properties of active filament bundles,” *Phys. Rev. E*, vol. 67, p. 051 913, 5 May 2003. DOI: [10.1103/PhysRevE.67.051913](https://doi.org/10.1103/PhysRevE.67.051913). [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevE.67.051913>.
- [24] X. Serra-Picamal, V. Conte, R. Vincent, *et al.*, “Mechanical waves during tissue expansion,” English, *Nature Physics*, vol. 8, no. 8, pp. 628–634, Aug. 2012, ISSN: 1745-2473. DOI: [10.1038/nphys2355](https://doi.org/10.1038/nphys2355).
- [25] P. M. Bendix, G. H. Koenderink, D. Cuvelier, *et al.*, “A quantitative analysis of contractility in active cytoskeletal protein networks.,” *Biophysical journal*, vol. 94 8, pp. 3126–36, 2008.
- [26] V. Narayan, S. Ramaswamy, and N. Menon, “Long-lived giant number fluctuations in a swarming granular nematic,” *Science*, vol. 317, no. 5834, pp. 105–108, 2007. DOI: [10.1126/science.1140414](https://doi.org/10.1126/science.1140414). eprint: <https://www.science.org/doi/pdf/10.1126/science.1140414>. [Online]. Available: <https://www.science.org/doi/abs/10.1126/science.1140414>.
- [27] W. Paxton, K. Kistler, C. Olmeda, *et al.*, “Catalytic nanomotors: Autonomous movement of striped nanorods,” *Journal of the American Chemical Society*, vol. 126, pp. 13 424–31, Nov. 2004. DOI: [10.1021/ja047697z](https://doi.org/10.1021/ja047697z).
- [28] J. Toner and Y. Tu, “Long-range order in a two-dimensional dynamical XY model: How birds fly together,” *Phys. Rev. Lett.*, vol. 75, pp. 4326–4329, 23 Dec. 1995. DOI: [10.1103/PhysRevLett.75.4326](https://doi.org/10.1103/PhysRevLett.75.4326). [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevLett.75.4326>.
- [29] E. Bertin, M. Droz, and G. Grégoire, “Boltzmann and hydrodynamic description for self-propelled particles,” *Phys. Rev. E*, vol. 74, p. 022 101, 2 Aug. 2006. DOI: [10.1103/PhysRevE.74.022101](https://doi.org/10.1103/PhysRevE.74.022101). [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevE.74.022101>.
- [30] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, ser. Springer Series in Statistics. New York, NY, USA: Springer New York Inc., 2001.
- [31] R. Tibshirani, “Regression shrinkage and selection via the lasso,” *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.
- [32] A. E. Hoerl and R. W. Kennard, “Ridge regression: Biased estimation for nonorthogonal problems,” *Technometrics*, vol. 42, no. 1, pp. 80–86, 2000, ISSN: 00401706. [Online]. Available: <http://www.jstor.org/stable/1271436> (visited on 11/17/2022).

- [33] H. Zou and T. Hastie, “Regularization and variable selection via the elastic net,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 67, no. 2, pp. 301–320, 2005. DOI: <https://doi.org/10.1111/j.1467-9868.2005.00503.x>. eprint: <https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-9868.2005.00503.x>. [Online]. Available: <https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2005.00503.x>.
- [34] S. L. Brunton, J. L. Proctor, and J. N. Kutz, “Discovering governing equations from data by sparse identification of nonlinear dynamical systems,” *Proceedings of the National Academy of Sciences*, vol. 113, no. 15, pp. 3932–3937, 2016. DOI: [10.1073/pnas.1517384113](https://doi.org/10.1073/pnas.1517384113). eprint: <https://www.pnas.org/doi/pdf/10.1073/pnas.1517384113>. [Online]. Available: <https://www.pnas.org/doi/abs/10.1073/pnas.1517384113>.
- [35] R. Supekar, B. Song, A. Hastewell, G. P. T. Choi, A. Mietke, and J. Dunkel, *Learning hydrodynamic equations for active matter from particle simulations and experiments*, 2021. DOI: [10.48550/ARXIV.2101.06568](https://doi.org/10.48550/ARXIV.2101.06568). [Online]. Available: <https://arxiv.org/abs/2101.06568>.
- [36] L. Van Der Maaten, E. Postma, and J. Van den Herik, “Dimensionality reduction: A comparative review,” *J Mach Learn Res*, vol. 10, pp. 66–71, 2009.
- [37] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, 2nd ed. Academic Press, Oct. 1990.
- [38] C. Spearman, ““general intelligence,” objectively determined and measured,” *The American Journal of Psychology*, vol. 15, no. 2, pp. 201–292, 1904, ISSN: 00029556. [Online]. Available: <http://www.jstor.org/stable/1412107> (visited on 11/19/2022).
- [39] W. S. Torgerson, “Multidimensional scaling: I. theory and method,” *Psychometrika*, vol. 17, pp. 401–419, 1952.
- [40] Y. LIANG, H. LEE, S. LIM, W. LIN, K. LEE, and C. WU, “Proper orthogonal decomposition and its applications—part i: Theory,” *Journal of Sound and Vibration*, vol. 252, no. 3, pp. 527–544, 2002, ISSN: 0022-460X. DOI: <https://doi.org/10.1006/jsvi.2001.4041>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0022460X01940416>.
- [41] M. Mendez, M. Scelzo, and J.-M. Buchlin, “Multiscale modal analysis of an oscillating impinging gas jet,” *Experimental Thermal and Fluid Science*, vol. 91, pp. 256–276, 2018, ISSN: 0894-1777. DOI: <https://doi.org/10.1016/j.expthermflusci.2017.10.032>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0894177717303370>.
- [42] S. L. Brunton and B. R. Noack, “Closed-Loop Turbulence Control: Progress and Challenges,” *Applied Mechanics Reviews*, vol. 67, no. 5, Aug. 2015, 050801, ISSN: 0003-6900. DOI: [10.1115/1.4031175](https://doi.org/10.1115/1.4031175). [Online]. Available: <https://doi.org/10.1115/1.4031175>.

- [43] A. Deane, I. Kevrekidis, G. Karniadakis, and S. Orszag, “Low-dimensional models for complex geometry flows: Application to grooved channels and circular cylinders,” English (US), *Physics of fluids. A, Fluid dynamics*, vol. 3, no. 10, pp. 2337–2354, 1991, Copyright: Copyright 2017 Elsevier B.V., All rights reserved., ISSN: 0899-8213. DOI: [10.1063/1.857881](https://doi.org/10.1063/1.857881).
- [44] *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. 1996.
- [45] C. W. Rowley, T. Colonius, and R. M. Murray, “Model reduction for compressible flows using pod and galerkin projection,” *Physica D: Nonlinear Phenomena*, vol. 189, no. 1, pp. 115–129, 2004, ISSN: 0167-2789. DOI: <https://doi.org/10.1016/j.physd.2003.03.001>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167278903003841>.
- [46] M. A. Mendez, M. Balabane, and J.-M. Buchlin, “Multi-scale proper orthogonal decomposition of complex fluid flows,” *Journal of Fluid Mechanics*, vol. 870, pp. 988–1036, 2019. DOI: [10.1017/jfm.2019.212](https://doi.org/10.1017/jfm.2019.212).
- [47] M. Loeve, *Probability Theory I*, ser. Graduate Texts in Mathematics. Springer New York, 1977, ISBN: 9780387902104. [Online]. Available: <https://books.google.com.lb/books?id=n5vgjacnWDQC>.
- [48] N. Aubry, “On the hidden beauty of the proper orthogonal decomposition,” *Theoretical and Computational Fluid Dynamics*, vol. 2, no. 5-6, pp. 339–352, Aug. 1991. DOI: [10.1007/BF00271473](https://doi.org/10.1007/BF00271473).
- [49] C. Eckart and G. Young, “The approximation of one matrix by another of lower rank,” *Psychometrika*, vol. 1, no. 3, pp. 211–218, Sep. 1936. DOI: [10.1007/BF02288367](https://doi.org/10.1007/BF02288367). [Online]. Available: <https://ideas.repec.org/a/spr/psycho/v1y1936i3p211-218.html>.
- [50] F. Kwasniok, “Optimal galerkin approximations of partial differential equations using principal interaction patterns,” *Phys. Rev. E*, vol. 55, pp. 5365–5375, 5 May 1997. DOI: [10.1103/PhysRevE.55.5365](https://doi.org/10.1103/PhysRevE.55.5365). [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevE.55.5365>.
- [51] S. Mallat, *A wavelet tour of signal processing*. Elsevier, 1999.
- [52] P. Vaseghi, *Advanced Digital Signal Processing and Noise Reduction*. Dec. 2008, ISBN: 9780470740163. DOI: [10.1007/978-3-322-92773-6](https://doi.org/10.1007/978-3-322-92773-6).
- [53] J. O. Smith, *Spectral Audio Signal Processing*. <http://ccrma.stanford.edu/~jos/sasp/>, accessed [date], online book, 2011 edition.
- [54] K. J. Burns, G. M. Vasil, J. S. Oishi, D. Lecoanet, and B. P. Brown, “Dedalus: A flexible framework for numerical simulations with spectral methods,” *Physical Review Research*, vol. 2, no. 2, 023068, p. 023068, Apr. 2020. DOI: [10.1103/PhysRevResearch.2.023068](https://doi.org/10.1103/PhysRevResearch.2.023068). arXiv: [1905.10388](https://arxiv.org/abs/1905.10388) [astro-ph.IM].

- [55] Y. Bar-Sinai, S. Hoyer, J. Hickey, and M. P. Brenner, “Learning data-driven discretizations for partial differential equations,” *Proceedings of the National Academy of Sciences*, vol. 116, no. 31, pp. 15 344–15 349, 2019. DOI: [10.1073/pnas.1814058116](https://doi.org/10.1073/pnas.1814058116). eprint: <https://www.pnas.org/doi/pdf/10.1073/pnas.1814058116>. [Online]. Available: <https://www.pnas.org/doi/abs/10.1073/pnas.1814058116>.
- [56] S. L. Brunton, B. R. Noack, and P. Koumoutsakos, “Machine learning for fluid mechanics,” *Annual Review of Fluid Mechanics*, vol. 52, no. 1, pp. 477–508, 2020. DOI: [10.1146/annurev-fluid-010719-060214](https://doi.org/10.1146/annurev-fluid-010719-060214). [Online]. Available: <https://doi.org/10.1146/annurev-fluid-010719-060214>.
- [57] J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott, “Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach,” *Phys. Rev. Lett.*, vol. 120, p. 024 102, 2 Jan. 2018. DOI: [10.1103/PhysRevLett.120.024102](https://doi.org/10.1103/PhysRevLett.120.024102). [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevLett.120.024102>.