

DYNAMIC MODE DECOMPOSITION:
THEORY AND APPLICATIONS

JONATHAN H. TU

A DISSERTATION
PRESENTED TO THE FACULTY
OF PRINCETON UNIVERSITY
IN CANDIDACY FOR THE DEGREE
OF DOCTOR OF PHILOSOPHY

RECOMMENDED FOR ACCEPTANCE
BY THE DEPARTMENT OF
MECHANICAL AND AEROSPACE ENGINEERING
ADVISER: CLARENCE W. ROWLEY III

SEPTEMBER 2013

© Copyright by Jonathan H. Tu, 2013.

All rights reserved.

Abstract

Used to analyze the time-evolution of fluid flows, dynamic mode decomposition (DMD) has quickly gained traction in the fluids community. However, the existing DMD literature focuses primarily on applications, rather than theory. In this thesis, we present new results of both types.

First, we propose a new definition in which we interpret DMD as an approximate eigendecomposition of the best-fit (in a least-squares/minimum-norm sense) operator relating two data matrices. This definition preserves the link between DMD and Koopman operator theory; it also highlights the relationship between DMD and linear inverse modeling. Using our definition, we are able to generalize the DMD algorithm to arbitrary datasets, not just sequential time-series (as are typically considered). Then, turning to applications, we use DMD to estimate the slow eigenvectors that dominate the long-time behavior of impulse responses. We use these in developing a variant of balanced proper orthogonal decomposition that is both more accurate and more computationally efficient.

We also apply DMD to analyze oscillatory fluid flows, which is its most common use. In one example, we apply both DMD and proper orthogonal decomposition (POD) to study the effects of zero-net-mass-flux actuation on separated flows. We find a correlation between the separation bubble height and the distribution of energy among the POD modes. We also find that the most effective control strategy is characterized by frequency lock-on between the wake and the shear layer. In another example, we use DMD to investigate the source of low-frequency oscillations in shock-turbulent boundary layer interactions. Using data from direct numerical simulations, we find modes whose characteristics match those suggested by linear stability analysis.

The last part of this thesis deals with issues of time-resolution. DMD requires data that are collected at least twice as fast as any frequency of interest. We propose two approaches for identifying oscillatory flow structures when such sampling rates are not possible. First, we demonstrate a procedure for dynamically estimating a time-resolved trajectory from non-time-resolved data; DMD can be computed from the estimated trajectory. Second, we develop a method in which oscillatory modes are computed using compressed sensing techniques.

Acknowledgements

I was told that this would be the easy part of my thesis to write, but I am almost certain that it is in fact the hardest. How can I possibly even begin to express the debt of gratitude I owe to those who have helped me reach this point? The last five years have been a time of immense growth for me, both personally and professionally, and the list of those who have guided me along that journey seems nearly inexhaustable.

Certainly, I must thank my adviser, Clancy Rowley. He has been a great mentor to me, never letting me flounder due to a lack of direction, yet always giving me enough space to grow as an independent researcher. He has shown me that it really is possible to achieve combined success in research, teaching, and maintaining a healthy work/life balance. I can only hope that I might one day come close to replicating that success.

Of the many faculty whose help has made this thesis possible, I would like to specifically thank my thesis readers, examiners, and committee, who collectively comprise Phil Holmes, Lou Cattafesta, Lex Smits, Howard Stone, and Naomi Leonard. Their comments and guidance have been invaluable over the past few months, and really, years. I would also like to acknowledge my undergraduate adviser, Nathan Kutz. Without his advice, I would never have thought to apply to Princeton for graduate school. For that, and for his continued mentorship and support, I owe him many thanks. I must also thank Jill Ray and Jessica O'Leary, without whom I would have been unable to navigate a five-year long maze of paperwork and procedures.

Life at Princeton has been thoroughly enjoyable for me, and for that I must thank my friends, of whom there are too many to list. Some of you I seemingly brought with me from Seattle, and others I met here. Whether you were fellow Husky fans, lab mates, climbing buddies, tennis partners, all of the above, or none of the above, I couldn't have done this without your support and companionship. My time at Princeton would not have been complete if not for my various musical endeavors, and as such I must also thank Jo-Ann Sternberg, my clarinet teacher, and Ruth Ochs, my orchestra conductor, who have both become valued friends over the years. To my best friends, who know they are, I wouldn't be the person I am today without your unwavering love and confidence in me, and for that I am forever grateful.

Finally, and most importantly, I must thank my family for all that they have done. I am forever indebted to my parents, Harry Tu and Jennifer Ngo, for their many sacrifices over the years. I have never seen them spend a dollar on themselves that could have instead been spent to help me or my siblings. Their years of hard work and sacrifice can never be repaid, but I hope that someday I can return the favor to my own family. My brother Alexander Tu and my sister Katarina Tu have been both friends and foes over the years, but as we've grown up together, we've grown closer and closer. I am so proud of their many accomplishments, much more so than I am of my own. Words truly cannot express the depth of my love for and loyalty to my family. I dedicate this thesis to them.

The work presented in this thesis was supported by the Air Force Office of Scientific Research (grant FA9550-09-1-0257, under Dr. Doug Smith) and the National Science Foundation Graduate Research Fellowship Program.

This dissertation carries the number T-3268 in the records of the Department of Mechanical and Aerospace Engineering.

Contents

Abstract	iii
Acknowledgements	iv
List of Tables	ix
List of Figures	x
1 Introduction	1
1.1 Motivation	1
1.2 A brief history	2
1.3 Organization and contributions	4
1.4 Use of notation	6
2 Extensions of DMD theory	7
2.1 Motivation	7
2.2 A new definition	8
2.3 Algorithms	11
2.3.1 Sequential time-series	11
2.3.2 Non-sequential time-series	12
2.4 Examples	13
2.4.1 Stochastic dynamics	14
2.4.2 Standing waves	14
2.4.3 Non-uniform sampling	15
2.4.4 Combining multiple trajectories	17
2.5 Connections to Koopman operator theory	20
2.6 Equivalence to LIM	23
2.6.1 Nomenclature	23
2.6.2 Defining LIM	25
2.6.3 Proof of equivalence	26
2.7 Conclusions	27
3 A DMD-based extension of the BPOD algorithm	29
3.1 Motivation	30
3.2 Background	31
3.2.1 Empirical Gramians	31
3.2.2 Balanced truncation	32
3.2.3 BPOD	33
3.3 Analytic tail method	34
3.3.1 Motivation	34
3.3.2 Complex formulation	34

3.3.3	Real formulation	36
3.3.4	Application to BPOD	38
3.4	Snapshot-based eigenvector estimation using DMD	38
3.5	Results and discussion	39
3.5.1	Computing the controllability Gramian	39
3.5.2	Model reduction	43
3.6	Conclusions	49
4	Application to oscillatory fluid flows	51
4.1	A canonical separated flow	51
4.1.1	Motivation	52
4.1.2	Flow configuration and solver	53
4.1.3	Results	55
4.1.4	Summary	59
4.2	Shock-turbulent boundary layer interactions	60
4.2.1	Motivation	60
4.2.2	Flow configuration and solver	61
4.2.3	Preliminary results	62
4.2.4	Summary	64
4.3	Conclusions	65
5	Overcoming time resolution issues via dynamic estimation	67
5.1	Motivation	68
5.2	Background	70
5.2.1	Stochastic estimation	70
5.2.2	Dynamic estimation	71
5.3	Numerical methods	72
5.3.1	Modified stochastic estimation	72
5.3.2	Dynamic estimation	74
5.3.3	Dynamic estimator implementation	78
5.4	Experimental setup	79
5.4.1	Facility and instrumentation	79
5.4.2	Data acquisition	81
5.4.3	Data partitioning	82
5.5	Results and discussion	83
5.5.1	Wake characteristics	83
5.5.2	Low-dimensional flow-field estimation	87
5.5.3	Causal implementation	94
5.6	Conclusions	95
6	Towards a compressed DMD algorithm	96
6.1	Motivation	97
6.2	Numerical method	98
6.2.1	Scalar signals	98
6.2.2	Choice of basis, measurement	100
6.2.3	Vector-valued signals	100
6.2.4	Efficiency through POD projection	101
6.2.5	Sampling strategy	102

6.2.6	Summary of method	104
6.3	Results	104
6.3.1	Canonical dataset	105
6.3.2	Flow past a two-dimensional cylinder	107
6.4	Conclusions	112
7	Conclusions and future directions	113
7.1	Summary	113
7.2	Suggestions for future work	115
A	Background on POD	117
	Bibliography	119

List of Tables

2.1	Comparison of DMD eigenvalues computed using uniform/nonuniform sampling	18
3.1	BPOD costs for a 2-D cylinder flow	49
4.1	Separation bubble size for various ZNMF forcing frequencies	55

List of Figures

2.1	Dynamics of a noisy 1-D system	14
2.2	Domain for 2-D cylinder simulation	16
2.3	Inputs and outputs for control of a 2-D cylinder flow	17
2.4	Comparison of DMD eigenvalues computed using uniform/nonuniform sampling	18
2.5	Comparison of DMD modes computed using uniform/nonuniform sampling	19
2.6	Schematic of setup for bluff-body wake experiment	20
2.7	Typical PIV vorticity field	20
2.8	Comparison of DMD spectra computed using one/multiple experimental runs	21
2.9	Comparison of DMD modes computed using one/multiple experimental runs	21
3.1	Analytic tail method with a non-normal 3×3 linear system	40
3.2	Impulse response of a pseudorandom 100×100 system	42
3.3	Analytic tail method with a pseudorandom 100×100 system	42
3.4	Analytic tail method with the CGL equation: effect of simulation length	44
3.5	Analytic tail with the CGL equation: effect of model order	45
3.6	Unstable equilibrium of the flow past a 2-D cylinder	46
3.7	Unstable global modes of the flow past a 2-D cylinder	46
3.8	Inputs and outputs for control of a 2-D cylinder flow	47
3.9	Impulse response of a 2-D cylinder flow	47
3.10	Analytic tail method with a 2-D cylinder flow	48
4.1	Schematic of the canonical separated flow	54
4.2	Comparison of dominant POD modes	56
4.3	Comparison of secondary POD modes	57
4.4	POD energy content	57
4.5	Comparison of DMD spectra	58
4.6	Comparison of fundamental DMD modes	59
4.7	Comparison of secondary DMD modes	59
4.8	Schematic of domain used for DNS of STBLI	62
4.9	DMD spectrum computed from spanwise-averaged data	63
4.10	DMD mode computed from spanwise-averaged data	63
4.11	Comparison of ΔC_f profiles	64
4.12	DMD spectrum computed from 3-D data	65

4.13	Wall-normal plane from 3-D DMD mode	65
4.14	DMD mode computed from 3-D data	66
5.1	Flow chart of dynamic estimator implementation	79
5.2	Schematic of experimental setup	80
5.3	Cartoon of data acquisition method	82
5.4	Instantaneous PIV vorticity field	83
5.5	Energy content of the first r POD modes	84
5.6	POD modes computed from TRPIV fields	84
5.7	DMD spectrum computed from TRPIV velocity fields	85
5.8	DMD modes computed from TRPIV velocity fields	85
5.9	Measurement of v' taken in the wake at $x/h = 2.24$ and $y/h = 0.48$.	86
5.10	Autospectra of probe signals	87
5.11	Mean energy in MTD-mLSE estimation error	88
5.12	Mean and distribution of energy in estimation error	90
5.13	Time history of estimation error	91
5.14	Comparison of true and projected spanwise vorticity fields	92
5.15	Comparison of estimated spanwise vorticity fields	92
5.16	Estimation-based DMD modes	93
5.17	Mean and distribution of energy in estimation error for causal estimation	94
5.18	Mean and distribution of energy in estimation error for causal estimation using only point measurements	95
6.1	True oscillatory modes for canonical dataset	105
6.2	POD modes of canonical dataset	105
6.3	Random sampling for canonical dataset	106
6.4	Compressed sensing spectrum for canonical dataset	106
6.5	Compressed sensing modes for canonical dataset	107
6.6	DMD spectrum for 2-D cylinder flow	108
6.7	DMD modes for 2-D cylinder flow	108
6.8	POD energy for 2-D cylinder flow	109
6.9	First six POD modes for 2-D cylinder flow	110
6.10	DMD-like POD modes for 2-D cylinder flow	110
6.11	Random sampling for 2-D cylinder flow	111
6.12	OMP spectrum for 2-D cylinder flow	111
6.13	Dominant OMP mode for 2-D cylinder flow	111

Chapter 1

Introduction

Dynamic mode decomposition (DMD) is a method used to analyze the time evolution of fluid flows. First introduced in 2008, DMD has quickly gained a following. However, the majority of its practitioners have focused on applying the method, rather than on developing its underlying theory. In this thesis, we strive to do both. This chapter sets the stage for such a discussion, first motivating the need for methods like DMD, then providing a brief history of modal decomposition techniques in fluid mechanics, and finally, outlining the organization and key contributions of this thesis.

1.1 Motivation

Fluid mechanical devices abound in engineering. From pumps to compressors to airplanes to rockets, it is easy to think of devices that operate in fluid environments. In a sense, we are lucky, in that most fluid mechanical systems of interest are described by the same governing equations: the Navier–Stokes equations. Unfortunately, the Navier–Stokes equations are a set of nonlinear partial differential equations (PDEs) that give rise to all manner of dynamics, including those characterized by bifurcations, limit cycles, resonances, and full-blown turbulence. As a result, the complex geometries and challenging flow regimes (e.g., high speeds, pressures, or temperatures) typical in engineered systems can easily confound our ability to generate analytic solutions. This forces us to rely on experiments and high-performance computations when studying such systems.

As experiments and computations become more advanced, they generate ever-increasing amounts of data. (In 2011, the amount of data (of all types) generated worldwide was more than double the amount of available storage space [19, 60]!) While having more data is certainly better than having less, the sheer size of modern datasets poses its own challenges. Manipulating such data to find anything but the most obvious trends requires a skillset all its own. This has led to a growing need for *data-driven methods* that can take a dataset and characterize it in meaningful ways with minimal guidance. Along with the development of the methods themselves, there is a parallel need to develop foundational theory upon which the methods can rest. This is especially key in the data-driven approach, as we are relying on the data itself to guide us, rather than imposing our expert scientific knowledge.

Without sound theory, we may design algorithms that in some cases produce results that we cannot explain or trust.

Many in fluid mechanics have turned to *modal decomposition* as the tool of choice for data-driven analysis. Generally, a modal decomposition takes a set of data and from it computes a set of *modes*, or characteristic features. The meaning of the modes depends on the particular type of decomposition used. However, in all cases, the hope is that the modes identify features of the data that elucidate the underlying physics.

This thesis focuses in particular on DMD, which has gained popularity as a tool for analyzing the dynamics of nonlinearly evolving fluid flows. In fact, it is the only modal decomposition developed specifically for this task. Proper orthogonal decomposition (POD), perhaps the most common modal decomposition in the fluids community, is at its heart a statistical method designed to identify the features of a flow most important for reconstructing a dataset. It can certainly be applied to flows with nonlinear dynamics, but the POD modes are not necessarily optimal for modeling those dynamics. For instance, when analyzing a time-series, the POD modes remain unchanged if the data are reordered; the modes do not depend on the time evolution/dynamics encoded in the data. Balanced POD (BPOD) does identify modes that are dynamically important, but it is only applicable to flow control configurations, in which there are clearly defined inputs (actuators or disturbances) and outputs (sensor measurements). Even more restrictive, it is only applicable to systems where the input-output dynamics are linear.

In engineering flows, nonlinear dynamics may be unavoidable. Furthermore, as we continue to push for better performance and higher efficiency in engineered devices, an understanding of transient dynamics, not just steady-state behaviors, becomes critical. As such, it seems there is a clear role for DMD in engineering analysis going forward. In this thesis, we not only provide examples of DMD analysis, but also further develop DMD theory. We hope that by doing so, we give DMD practitioners a solid framework within which to work, demonstrating strategies for how to leverage DMD in different situations and establishing theory that makes it clear what DMD analysis can, and just as importantly cannot, tell us.

1.2 A brief history

To place DMD in the proper context, we must consider the recent history of modal decompositions in fluid mechanics. The most well-known of these is POD, which is attributed to Lumley [96, 97]. POD is a method that identifies the set of orthogonal modes that best reconstructs a dataset. That is, suppose we are given a dataset and asked what single mode best represents the entirety of that data. The solution is the first POD mode. If we are allowed two (orthogonal) modes to represent the data, then the optimal choice is the first two POD modes, and so on. (This notion of optimality is made more precise in Appendix A.) POD is known by many different names, depending on the field of application. Of these, principal component analysis (PCA) is the oldest, generally traced back more than a century to Pearson [123]. Other equivalent methods include the Hotelling transform [79, 80], the Karhunen–Loève transform [94], and empirical orthogonal function (EOF) analysis [95]. All of these reduce to computing the singular value decomposition (SVD) of a dataset.

The utility of POD was first demonstrated to the fluids community in the study of turbulent boundary layers, due mainly to the work of Lumley [96, 97], Sirovich [155], and Aubry et al. [12]. Since then, the number of fluids researchers using POD has exploded, with a quick web search for “POD and fluids” returning nearly 700 results. It has also spawned a number of variants, including double POD [153, 174], observer-inferred decomposition (OID) [140], and temporal POD [64], among others. (We note that though it has a similar name, BPOD is more of a related method than a variant, as its formulation is quite distinct from that of POD.) Due to the success of POD, the fluids community has been fairly receptive to the continued development and application of modal decomposition techniques.

As a result, when DMD was first introduced at a 2008 conference by Schmid and Sesterhenn [143], it was well-received and quickly embraced. The first archival article featuring DMD was published by Rowley et al. [138] only a year later, in 2009. Schmid [141] followed this with his own archival article in 2010. These are generally considered the two seminal works on the topic. However, they differ quite significantly in the way they regard DMD.

The article by Rowley et al. [138] builds on the theory of *Koopman spectral analysis*, which dates back to 1931 [89]. Mezić [103] was the first to apply this theory for the purposes of model reduction, with Rowley et al. [138] taking a similar approach in the context of fluid mechanics. Both leverage the fact that based on Koopman operator theory, there exists a set of modes, the “Koopman modes,” that completely characterizes the dynamics of a nonlinear system. Each mode is associated with a particular Koopman eigenvalue. Together, the modes and eigenvalues constitute a *Koopman decomposition*, which can be considered a generalization of an eigenvector/eigenvalue decomposition to nonlinear systems. Rowley et al. [138] regard DMD as a particular numerical algorithm that computes an approximate Koopman decomposition. Thus, from this perspective, DMD is not an end in itself, but a means to performing Koopman spectral analysis.

In contrast, the article by Schmid [141] focuses squarely on DMD. Again, the method is presented as a generalization of linear stability analysis, but here it is argued that DMD is valid for nonlinear systems because it analyzes a “linear tangent approximation to the underlying flow” [141]. This is in contrast to the Koopman operator framework discussed above. Schmid [141] also explores the connections between DMD and POD, leading to an SVD-based algorithm that is more well-conditioned than the companion matrix-based algorithm presented by Rowley et al. [138]. This algorithm has become the standard for DMD practitioners.

The differing perspectives found in these two articles has led to some confusion regarding the difference between Koopman modes and DMD modes. In fact, the two terms are often used interchangeably. At this point in time, both Koopman and DMD analysis rely on the same algorithms. It is only the theoretical framework in which the results are interpreted that differs.

Though it has been fewer than five years since its introduction, DMD has already been used to analyze a wide variety of flows. Rowley et al. [138] studied a jet in crossflow, showing that Koopman analysis correctly identifies the shedding frequency when linear stability analysis cannot. Furthermore, the Koopman modes decouple the shear-layer and wall vortices; POD modes mix them together. Schmid [141] applied DMD to a number of examples, including the flow over a square cavity, the wake behind a flexible membrane, and a jet between two cylinders, showing

that DMD is viable for both computational and experimental data. Others have applied DMD to the flow around high-speed trains [109], instabilities in annular liquid sheets [47], shock-turbulent boundary layer interactions [73], detonation waves [102], cavity flows [146], and various jets [142, 144, 145, 151].

There have also been a number of theoretical efforts. For instance, Duke et al. [48] analyzed the error properties of DMD when attempting to capture the growth rate of flow instabilities. Chen et al. [31] built on the work of Rowley et al. [138], offering existence and uniqueness theorems and showing further connections between DMD and Fourier analysis. That article also proposed a variant of DMD called “Optimized DMD,” which in some cases is better able to identify characteristic flow frequencies. Recently, Goulart et al. [65] introduced the “Optimal Mode Decomposition,” another closely related method. It involves a simultaneous optimization of a modal subspace and a “dynamic matrix,” whereas DMD can be interpreted as a similar optimization where the modal subspace is constrained to be spanned by POD modes. The review article by Mezić [104] offers a nice overview of the DMD literature.

1.3 Organization and contributions

The contributions of this thesis comprise two main parts. First, we develop a rigorous, linear algebra-based theory of DMD, building upon and extending previous work. In particular, we present a precise definition of DMD and show that it agrees with the algorithms developed by Rowley et al. [138] and Schmid [141]. From this definition, we build an entire DMD framework, which we use to explain seemingly anomolous results, develop extended algorithms, and elucidate connections between DMD and other topics. Second, we demonstrate strategies for implementing DMD in practice. Not only do we apply DMD to a number of fluid flows, but we also use DMD in conjunction with other methods, such as BPOD and dynamic estimation.

Chapter 2 deals with the theory of DMD. In Section 2.2, we present our new definition of DMD. Topics relating to the DMD algorithm are discussed in Section 2.3. Numerical examples demonstrating the utility of this generalized theory are presented in Section 2.4. In Section 2.5, the relationship between DMD and Koopman spectral analysis is discussed in detail, with care taken to identify cases where the two theories agree and cases where they do not. Then in Section 2.6, we show that under certain assumptions, DMD is equivalent to linear inverse modeling, a method from the climate science literature.

The remainder of this thesis centers on applications of DMD. Chapter 3 presents an extension of the BPOD algorithm that uses DMD to compute the slow eigenvectors and eigenvalues that dominate the tails of impulse responses. By doing so, we are able to simultaneously improve the speed and accuracy of BPOD computations. Chapter 4 focuses on DMD analysis of oscillatory fluid flows. In one example, we use DMD to look at the effect of open-loop, oscillatory forcing on the various frequencies that characterize a separated flow. We also apply DMD to identify coherent structures associated with oscillatory instabilities observed in shock-turbulent boundary layer interactions. Chapters 5 and 6 deal with time-resolution issues that are common when working with experimental data. In the former, we use dynamic estimation to estimate a time-resolved trajectory, applying DMD to the estimated

state history. In the latter we compute oscillatory modes directly from non-time-resolved data using compressed sensing techniques.

The work composing this thesis has resulted in the following archival publications:

- J. H. TU AND C. W. ROWLEY. An improved algorithm for balanced POD through an analytic treatment of impulse response tails. *J. Comput. Phys.*, **231**(16): 5317-5333, June 2012.
- J. H. TU, C. W. ROWLEY, J. GRIFFIN, L. N. CATTAFESTA III, A. HART, AND L. S. UKEILEY. Integration of non-time-resolved PIV and time-resolved velocity point sensors for dynamic estimation of time-resolved velocity fields. *Exp. Fluids*, **54**(2): 1429, Feb. 2013.
- J. H. TU, D. M. LUCHTENBURG, C. W. ROWLEY, S. L. BRUNTON, AND J. N. KUTZ. Generalizing dynamic mode decomposition to a larger class of datasets. *J. Comput. Dyn.* (in preparation).
- J. H. TU, C. W. ROWLEY, AND J. N. KUTZ. Spectral analysis of fluid flows using sub-Nyquist-rate PIV data. *Exp. Fluids* (in preparation).

It has also been presented at the following conferences:

- J. H. TU AND C. W. ROWLEY. Dynamic mode decomposition with sub-Nyquist-rate data samples. *SIAM Conference on Applications of Dynamics Systems*, May 2013.
- J. H. TU, C. W. ROWLEY. An improved algorithm for balanced proper orthogonal decomposition using analytic tails. *APS Division of Fluid Dynamics — 65th Annual Meeting*, Nov. 2012.
- J. H. TU, C. W. ROWLEY, J. GRIFFIN, L. N. CATTAFESTA III, A. HART, AND L. S. UKEILEY. Integration of non-time-resolved PIV and time-resolved velocity point sensors for dynamic estimation of time-resolved velocity fields. *50th AIAA Aerospace Sciences Meeting*, AIAA paper 2012-0033, Jan. 2012.
- J. H. TU, C. W. ROWLEY, S. ARAM, AND R. MITTAL. Identification of frequency lock-on using Koopman spectral analysis. *APS Division of Fluid Dynamics — 64th Annual Meeting*, Nov. 2011.
- J. H. TU, C. W. ROWLEY, E. ARAM, AND R. MITTAL. Koopman spectral analysis of separated flow over a finite-thickness flat plate with elliptical leading edge. *49th AIAA Aerospace Sciences Meeting*, AIAA paper 2011-0038, Jan. 2011.

Details on related work lying outside the scope of this thesis can be found here:

- K. K. CHEN, J. H. TU, AND C. W. ROWLEY. Variants of dynamic mode decomposition: connections between Koopman and Fourier analyses. *J. Non-linear Sci.*, **22**(6): 887–915, Dec. 2012.
- B. A. BELSON, J. H. TU, AND C. W. ROWLEY. A parallelized model reduction library. *ACM T. Math Software* (submitted).

1.4 Use of notation

This thesis brings together work from many fields, including fluid mechanics, control theory, stochastic estimation, and compressed sensing, to name a few. Rather than attempt to develop a single set of non-overlapping notation encompassing the combined set of variables used in this thesis, we have decided to adhere as closely as possible to the conventions of each individual field. This avoids the use of unfamiliar variables for familiar quantities, which could be a source of confusion. As such, certain variables may take on different meanings depending on the context. We are of course careful to define notation as it is introduced. However, it should be noted that between chapters, variable names may be reused. In other words, a particular variable name may not retain its meaning across chapters.

Chapter 2

Extensions of DMD theory

Theoretical results concerning dynamic mode decomposition (DMD) deal primarily with sequential time-series in which the measurement dimension is much larger than the number of measurements taken. In this chapter, we present a theoretical framework that generalizes DMD to a larger class of datasets. Using this framework, we develop extensions of the DMD algorithm to non-sequential time-series and demonstrate their utility through a number of examples. We also provide examples showing both the potential benefits and the potential pitfalls of applying DMD to rank-deficient datasets. (We more precisely define what we mean by the rank of a dataset in Section 2.1.) Such computations are not considered in the existing literature, but can be understood using our generalized framework. Though our theory is built entirely on linear algebra, it preserves the connections between DMD and Koopman operator theory. It also highlights the connections between DMD and linear inverse modeling (LIM), a method from the climate science community.

The remainder of this chapter is organized as follows: in Section 2.2, we present and discuss our new definition of DMD. In Section 2.3 we show that our definition agrees with the accepted, algorithmic definition of DMD and develop a generalized variant of the standard algorithm. A number of examples are presented in Section 2.3. These explore the application of DMD to rank-deficient datasets and non-sequential time-series. Sections 2.5 and 2.6 describe the connections between DMD and Koopman operator theory and between DMD and LIM, respectively. We summarize our results in Section 2.7.

The material presented in this chapter was developed in collaboration with Dirk M. Luchtenburg (Princeton University), Professor J. Nathan Kutz (University of Washington), and Steven L. Brunton (University of Washington). Dirk M. Luchtenburg provided valuable insight regarding practical implications of this theory. Professor J. Nathan Kutz provided the impetus for exploring the connections between DMD and LIM; the equivalence of the methods was proven with the help of Steven L. Brunton.

2.1 Motivation

The main theoretical results regarding DMD build on the theory of Krylov subspaces [31, 138, 141], due to the original derivation of DMD as a variant of the Arnoldi algorithm [138, 143]. As such, these results apply only to DMD computa-

tions wherein the underlying dataset comes from a sequential time-series. Furthermore, these results assume that when the last element of the time-series is ignored, the elements of the time-series are linearly independent; we say that such a dataset (comprising all but the last element of the time-series) has full rank. (Some results also require that the DMD eigenvalues are distinct.) In contrast, the SVD-based DMD algorithm [141] produces unique modes and eigenvalues even when these conditions are not satisfied. Given that this algorithm has become standard among DMD practitioners, a complete theory of DMD should account for the validity (or lack thereof) of such computations.

To develop such a theory, we build on the idea that DMD is able to characterize nonlinear dynamics through an analysis of some approximating linear system, first suggested by Schmid [141]. Consider two data matrices \mathbf{X} and \mathbf{X}' with arbitrary but matching sizes. Solving $\mathbf{A}\mathbf{X} = \mathbf{X}'$ for \mathbf{A} using a right pseudoinverse yields the best-fit linear operator relating \mathbf{X} and \mathbf{X}' , in a least-squares/minimum-norm sense. We show that DMD can be defined as an approximate eigendecomposition of this approximating linear operator. (Note that this operator is not defined abstractly, but rather in terms of the data matrices \mathbf{X} and \mathbf{X}' .)

There is of course no guarantee that analyzing this particular approximation is meaningful when \mathbf{X} and \mathbf{X}' are generated by nonlinear dynamics. To this end, we show that when \mathbf{X} has linearly independent columns, our definition preserves the connections between DMD and Koopman operator theory. This is key, as it allows us to interpret DMD as an approximation to Koopman spectral analysis. We can then be confident that DMD is useful for characterizing nonlinear dynamics. When \mathbf{X} is rank-deficient, the Koopman analogy can break down and DMD analysis may produce either meaningful or misleading results. In Section 2.4 we show an example of each and explain the results using our approximating-operator definition of DMD. (For a more detailed investigation of how well DMD eigenvalues approximate Koopman eigenvalues, we refer the reader to [14].)

By defining DMD in this way, we emphasize the nature of DMD analysis, rather than its implementation. This allows us to consider alternative DMD algorithms. For instance, it leads to natural extensions of DMD that deal with concatenated datasets and non-uniformly sampled data. It also highlights connections between DMD and linear inverse modeling (LIM), a method developed in the climate science community decades ago [124, 125]. In fact, under certain conditions, DMD is equivalent to LIM. As such, it stands to reason that practitioners of DMD could benefit greatly from an awareness of the climate science/LIM literature.

2.2 A new definition

DMD is typically applied to data matrices \mathbf{X} and \mathbf{X}' whose columns are taken from a sequential time-series $\{\mathbf{x}_k\}_{k=0}^m$. In this section, we relax this restriction on the data matrices, developing a definition of DMD that applies to generic \mathbf{X} and \mathbf{X}' . We do so by defining DMD as an approximate eigendecomposition of the operator $\mathbf{A} \triangleq \mathbf{X}'\mathbf{X}^\dagger$, where \mathbf{X}^\dagger is the Moore-Penrose generalized inverse of \mathbf{X} . This operator is the least-squares/minimum-norm solution to the potentially over- or under-constrained problem $\mathbf{A}\mathbf{X} = \mathbf{X}'$, and as such we can consider \mathbf{A} to be the best-fit operator relating \mathbf{X} and \mathbf{X}' .

Definition 1. Consider $\mathbf{X}, \mathbf{X}' \in \mathbb{R}^{n \times m}$. Define the operator \mathbf{A} as

$$\mathbf{A} \triangleq \mathbf{X}'\mathbf{X}^\dagger,$$

where \mathbf{X}^\dagger indicates the Moore-Penrose pseudoinverse of \mathbf{X} . Let the SVD of \mathbf{X} be given by

$$\mathbf{X} = [\Phi \ \cdots] \begin{bmatrix} \Sigma & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{W}^* \\ \vdots \end{bmatrix}, \quad (2.1)$$

with $\Phi \in \mathbb{R}^{n \times q}$, $\Sigma \in \mathbb{R}^{q \times q}$, and $\mathbf{W} \in \mathbb{R}^{m \times q}$. q is the rank of \mathbf{X} , Σ is diagonal, and \mathbf{W}^* denotes the conjugate transpose of \mathbf{W} . Then we can write

$$\mathbf{A} = \mathbf{X}'\mathbf{W}\Sigma^{-1}\Phi^*, \quad (2.2)$$

with its projection onto the column space of Φ^* given by

$$\tilde{\mathbf{A}} \triangleq \Phi^* \mathbf{A} \Phi. \quad (2.3)$$

Denoting the eigendecomposition of $\tilde{\mathbf{A}}$ by

$$\tilde{\mathbf{A}}\tilde{\mathbf{V}} = \tilde{\mathbf{V}}\Lambda, \quad (2.4)$$

the DMD eigenvalues are given by the eigenvalues lying on the diagonal of Λ (a diagonal matrix), and the corresponding DMD modes by the columns of

$$\Psi \triangleq \Phi\tilde{\mathbf{V}}. \quad (2.5)$$

Together, the pair (Ψ, Λ) composes the dynamic mode decomposition of the data $(\mathbf{X}, \mathbf{X}')$.

Remark 1. In (2.1), the columns of Φ are the POD modes of the matrix \mathbf{X} . As such, (2.3) defines $\tilde{\mathbf{A}}$ to be the POD projection of \mathbf{A} and (2.5) defines the DMD modes to be a linear combination of the POD modes.

From this definition, it is clear that Ψ and Λ are related to the operator \mathbf{A} . To elucidate the nature of this relationship, we make use of the following property:

Theorem 1. If the columns of \mathbf{X}' are spanned by those of \mathbf{X} , the DMD modes and eigenvalues defined in Definition 1 are eigenvectors and eigenvalues of the operator \mathbf{A} defined by (2.2).

Proof. Substituting (2.3) into (2.4), we see that the DMD modes and eigenvalues satisfy

$$\begin{aligned} \Phi^* \mathbf{A} \Phi \tilde{\mathbf{V}} &= \tilde{\mathbf{V}} \Lambda \\ \Phi \Phi^* \mathbf{A} \Phi \tilde{\mathbf{V}} &= \Phi \tilde{\mathbf{V}} \Lambda \\ \Phi \Phi^* \mathbf{A} \Psi &= \Psi \Lambda. \end{aligned} \quad (2.6)$$

(We note that $\Phi^* \Phi = \mathbf{W}^* \mathbf{W} = \mathbf{I}$, but in general $\Phi \Phi^*$ and $\mathbf{W} \mathbf{W}^*$ are not equal to the identity matrix.) As such, they are eigenvectors and eigenvalues of the projection of \mathbf{A} onto the column space of Φ . Because the columns of Φ are orthonormal, we

can write

$$\mathbf{X}' = \Phi\Phi^*\mathbf{X}' + \mathbf{R}',$$

where $\Phi^*\mathbf{R}' = \mathbf{0}$. Combining this expression with (2.2), we find that \mathbf{A} can be written as

$$\begin{aligned} \mathbf{A} &= (\Phi\Phi^*\mathbf{X}' + \mathbf{R}')\mathbf{W}\Sigma^{-1}\Phi^* \\ &= \Phi\Phi^*\mathbf{A} + \mathbf{R}'\mathbf{W}\Sigma^{-1}\Phi^*. \end{aligned} \quad (2.7)$$

Plugging (2.7) into (2.6) and making use of (2.5), along with the fact that $\Phi^*\Phi$ is the identity, we find that

$$\begin{aligned} (\mathbf{A} - \mathbf{R}'\mathbf{W}\Sigma^{-1}\Phi^*)\Psi &= \Psi\Lambda \\ \mathbf{A}\Psi - \Psi\Lambda &= \mathbf{R}'\mathbf{W}\Sigma^{-1}\tilde{\mathbf{V}}. \end{aligned} \quad (2.8)$$

When the columns of \mathbf{X}' are spanned by those of \mathbf{X} (or equivalently, those of Φ), we have $\mathbf{R}' = \mathbf{0}$ and the DMD modes and eigenvalues are eigenvectors and eigenvalues of \mathbf{A} . \square

Remark 2. *We note that when \mathbf{R}' is not identically zero, there is no guarantee that the pair (Ψ, Λ) closely approximates the true eigendecomposition of \mathbf{A} . However, we find that in practice, \mathbf{R}' being small is generally a good metric for the convergence of the DMD modes and eigenvalues to the correct values. Arguments similar to the one used to prove Theorem 1 are used to prove the convergence of the Arnoldi algorithm [139], which is a standard algorithm for iteratively computing approximate eigendecompositions of a linear operator. In fact, when applied to sequential time-series, DMD can be considered a variant of the Arnoldi algorithm [138, 141]. As such, we consider DMD to be a method for computing an approximate eigendecomposition of \mathbf{A} .*

Given Definition 1 and Theorem 1, we see that the defining property of DMD is that it yields an approximate eigendecomposition of the best-fit linear operator relating two data matrices. Consequently, *any* algorithm that produces modes and eigenvalues satisfying Definition 1 (and thus Theorem 1) should be considered an implementation of DMD. While one could compute a DMD using the equations given in Definition 1, we emphasize that this is *not* how DMD should be computed in practice. For instance, if the size of the data vectors n is very large, then it may be inefficient, or even impossible, to form the matrix \mathbf{A} . We discuss practical DMD algorithms in Section 2.3.

As DMD theory continues to develop, it may be necessary to consider more general definitions of DMD. A number of possible generalizations arise naturally from modifications of Definition 1. Suppose we continue to define DMD as an approximate eigendecomposition of \mathbf{A} , as given by (2.2). We can generalize Definition 1 by altering the nature of the approximation: rather than defining $\hat{\mathbf{A}}$ through (2.3), we can allow for more general projections of \mathbf{A} . For instance, we can truncate the SVD of \mathbf{X} , keeping only the first r columns of Φ and truncating Σ and \mathbf{W} accordingly. (This was first suggested in [141].) The error in the eigendecomposition is still given by (2.8), but with \mathbf{R}' computed with respect to the truncated SVD basis. This approach might be appropriate if the data \mathbf{X} and \mathbf{X}' are known to evolve in a

relatively low-dimensional space, with excursions in other directions due primarily to noise or unimportant dynamics. Other, more general projections could also be used, though if DMD is to remain a data-driven method, such projections should be constrained to bases lying in the column space of \mathbf{X} .

2.3 Algorithms

Here we discuss how to compute DMD modes and eigenvalues that satisfy the definition presented in Section 2.2. Specifically, we show that the standard, SVD-based DMD algorithm agrees with Definition 1. We also present a variant of this algorithm that extends DMD to non-sequential time-series. To do so, we take advantage of the fact that Definition 1 allows for arbitrary data matrices \mathbf{X} and \mathbf{X}' .

2.3.1 Sequential time-series

There have been a number of algorithms proposed for computing DMD, which differ mainly in their use of a companion matrix [138] versus an SVD [141]. (The two approaches are equivalent for full-rank datasets [141].) All are presented in the context of analyzing sequential time-series. The SVD-based algorithm has come to dominate among DMD practitioners due to its numerical stability. Effectively, it has become the working definition of DMD.

In the standard SVD-based algorithm [141], the data matrices \mathbf{X} and \mathbf{X}' are defined as

$$\mathbf{X} \triangleq \begin{bmatrix} | & & | \\ \mathbf{x}_0 & \cdots & \mathbf{x}_{m-1} \\ | & & | \end{bmatrix}, \quad \mathbf{X}' \triangleq \begin{bmatrix} | & & | \\ \mathbf{x}_1 & \cdots & \mathbf{x}_m \\ | & & | \end{bmatrix}. \quad (2.9)$$

It is assumed that the data come from a dynamical system whose evolution is given by

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k),$$

where each $\mathbf{x}_k \in \mathbb{R}^n$. (Alternatively, the vectors \mathbf{x}_k can be sampled from a continuous evolution $\mathbf{x}(t)$, in which case $\mathbf{x}_k = \mathbf{x}(k\Delta t)$ and a fixed sampling rate Δt is assumed.) Thus \mathbf{X} and \mathbf{X}' describe the sequential time-series $\{\mathbf{x}_k\}_{k=0}^m$.

The rest of the algorithm follows Definition 1, except that $\tilde{\mathbf{A}}$ is computed directly, rather than as a projection of \mathbf{A} . Doing so relies on the following identity, which can be derived by plugging (2.2) into (2.3) and recalling that $\Phi^*\Phi$ is the identity:

$$\begin{aligned} \tilde{\mathbf{A}} &= \Phi^* \mathbf{A} \Phi \\ &= \Phi^* (\mathbf{X}' \mathbf{W} \Sigma^{-1} \Phi^*) \Phi \\ &= \Phi^* \mathbf{X}' \mathbf{W} \Sigma^{-1}. \end{aligned} \quad (2.10)$$

Thus DMD modes and eigenvalues computed using the standard SVD-based algorithm satisfy Definition 1.

In fluid systems, the state dimension is generally much larger than the number of snapshots (i.e., $n \gg m$). This can be due to the fine spatial resolution necessary to accurately simulate the Navier–Stokes equations, or the fine image resolution necessary to resolve pertinent flow features in experiments. As such, the use of (2.10) is key, as it bypasses the computation of the $n \times n$ operator \mathbf{A} . (It is not uncommon

for n to be 10^6 or greater in fluid systems, making the storage of an $n \times n$ matrix computationally prohibitive.) Further efficiency can be achieved by computing the SVD of \mathbf{X} using the so-called “method of snapshots,” often used to compute POD modes [155]. To do so, we first solve the symmetric $m \times m$ eigenvalue problem

$$\mathbf{X}^* \mathbf{X} \mathbf{W} = \mathbf{W} \mathbf{\Sigma}^2.$$

By definition, the columns of \mathbf{W} and diagonal elements of $\mathbf{\Sigma}$ are right singular vectors and values of \mathbf{X} , respectively. We then define

$$\mathbf{\Phi} \triangleq \mathbf{X} \mathbf{W} \mathbf{\Sigma}^{-1}. \quad (2.11)$$

It is easy to show that $\mathbf{\Phi}$ satisfies $\mathbf{X} \mathbf{X}^* \mathbf{\Phi} = \mathbf{\Phi} \mathbf{\Sigma}^2$, and as such, contains left singular vectors of \mathbf{X} . (Other efficiency-driven modifications to the standard SVD-based algorithm are detailed in [171] and [21].)

The original presentation of the SVD-based algorithm in [141] discusses the meaning of the DMD modes and eigenvalues in terms of a linear operator satisfying $\mathbf{A} \mathbf{X} = \mathbf{X}'$. Such an operator would exist if the trajectory $\{\mathbf{x}_k\}_{k=0}^m$ were generated by linear dynamics $\mathbf{x}_{k+1} = \mathbf{A} \mathbf{x}_k$. It would also exist if the trajectory were nonlinear but could be approximated locally by a linear operator. Using this operator, one can show that (2.3) and (2.10) agree; in [141], this agreement is used to show the connection between DMD and POD. However, the assumption that such an operator exists is unnecessary: Remark 1 establishes the same connection without it. Similarly, the equivalence of (2.3) and (2.10) was shown above using only Definition 1. Thus we are able to prove the relationship between DMD and POD more generally, eliminating the need to assume linear dynamics and extending the result from sequential time-series to generic data matrices \mathbf{X} and \mathbf{X}' .

Another key result in DMD theory concerns sequential time-series $\{\mathbf{x}_k\}_{k=0}^m$ for which $\mathbf{x}_{k+1} = \mathbf{A} \mathbf{x}_k$. Suppose the last snapshot \mathbf{x}_m is linearly dependent on the first m snapshots. Then the DMD eigenvalues are equal to those of a companion matrix describing the snapshot evolution. This is proven in both [138] and [141] and shows that when applied to sequential time-series, DMD can be considered a variant of the Arnoldi algorithm. However, this result is a special case of Theorem 1. In [138] and [141], the first $m - 1$ columns of \mathbf{X}' are also columns of \mathbf{X} and the last column is \mathbf{x}_m , so any non-zero entries of \mathbf{R}' must lie in its last column. Thus the condition that \mathbf{x}_m be linearly dependent on the first m snapshots is equivalent to $\mathbf{R}' = \mathbf{0}$. As such, we see that Theorem 1 generalizes this result from sequential time-series to arbitrary data matrices \mathbf{X} and \mathbf{X}' . Furthermore, it does so without assuming a linear relationship between the snapshots.

2.3.2 Non-sequential time-series

We observe that the standard SVD-based DMD algorithm, described in Section 2.3.1, does not make use of the structure imposed by (2.9). That is, the algorithm can be carried out for arbitrary \mathbf{X} and \mathbf{X}' . We recall from Definition 1 that DMD can be interpreted as an analysis of the best-fit operator relating \mathbf{X} and \mathbf{X}' . This operator relates the columns of \mathbf{X} to those of \mathbf{X}' in a pairwise fashion. As such, if we wish to use DMD to analyze the dynamics of a generic (not necessarily

sequential) time-series, we simply require that the columns of \mathbf{X} and \mathbf{X}' be related by the dynamics of interest.

As before, consider a dynamical system governed by

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k),$$

with each $\mathbf{x}_k \in \mathbb{R}^n$. We can use DMD to analyze these dynamics by collecting a sequence of m vectors $\{\mathbf{x}_{k_0}, \mathbf{x}_{k_1}, \dots, \mathbf{x}_{k_{m-1}}\}$ and a corresponding set $\{\mathbf{f}(\mathbf{x}_{k_0}) = \mathbf{x}_{k_0+1}, \mathbf{f}(\mathbf{x}_{k_1}) = \mathbf{x}_{k_1+1}, \dots, \mathbf{f}(\mathbf{x}_{k_{m-1}}) = \mathbf{x}_{k_{m-1}+1}\}$. Unlike in Section 2.3.1, the index set $\{k_0, k_1, \dots, k_{m-1}\}$ does not need to be sequential or ordered. Defining the data matrices

$$\mathbf{X} \triangleq \begin{bmatrix} | & & | \\ \mathbf{x}_{k_0} & \cdots & \mathbf{x}_{k_{m-1}} \\ | & & | \end{bmatrix}, \quad \mathbf{X}' \triangleq \begin{bmatrix} | & & | \\ \mathbf{x}_{k_0+1} & \cdots & \mathbf{x}_{k_{m-1}+1} \\ | & & | \end{bmatrix}, \quad (2.12)$$

the DMD modes and eigenvalues are computed as described in Section 2.3.1.

Remark 3. *We observe that only the pairwise correspondence of the columns of \mathbf{X} and \mathbf{X}' is important, and not the overall ordering. That is, permuting the order of the columns of \mathbf{X} has no effect on the subsequent computation of DMD modes and eigenvalues, so long as the same permutation is applied to the columns of \mathbf{X}' . This is true even for data taken from a sequential time-series (i.e., \mathbf{X} and \mathbf{X}' satisfying (2.9)).*

By using (2.12) in place of (2.9), we generalize the algorithm presented in [141] to a larger class of datasets: the data need not be sequential or uniformly sampled in time, \mathbf{X} and \mathbf{X}' can have distinct columns, and there is no requirement of linear independence in the columns of \mathbf{X} . The computation of DMD from a sequential time-series can be considered a special case of this generalized approach. However, it is important to note that for sequential time-series, there exist memory-efficient variants of the standard SVD-based DMD algorithm [21, 171]. These improved algorithms take advantage of the overlap in the columns of \mathbf{X} and \mathbf{X}' (when defined as in (2.9)) to avoid redundant computations. This is not possible for the general definitions of \mathbf{X} and \mathbf{X}' given by (2.12).

2.4 Examples

In this section we present examples that demonstrate the utility of a DMD theory based on Definition 1. The first two examples consider DMD computations involving rank-deficient datasets, which are not treated in the existing DMD literature. We show that in some cases, DMD can still provide meaningful information about the underlying dynamical system, but in others, the results can be misleading. The second two examples use the generalized approach described in Section 2.3.2 to perform DMD analysis using non-sequential datasets. First, we use non-uniform sampling to dramatically increase the efficiency of DMD computations. Then, we concatenate time-series taken from multiple runs of an experiment, reducing the effects of noise.

2.4.1 Stochastic dynamics

Consider a system with stochastic dynamics

$$x_{k+1} = \lambda x_k + n_k, \quad (2.13)$$

where each $x_k \in \mathbb{R}$. We choose a decay rate $\lambda = 0.5$ and let n_k be white noise with variance $\sigma^2 = 10$. (This system was first used as a test of DMD in [186].) Figure 2.1 (left) shows a typical trajectory for an initial condition $x_0 = 0$. If we apply DMD to this trajectory, we estimate a decay rate $\tilde{\lambda} = 0.55$. This is despite the fact that the nominal (noiseless) trajectory is simply given by $x_k = 0$ for all k ; a global, linear analysis of the trajectory shown in Figure 2.1 (left) would identify a stationary process ($\tilde{\lambda} = 0$).

Because the existing DMD literature focuses on high-dimensional systems, existing DMD theory deals primarily with time-series whose elements are linearly independent. As such, it cannot be applied to explain the ability of DMD to accurately estimate the dynamics underlying this noisy data (a rank-one time-series). Recalling Definition 1, we can interpret DMD in terms of a linear operator that relates the columns of a data matrix \mathbf{X} to those of \mathbf{X}' , in column-wise pairs. Figure 2.1 (right) shows the time-series from Figure 2.1 (left) plotted in this pairwise fashion. We see that though the data are noisy, there is clear evidence of a linear relationship between x_k and x_{k+1} . For rank-deficient data, DMD approximates the dynamics relating \mathbf{X} and \mathbf{X}' through a least-squares fit, and so it is no surprise that we can accurately estimate λ from this time-series.

2.4.2 Standing waves

Because each DMD mode has a corresponding DMD eigenvalue (and thus a corresponding growth rate and, in the case of a complex eigenvalue, frequency), DMD is often used to analyze oscillatory behavior, whether the underlying dynamics are linear or nonlinear. Consider data generated by a standing wave:

$$\mathbf{x}_k = \sin(\omega t_k) \mathbf{v}, \quad (2.14)$$

where \mathbf{v} is a fixed vector in \mathbb{R}^n .

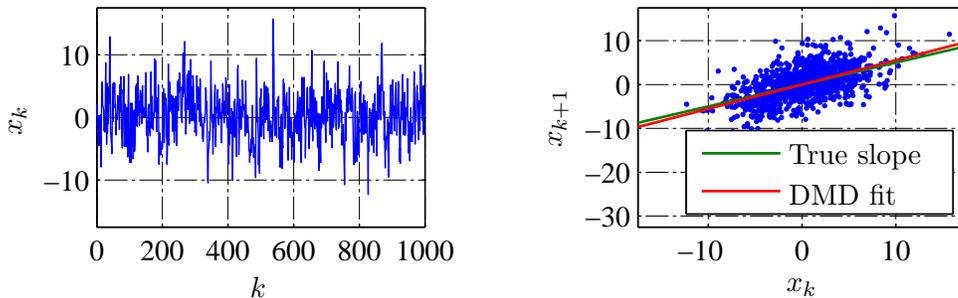


Figure 2.1: (Left) Typical trajectory of a noisy one-dimensional system governed by (2.13). (Right) Scatter plot showing the correlation of x_{k+1} with x_k . DMD is able to identify the relationship between future and past values of x even though the dataset is rank-deficient.

Such data can arise from a linear system. For instance, consider dynamics

$$\begin{aligned}\dot{x} &= -\omega y \\ \dot{y} &= \omega x.\end{aligned}$$

If we measure only the state x , then we observe a standing wave. (Since our measurement is one-dimensional, we simply have $v = 1$.) Such behavior can also arise in nonlinear systems, for instance by measuring only one component of a multi-dimensional limit cycle.

Suppose we compute DMD from data satisfying (2.14). By construction, the columns of the data matrix \mathbf{X} will be spanned by the single basis vector \mathbf{v} . As such, the SVD of \mathbf{X} will generate a matrix Φ with a single column, and the matrix $\tilde{\mathbf{A}}$ will be 1×1 . Then there will be precisely one DMD eigenvalue λ . Since we assume \mathbf{x} is real-valued, then so is λ , meaning it captures only exponential growth/decay, and no oscillations. This is despite the fact that the original data are known to oscillate with a fixed frequency.

The inability to capture standing waves is a general property of DMD. DMD eigenvalues and modes are determined by the eigenvalue problem (2.4). If the data matrix \mathbf{X} is real-valued, then any complex eigenvalues of $\tilde{\mathbf{A}}$ must come in conjugate pairs, with corresponding complex conjugate eigenvectors. Consequently, any oscillation with a frequency ω must correspond to a subspace whose dimension is at least two. For this reason, DMD cannot correctly capture standing wave behavior, which oscillates in a subspace of rank one. More generally, DMD cannot capture oscillatory behavior occurring in a subspace whose rank is less than twice the number of observed frequencies.

We note that in practice, we observe the same deficiency when the data don't satisfy (2.14) exactly, so long as the dynamics are dominated by such behavior (a standing wave). Thus the presence of random noise, which may increase the rank of the dataset, does not alleviate the problem. This is not surprising, as the addition of random noise should not enlarge the subspace in which the oscillation occurs. However, if we append the measurement with a time-shifted value, i.e., performing DMD on a sequence of vectors $[\mathbf{x}_k \ \mathbf{x}_{k+1}]^T$, then we are often able to identify the correct oscillation frequency.

2.4.3 Non-uniform sampling

Systems with a wide range of time scales can be challenging to analyze. If data are collected too slowly, dynamics on the fastest time scales will not be captured. On the other hand, uniform sampling at a high frequency can yield an overabundance of data, which can prove challenging to deal with numerically. Such a situation can be handled using the following sampling strategy:

$$\mathbf{X} \triangleq \begin{bmatrix} | & | & & | \\ \mathbf{x}_0 & \mathbf{x}_P & \cdots & \mathbf{x}_{(m-1)P} \\ | & | & & | \end{bmatrix}, \quad \mathbf{X}' \triangleq \begin{bmatrix} | & | & & | \\ \mathbf{x}_1 & \mathbf{x}_{P+1} & \cdots & \mathbf{x}_{(m-1)P+1} \\ | & | & & | \end{bmatrix}, \quad (2.15)$$

where we again assume dynamics of the form $\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k)$. The columns of \mathbf{X} and \mathbf{X}' are separated by a single iteration of \mathbf{f} , capturing its fastest dynamics. However,

the tuning parameter P allows a separation of time scales between the flow map iteration and the rate of data collection.

We demonstrate this strategy using a flow control example. (The flow configuration described below is the same one used in the last example in Section 3.5.) Consider the flow past a two-dimensional cylinder, which is governed by the incompressible Navier–Stokes equations. We simulate the dynamics using the fast immersed boundary projection method detailed in [36, 158]. The (non-dimensionalized) equations of motion are

$$\begin{aligned} \frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla) \vec{u} &= -\nabla p + \frac{1}{\text{Re}} \nabla^2 \vec{u} + \int_{\partial \mathcal{B}} \vec{f}(\vec{x}) \delta(\vec{x} - \vec{\xi}) d\vec{\xi} \\ \nabla \cdot \vec{u} &= 0, \end{aligned} \quad (2.16)$$

where \vec{u} is the velocity, p is the pressure, and \vec{x} is the spatial coordinate. The Reynolds number $\text{Re} \triangleq U_\infty D / \nu$ is a nondimensional parameter defined by the freestream velocity U_∞ , the cylinder diameter D , and the kinematic viscosity ν . $\partial \mathcal{B}$ is the union of the boundaries of any bodies in the flow. \vec{f} is a boundary force that can be thought of a Lagrange multiplier used to enforce the no-slip boundary condition. δ is the Dirac delta function. (We use arrows to denote the vectors in these equations, to avoid confusion with the variables defined previously.)

The fast immersed boundary method uses nested domains, each with increasing mesh resolution. For the finest domain we consider $\vec{x} \in [-15, 15] \times [-5, 5]$, with a cylinder of diameter $D = 1$ centered at $(0, 0)$. The large upstream region is useful for the adjoint simulations, for which the flow moves in the reverse direction. With three nested grids, the full computational domain spans a region $(x, y) \in [-60, 60] \times [-20, 20]$. (See Figure 2.2 for an illustration of the computational domain.) Convergence tests show that this domain is sufficiently large, avoiding blockage effects and fully capturing the features of the wake. In terms of grid cells, each of the nested domains has dimension 1500×500 , corresponding to $dx = dy = 0.02$ for the innermost domain. Only the data from the innermost

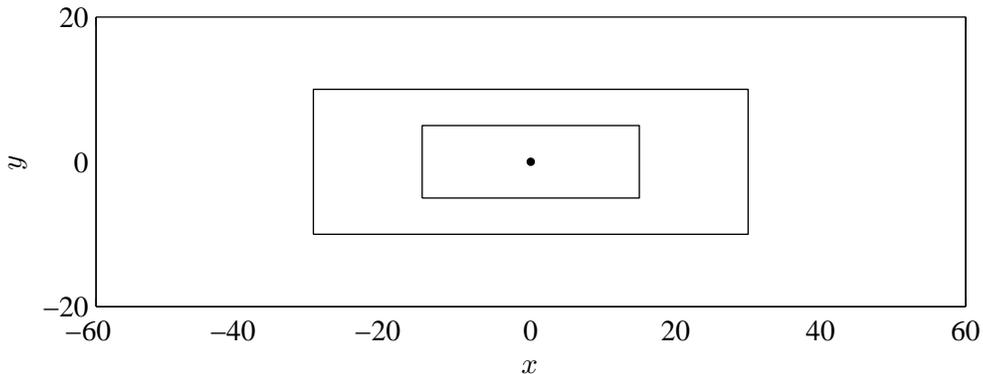


Figure 2.2: Domain used for simulating the flow past a two-dimensional cylinder. Each of the nested domains contains 1500×500 grid points, giving the finest grid a grid spacing $dx = dy = 0.02$. The large upstream region is useful for adjoint simulations, which flow from right to left.

domain, with the finest resolution, are used for DMD analysis. The outer domains are used only to ensure an accurate simulation.

We consider a uniform incoming flow with freestream velocity $U_\infty = 1$ and Reynolds number $Re = 100$. At this Reynolds number, the flow is globally unstable. Motivated by an application of BPOD, we restrict the linearized dynamics to their stable subspace. The system is actuated using a vertical velocity perturbation supported on a disk downstream of the cylinder and sensed using localized measurements of vertical velocity placed along the flow centerline. This setup is based on flow control benchmark proposed in [116] and is illustrated in Figure 2.3 (left).

The impulse response of this system is shown in Figure 2.3 (right). We see that from $t = 200$ to $t = 500$, the dynamics exhibit both a fast and slow oscillation. Suppose we want to identify the underlying frequencies and corresponding modes using DMD. In order to capture the fast frequency, we must sample the system every 50 timesteps, with each timestep corresponding to $\Delta t = 0.02$. (This is in order to satisfy the Nyquist-Shannon sampling criterion.) As such, we let $\mathbf{x}_k = x(50k\Delta t)$.

Table 2.1 and Figure 2.4 compare the DMD eigenvalues computed using uniform sampling and non-uniform sampling. (Referring back to (2.15), the former corresponds to $P = 1$ and the latter to $P = 10$.) We see that the dominant eigenvalues agree, with less than 10 % error in all cases. (We use the DMD eigenvalues computed with uniform sampling as truth values.) However, the larger errors occur for modes with norms on the order of 10^{-5} , two orders of magnitude smaller than those of the dominant two DMD modes. As such, these modes have negligible contribution to the evolution of the impulse response, and thus the error in the corresponding eigenvalues is not significant. The dominant DMD modes show similar agreement, as seen in Figure 2.5. This agreement is achieved despite using 90 % less data in the nonuniform sampling case, which results in a 85.8 % reduction in computation time.

2.4.4 Combining multiple trajectories

DMD is often applied to experimental data, which are typically noisy. While filtering or phase-averaging can be done to eliminate noise prior to DMD analysis, this is not always desirable, as it may remove features of the true dynamics. In POD analysis,

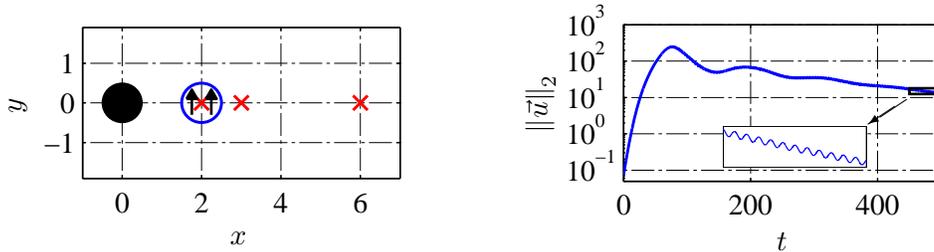


Figure 2.3: (Left) Schematic showing the placement of sensors (\times) and actuators (\circ) used to control the flow past a two-dimensional cylinder. (Right) Kinetic energy of the corresponding impulse response (restricted to the stable subspace). After an initial period of non-normal growth, oscillations with both short and long time scales are observed.

Table 2.1: Comparison of DMD eigenvalues*

Frequency			Growth rate		
Seq. DMD	Nonseq. DMD	Error	Seq. DMD	Nonseq. DMD	Error
0.118	0.118	0.00%	0.998	0.998	0.00%
0.127	0.127	0.01%	0.988	0.988	0.00%
0.107	0.106	3.40%	0.979	0.977	2.10%
0.138	0.139	7.50%	0.964	0.964	0.05%

* Row order corresponds to decreasing mode norm.

the effects of noise can be averaged out by combining multiple trajectories in a single POD computation. We can take the same approach in DMD analysis using (2.12).

Consider multiple dynamic trajectories, indexed by j : $\{\mathbf{x}_k^j\}_{k=0}^{m_j}$. These could be multiple runs of an experiment, or particular slices of a single, long trajectory. (The latter might be useful in trying to isolate the dynamics of a recurring dynamic event.) Suppose there are a total of J trajectories. DMD can be applied to the entire ensemble of trajectories by defining

$$\mathbf{X} \triangleq \begin{bmatrix} | & & | & & | & & | \\ \mathbf{x}_0^0 & \cdots & \mathbf{x}_{m_0-1}^0 & \mathbf{x}_0^1 & \cdots & \mathbf{x}_{m_1-1}^1 & \cdots & \mathbf{x}_0^J & \cdots & \mathbf{x}_{m_J-1}^J \\ | & & | & & | & & | \end{bmatrix}, \quad (2.17)$$

$$\mathbf{X}' \triangleq \begin{bmatrix} | & & | & & | & & | \\ \mathbf{x}_1^0 & \cdots & \mathbf{x}_{m_0}^0 & \mathbf{x}_1^1 & \cdots & \mathbf{x}_{m_1}^1 & \cdots & \mathbf{x}_1^J & \cdots & \mathbf{x}_{m_J}^J \\ | & & | & & | & & | \end{bmatrix}.$$

We demonstrate this approach using experimental data from a bluff-body wake experiment. A finite-thickness flat plate with an elliptical leading edge is placed in a uniform oncoming flow. Figure 2.6 shows a schematic of the experimental setup. We capture snapshots of the velocity field in the wake behind the body using a time-resolved particle image velocimetry (TRPIV) system. (This flow is also analyzed in

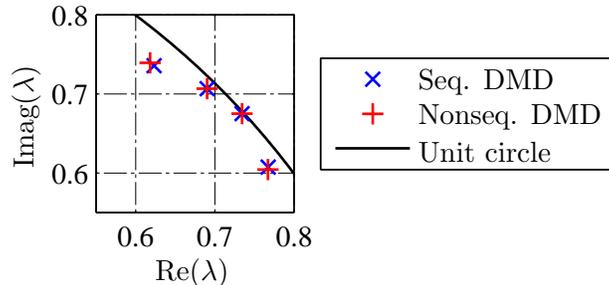


Figure 2.4: DMD estimates of the eigenvalues governing the decay of the impulse response shown in Figure 2.3 (right). The slowest decaying eigenvalues are captured well with both uniform sampling (sequential DMD) and nonuniform sampling (nonsequential DMD).

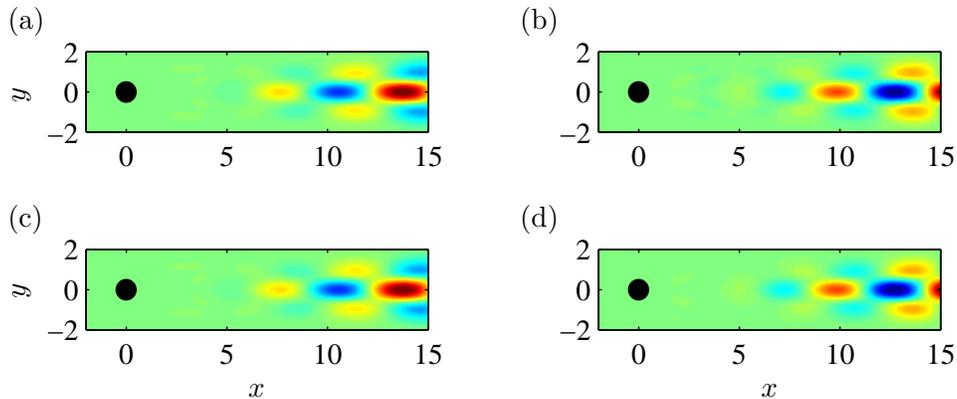


Figure 2.5: Comparison of dominant DMD modes computed from the impulse response shown in Figure 2.3 (right), illustrated using contours of vorticity. For each of the dominant frequencies, modes computed using nonuniform sampling (nonsequential DMD; bottom row) match those computed using uniform sampling (sequential DMD; top row). (For brevity, only the real part of each mode is shown; similar agreement is observed in the imaginary parts.) (a) $f = 0.118$, uniform sampling; (b) $f = 0.127$, uniform sampling; (c) $f = 0.118$, nonuniform sampling; (d) $f = 0.127$, nonuniform sampling.

Chapter 5; more details on the experimental setup can be found in Section 5.4.¹ Multiple experimental runs are conducted, with approximately 1,400 velocity fields captured in each run. This corresponds to the maximum amount of data that can be collected per run.

Due to the high Reynolds number ($Re = 50,000$), the flow is turbulent. As such, though we observe a standard von Kármán vortex street (Figure 2.7), the familiar vortical structures are contaminated by turbulent fluctuations. Figure 2.8 (left) shows a DMD spectrum computed using TRPIV data from a single experimental run.² The spectrum is characterized by a harmonic set of peaks, with the dominant peak corresponding to the wake shedding frequency. The corresponding modes are shown in Figure 2.9 (a–c). We see that the first pair of modes (Figure 2.9 (a)) exhibits top-bottom symmetry, with respect to the centerline of the body. The second pair of modes (Figure 2.9 (b)) shows something close to top-bottom antisymmetry, though variations in the vorticity contours make this antisymmetry inexact. The third pair of modes (Figure 2.9 (c)) again shows top-bottom symmetry, with structures that are roughly spatial harmonics of those seen in the first mode pair.

These modal features are to be expected, based on two-dimensional computations of a similar flow configuration [172]. However, when computed from noise-free simulation data, the symmetry/antisymmetry of the modes is more exact. Figures 2.8 (right) and 2.9 (d–g) show that when five experimental runs are used, the

¹Experimental data acquisition and processing was done by John Griffin (University of Florida) and Adam Hart (University of Florida Research and Engineering Education Facility).

²Instead of simply plotting the mode norms against their corresponding frequencies, as is generally done, we first scale the mode norms by λ^m . This reduces the height of spectral peaks corresponding to modes with large norm but quickly decaying eigenvalues. For dynamics known to lie on an attractor, such peaks can be misleading; they do not contribute to the long-time evolution of the system.

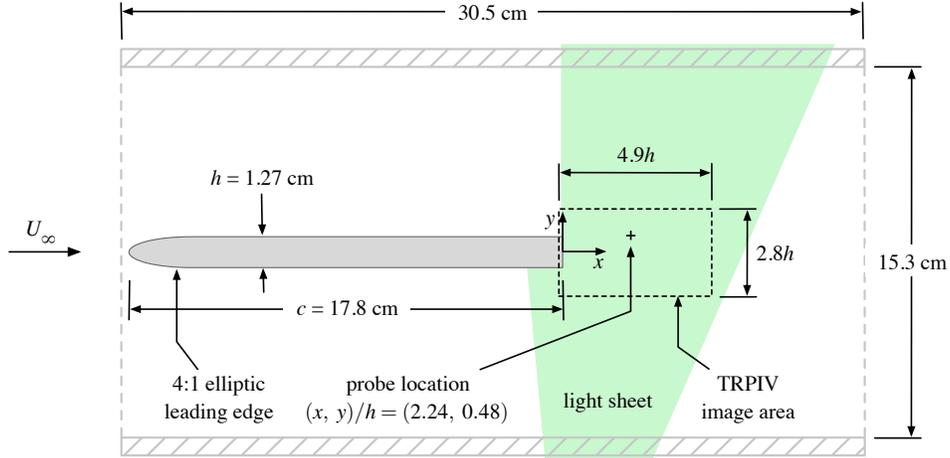


Figure 2.6: Schematic of setup for bluff-body wake experiment. (Figure courtesy of John Griffin.)

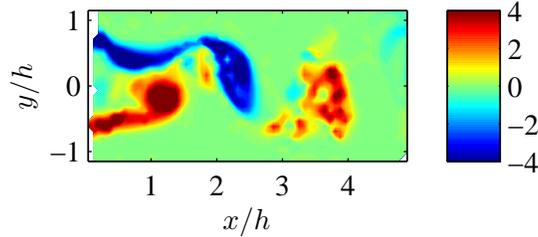


Figure 2.7: Typical vorticity field from the bluff-body wake experiment depicted in Figure 2.6. A clear von Kármán vortex street is observed, though the flow field is contaminated by turbulent fluctuations.

experimental DMD results improve, more closely matching computational results. In the DMD spectrum (Figure 2.8 (right)), we again observe harmonic peaks, with a fundamental frequency corresponding to the shedding frequency. The peaks are more isolated those in Figure 2.8 (left); in fact, we observe a fourth frequency peak, which is not observed in the single-run computation. The modal structures, shown in Figure 2.9 (d–g), display more obvious symmetry and antisymmetry, respectively. The structures are also smoother and more elliptical.

2.5 Connections to Koopman operator theory

The connections between Koopman operator theory and DMD were first explored in [138]. We summarize those findings below and discuss how they relate to the theory introduced in Section 2.2. Consider a discrete-time system with dynamics

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k),$$

where $\mathbf{x} \in \mathcal{M}$, a finite-dimensional manifold. The Koopman operator \mathcal{U} acts on scalar functions $g : \mathcal{M} \rightarrow \mathbb{R}$ or \mathbb{C} , mapping g to a new function $\mathcal{U}g$ whose action is

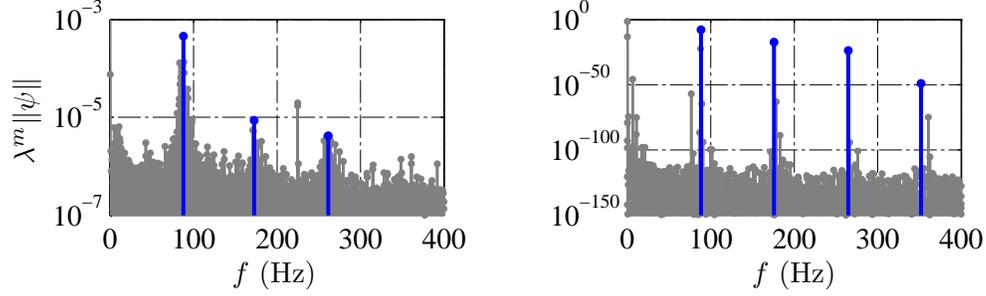


Figure 2.8: Comparison of DMD spectra computed using a single experimental run (left) and five experimental runs (right). When multiple runs are used, the spectral peaks are more isolated and occur at almost exactly harmonic frequencies. Furthermore, a fourth harmonic peak is identified; this peak is obscured in the single-run DMD computation. (Peaks corresponding to modes depicted in Figure 2.9 are shown in blue.)

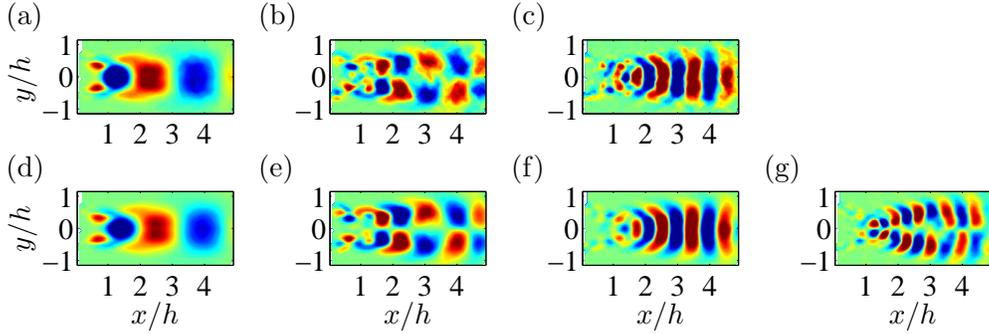


Figure 2.9: Representative DMD modes, illustrated using contours of vorticity. (For brevity, only the real part of each mode is shown.) The modes computed using multiple runs (bottom row) have more exact symmetry/antisymmetry and smoother contours, as expected. (a) $f = 87.75$ Hz, single run; (b) $f = 172.6$ Hz, single run; (c) $f = 261.2$ Hz, single run; (d) $f = 88.39$ Hz, five runs; (e) $f = 175.6$ Hz, five runs; (f) $f = 264.8$ Hz, five runs; (g) $f = 351.8$ Hz, five runs.

given by

$$\mathcal{U}g(\mathbf{x}_k) \triangleq g(\mathbf{f}(\mathbf{x}_k)) = g(\mathbf{x}_{k+1}). \quad (2.18)$$

We observe that \mathcal{U} acts linearly on functions g , even though the dynamics defined by \mathbf{f} may be nonlinear.

As such, consider the eigendecomposition

$$\mathcal{U}\theta_j(x) = \lambda_j\theta_j(x), \quad j = 0, 1, \dots \quad (2.19)$$

We can express a vector-valued function $\mathbf{h} : \mathcal{M} \rightarrow \mathbb{R}^n$ or \mathbb{C}^n in terms of these eigenfunctions as

$$\mathbf{h}(\mathbf{x}) = \sum_{j=0}^{\infty} \theta_j(\mathbf{x}) \hat{\psi}_j,$$

where $\hat{\boldsymbol{\psi}}_j \in \mathbb{R}^n$ or \mathbb{C}^n . (We assume each component of \mathbf{h} lies in the span of the eigenfunctions.) We refer to the vectors $\hat{\boldsymbol{\psi}}_j$ as the *Koopman modes*.

Substituting (2.19) into (2.18), we can then write

$$\mathbf{h}(\mathbf{x}_k) = \sum_{j=0}^{\infty} \lambda_j^k \theta_j(\mathbf{x}_0) \hat{\boldsymbol{\psi}}_j. \quad (2.20)$$

We see that each mode $\hat{\boldsymbol{\psi}}_j$ has a growth rate and frequency determined by λ_j , and that the Koopman modes and eigenvalues completely determine the evolution of \mathbf{h} along the trajectory starting at \mathbf{x}_0 .

Suppose we collect sequential snapshots of \mathbf{h} and stack them as in (2.9), with $\mathbf{h}_j = \mathbf{h}(\mathbf{x}_j)$ replacing \mathbf{x}_j . From these data we can compute DMD modes and eigenvalues using the standard SVD-based DMD algorithm. If the columns of \mathbf{X} are linearly independent and the DMD eigenvalues are distinct, then the DMD modes $\{\boldsymbol{\psi}_j\}_{j=0}^{m-1}$ and eigenvalues $\{\lambda_j\}_{j=0}^{m-1}$ are the unique solution to

$$\begin{aligned} \mathbf{h}_k &= \sum_{i=0}^{m-1} \lambda_i^k \boldsymbol{\psi}_i & k = 0, \dots, m-1 \\ \mathbf{h}_m &= \sum_{i=0}^{m-1} \lambda_i^m \boldsymbol{\psi}_i + \mathbf{r} & \mathbf{r} \perp \text{span}\{\mathbf{x}_0, \dots, \mathbf{x}_{m-1}\}, \end{aligned} \quad (2.21)$$

as shown in [31]. (When DMD is applied to non-sequential time-series, it is easily verified that (2.21) does not generally hold.)

We observe that (2.21) mirrors (2.20), with a finite sum in the former replacing the infinite sum in the latter. When there is no residual \mathbf{r} , then the DMD modes cannot be distinguished from the Koopman modes. It is for this reason that [138] considers DMD to be a numerical algorithm for computing approximate Koopman modes and eigenvalues. Because the modes $\{\boldsymbol{\psi}_j\}_{j=0}^{m-1}$ are generated using the standard SVD-based DMD algorithm, they satisfy Definition 1, and we say that Definition 1 preserves the connections between DMD and Koopman operator theory.

Using an operator theoretic argument, [13] interprets DMD the same way. Recalling (2.18), we see that in a sense, the Koopman operator \mathcal{U} sends $\mathbf{h}(\mathbf{x}_k)$ to $\mathbf{h}(\mathbf{x}_{k+1})$. This action can be represented with the infinite-dimensional left shift operator

$$\mathbf{S} \triangleq \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & & \ddots & & \vdots \end{bmatrix}.$$

For a data matrix \mathbf{X} with full column rank, the standard SVD-based DMD algorithm is equivalent to an analysis of the companion matrix

$$\mathbf{C} \triangleq \begin{bmatrix} 0 & 0 & \cdots & 0 & c_0 \\ 1 & 0 & \cdots & 0 & c_1 \\ 0 & 1 & \cdots & 0 & c_2 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & \cdots & 1 & c_{m-1} \end{bmatrix}.$$

We can think of \mathbf{C} as a finite-dimensional approximation of \mathbf{S} , restricted to column space of \mathbf{X} . Consequently, we can think of DMD as a finite-dimensional approximation of Koopman spectral analysis.

Though the Koopman analog provides a firm mathematical foundation for applying DMD to data generated by nonlinear systems, it is limited by the fact that it relies on (2.21). It was shown in [31] that if the columns of \mathbf{X} are linearly dependent or the DMD eigenvalues are not distinct, then the choice of modes and eigenvalues that satisfies (2.21) is not unique. It is even possible to generate matrices \mathbf{X} and \mathbf{X}' so that (2.21) cannot be satisfied by *any* set of modes and eigenvalues. For instance, one can easily verify that the modes and eigenvalues generated by the standard SVD-based DMD algorithm do not satisfy (2.21) in the case of repeated DMD eigenvalues. Thus the Koopman analogy is most appropriate in the case that \mathbf{X} describes a sequential time-series, \mathbf{X} has linearly independent columns, and the DMD eigenvalues are distinct.³ (The second condition requires that the number of snapshots be smaller than the measurement dimension n , which can be limiting when the observable \mathbf{h} is low-dimensional.) In contrast, an understanding of DMD built on Definition 1 holds even when these conditions break down.

2.6 Equivalence to LIM

In this section, we investigate the connections between DMD and LIM. To set up this discussion, we briefly introduce and define a number of terms used in the climate science literature. This is followed by a more in-depth description of LIM. Finally, we show that under certain conditions, LIM and DMD are equivalent.

2.6.1 Nomenclature

Empirical orthogonal functions (EOFs) were first introduced in 1956 by Lorenz [95]. While the term ‘‘EOF’’ is unique to the climate science literature, EOFs simply arise from the application of principal component analysis (PCA) [79, 80, 123] to meteorological data [124]. As a result, EOF analysis is equivalent to PCA, and thus also to POD and SVD. (We note that in PCA and EOF analysis, the data mean is always subtracted, so that the results can be interpreted in terms of variances; this is often done for POD as well.)

³In general, it is unclear how well DMD approximates Koopman spectral analysis. For instance, in some systems DMD modes and eigenvalues closely approximate those of the Koopman operator near an attractor, but not far from it [14]. DMD may also perform poorly when applied to dynamics whose Koopman spectral decomposition contains Jordan blocks [14].

In practice, EOFs are often used as a particular choice of *principal interaction patterns* (PIPs), a concept introduced in 1988 by Hasselmann [75]. The following discussion uses notation similar to that found in [177], which provides a nice review of PIP concepts. Consider a dynamical system with a high-dimensional state $\mathbf{x}(t) \in \mathbb{R}^n$. In some cases, such a system may be approximately “driven” by a lower-dimensional system with state $\mathbf{z}(t) \in \mathbb{R}^r$, where $r < n$. To be precise, we say that \mathbf{x} and \mathbf{z} are related as follows:

$$\begin{aligned}\mathbf{z}_{k+1} &= \mathbf{F}(\mathbf{z}_k; \boldsymbol{\alpha}) + \text{noise} \\ \mathbf{x}_k &= \mathbf{P}\mathbf{z}_k + \text{noise},\end{aligned}$$

where $\boldsymbol{\alpha}$ is a vector of parameters. From this we see that given a knowledge of \mathbf{z} and its dynamics, \mathbf{x} is completely specified by the static map \mathbf{P} , aside from the effects of noise. Though \mathbf{P} cannot be inverted, given a measurement of \mathbf{x} , we can approximate \mathbf{z} using a least-squares fit:

$$\mathbf{z}_k = (\mathbf{P}^T \mathbf{P})^{-1} \mathbf{P}^T \mathbf{x}_k.$$

In climate science, the general approach of modeling the dynamics of a high-dimensional variable \mathbf{x} through a lower-dimensional variable \mathbf{z} is referred to as *inverse modeling*. The inverse model described above requires definitions of \mathbf{F} , \mathbf{P} , and $\boldsymbol{\alpha}$. Generally, \mathbf{F} is chosen based on physical intuition. Once that choice is made, \mathbf{P} and $\boldsymbol{\alpha}$ are fitted simultaneously. The PIPs are the columns of \mathbf{P} for the choice of \mathbf{P} (and $\boldsymbol{\alpha}$) that minimizes the error

$$\epsilon(\mathbf{P}, \boldsymbol{\alpha}) \triangleq E \left(\left\| \mathbf{x}_{k+1} - \mathbf{x}_k - \mathbf{P} \left(\mathbf{F}(\mathbf{z}_k; \boldsymbol{\alpha}) - \mathbf{z}_k \right) \right\| \right),$$

where E is the expected value operator [177]. In general, the choice of \mathbf{P} is not unique.

Hasselmann also introduced the notion of *principal oscillation patterns* (POPs) in his 1988 paper [75]. Again, we use the notation based on the review by von Storch et al. [177]. Consider a system with unknown dynamics. We assume that we can approximate these dynamics with a linear system

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \text{noise}.$$

If we multiply both sides by \mathbf{x}_k^T and take expected values, we can solve for \mathbf{A} as

$$\mathbf{A} = E(\mathbf{x}_{k+1} \mathbf{x}_k^T) E(\mathbf{x}_k \mathbf{x}_k^T)^{-1}. \quad (2.22)$$

The eigenvectors of \mathbf{A} are referred to as POPs. That is, POPs are eigenvectors of a particular linear approximation of otherwise unknown dynamics.

Even within the climate science literature, there is some confusion between PIPs and POPs. This is due to the fact that POPs can be considered a special case of PIPs. In general, PIPs are basis vectors spanning a low-dimensional subspace useful for reduced-order modeling. Suppose we model our dynamics with the linear approximation described above, and do not reduce the order of the state. If we then express the model in its eigenvector basis, we are choosing our PIPs to be POPs.

2.6.2 Defining LIM

In 1989, Penland derived a method for computing a linear, discrete-time system that approximates the trajectory of a stochastic, continuous-time, linear system, which he referred to as a linear Markov model [124]. We describe this method, which came to be known as LIM, using the notation found in [125]. Consider an n -dimensional Markov process

$$\frac{d\mathbf{x}}{dt} = \mathbf{B}\mathbf{x}(t) + \boldsymbol{\xi}(t),$$

where $\boldsymbol{\xi}(t)$ is white noise with covariance

$$\mathbf{Q} = E(\boldsymbol{\xi}(t)\boldsymbol{\xi}^T(t)).$$

We assume the mean of the process has been removed. The covariance of \mathbf{x} is given by

$$\boldsymbol{\Lambda} = E(\mathbf{x}(t)\mathbf{x}^T(t)). \quad (2.23)$$

One can show that the following must hold:

$$\begin{aligned} \mathbf{B}\boldsymbol{\Lambda} + \boldsymbol{\Lambda}\mathbf{B}^T + \mathbf{Q} &= 0 \\ E(\mathbf{x}(t+\tau)\mathbf{x}^T(t)) &= \exp(\mathbf{B}\tau)\boldsymbol{\Lambda}. \end{aligned}$$

(See [124] for details.)

Defining the Green's function

$$\begin{aligned} \mathbf{G}(\tau) &\triangleq \exp(\mathbf{B}\tau) \\ &= E(\mathbf{x}(t+\tau)\mathbf{x}^T(t)) \boldsymbol{\Lambda}^{-1}, \end{aligned} \quad (2.24)$$

we can say that given a state $\mathbf{x}(t)$, the most probable state time τ later is

$$\mathbf{x}(t+\tau) = \mathbf{G}(\tau)\mathbf{x}(t).$$

The operator $\mathbf{G}(\tau)$ is computed from snapshots of the continuous-time system and has the same form as the linear approximation used in POP analysis (see (2.22)). We note that we arrive at the same model if we apply linear stochastic estimation to snapshots of the state \mathbf{x} , taking $\mathbf{x}(t)$ and $\mathbf{x}(t+\tau)$ to be the unconditional and conditional variables, respectively. (This is done in Chapter 5 to identify a model for the evolution of POD coefficients in a fluid flow.)

When this approach is applied to a nonlinear system, it can be shown that $\mathbf{G}(\tau)$ is equivalent to a weighted average of the nonlinear dynamics, evaluated over an ensemble of snapshots [23]. This is in contrast to a typical linearization, which involves evaluating the Jacobian of the dynamics at a fixed point. If the true dynamics are nearly linear, these two approaches will yield nearly the same model. However, if nonlinear effects are significant, $\mathbf{G}(\tau)$ will be closer to the ensemble average, and arguably a better model than a traditional linearization [23].

In [125], this method was applied to compute a linear Markov model in the space of EOF coefficients. This is an example of inverse modeling (equivalently, PIP analysis); a high-dimensional variable is modeled via a projection onto a lower-dimensional EOF subspace. Due to the assumption of linear dynamics, this approach

came to be known as *linear inverse modeling*. The combination of PIP and POP concepts in this early work has contributed to the continuing confusion between PIPs and POPs in the climate science literature today.

2.6.3 Proof of equivalence

In Definition 1, the DMD eigenvalues are given by the eigenvalues of a projected linear operator $\tilde{\mathbf{A}}$. The DMD modes are computed by lifting the eigenvectors of $\tilde{\mathbf{A}}$ to the original space via the left singular vectors Φ (see (2.5)). In [125], the eigendecomposition of a low-order linear model $\mathbf{G}(\tau)$ is computed and the low-order eigenvectors lifted to the original space via EOFs, in the same way as in DMD. The similarity in these two approaches is obvious. Recall that left singular vectors and EOFs are equivalent, so long as they are computed from the same data. Then to prove that an LIM-based eigenvector analysis is equivalent to DMD, we simply have to show the equivalence of $\mathbf{G}(\tau)$ and $\tilde{\mathbf{A}}$.

Consider two $n \times m$ data matrices \mathbf{X} and \mathbf{X}' , with columns $\mathbf{x}_j = \mathbf{x}(t_j)$ and $\mathbf{x}'_j = \mathbf{x}(t_j + \tau)$, respectively. \mathbf{X} and \mathbf{X}' may or may not share columns. As in (2.1), we assume that the EOFs to be used for LIM are computed from \mathbf{X} alone, giving us

$$\mathbf{X} = \Phi \Sigma \mathbf{W}^*,$$

where the columns of Φ are the EOFs. The EOF coefficients of \mathbf{X} and \mathbf{X}' are given by

$$\hat{\mathbf{X}} = \Phi^* \mathbf{X}, \quad \hat{\mathbf{X}}' = \Phi^* \mathbf{X}', \quad (2.25)$$

whose columns we denote by $\hat{\mathbf{x}}_j$ and $\hat{\mathbf{x}}'_j$, respectively.

In order to show that $\mathbf{G}(\tau)$ and $\tilde{\mathbf{A}}$ are equivalent, we must reduce (2.24) to (2.3). Because we are interested in the equivalence of LIM and DMD when the former is performed in the space of EOF coefficients, we replace all instances of \mathbf{x} in (2.23) and (2.24) with $\hat{\mathbf{x}}$. Recall that the expected value of \mathbf{a} , for an ensemble $\{\mathbf{a}_j\}_{j=0}^{m-1}$, is given by

$$E(\mathbf{a}_j) \triangleq \frac{1}{m} \sum_{j=0}^{m-1} \mathbf{a}_j. \quad (2.26)$$

Then we can rewrite (2.23) as

$$\begin{aligned} \mathbf{\Lambda} &= \frac{1}{m} \sum_{j=0}^{m-1} \hat{\mathbf{x}}_j \hat{\mathbf{x}}_j^* \\ &= \frac{1}{m} \hat{\mathbf{X}} \hat{\mathbf{X}}^* \\ &= \frac{1}{m} \Phi^* \mathbf{X} \mathbf{X}^* \Phi \\ &= \frac{1}{m} \Phi^* \Phi \Sigma^2 \\ &= \frac{1}{m} \Sigma^2, \end{aligned}$$

using the fact that $\mathbf{X}\mathbf{X}^*\Phi = \Phi\Sigma^2$, by the definition of left singular vectors. This result, along with (2.1), allows us to rewrite (2.24) as

$$\begin{aligned}\mathbf{G}(\tau) &= \left(\frac{1}{m} \sum_{j=0}^{m-1} \hat{\mathbf{x}}'_j \hat{\mathbf{x}}_j^* \right) (m\Sigma^{-2}) \\ &= \hat{\mathbf{X}}' \hat{\mathbf{X}}^* \Sigma^{-2} \\ &= \Phi^* \mathbf{X}' \mathbf{X}^* \Phi \Sigma^{-2} \\ &= \Phi^* \mathbf{X}' \mathbf{W} \Sigma \Phi^* \Phi \Sigma^{-2} \\ &= \Phi^* \mathbf{X}' \mathbf{W} \Sigma^{-1}.\end{aligned}$$

(Recall that $\mathbf{x}(t_j + \tau) = \mathbf{x}'_j$, and $\mathbf{x}(t_j) = \mathbf{x}_j$.) From (2.10), we then have $\mathbf{G}(\tau) = \tilde{\mathbf{A}}$, and we see that DMD and LIM are built on the same low-dimensional, approximating linear dynamics.⁴

We emphasize that this equivalence relies on a number of assumptions. First, we assume that we perform LIM in the space of EOF coefficients. Second, we assume that the EOFs are computed from \mathbf{X} alone. This may not be an intuitive choice if \mathbf{X} and \mathbf{X}' are completely distinct, but for a sequential snapshot sequence where \mathbf{X} and \mathbf{X}' differ by a single column, this is not a significant difference. Given these assumptions, the equivalence of DMD and LIM gives us yet another way to interpret DMD analysis. If the data mean is removed, then the low-order map that generates the DMD eigenvalues and eigenvectors is simply the one that yields the statistically most likely state in the future. In a small sense, the DMD framework is more general, as the interpretation provided by Definition 1 holds even for data that is not mean-subtracted. Then again, in LIM the computation of the EOFs is completely divorced from the modeling procedure, allowing for a computation using both \mathbf{X} and \mathbf{X}' . Nevertheless, the similarities between the two methods suggests that practitioners of DMD would be well-served in studying and learning from the climate science/LIM literature.

2.7 Conclusions

We have presented a new definition in which DMD is defined to be an approximate eigendecomposition of an approximating linear operator. Whereas existing DMD theory focuses on full-rank, sequential time-series, our theory applies to arbitrary datasets. At the same time, our definition agrees with the commonly used, SVD-based DMD algorithm and preserves the links between DMD and Koopman operator theory. Thus our framework can be considered to be an extension of existing DMD theory to a more general class of datasets.

For instance, when analyzing data generated by a dynamical system, we require only that the columns of the data matrices \mathbf{X} and \mathbf{X}' be related by the dynamics of interest, in a pairwise fashion. Unlike existing DMD algorithms, we do not require that the data come from uniform sampling of a single time-series, nor do we require that the columns of \mathbf{X} and \mathbf{X}' overlap. We demonstrated the utility of this approach using two numerical examples. In the first, we sampled a trajectory non-uniformly,

⁴The above proof was developed together with Steven L. Brunton (University of Washington).

significantly reducing computational costs. In the second, we concatenated multiple datasets in a single DMD computation, effectively averaging out the effects of noise. Our generalized interpretation of DMD also proved useful in explaining the results of DMD computations involving rank-deficient datasets. Such computations may provide either meaningful or misleading information, depending on the dataset, and are not treated in the existing DMD literature.

Finally, we showed that DMD is closely related to LIM, a method developed decades ago in the climate science community. The two methods are in fact equivalent under certain assumptions. This suggests that lessons learned from past applications of LIM can inform strategies for future applications of DMD.

Chapter 3

A DMD-based extension of the BPOD algorithm

We now shift our focus from the theory of dynamic mode decomposition (DMD) to its application. While one of the main draws of DMD is that it can be used to analyze nonlinear dynamics, there are also situations in which DMD may be an appropriate tool for analyzing linear systems. In the present chapter, we demonstrate the utility of DMD in computing *snapshot-based* eigendecompositions. The availability of a snapshot-based algorithm is critical in applications where it is inefficient, or impossible, to compute the action of a dynamical operator on arbitrary initial conditions. This restriction prohibits the use of standard methods like the Lanczos and Arnoldi algorithms [166].

We focus on a DMD-based modification of the balanced proper orthogonal decomposition (BPOD) algorithm for systems with simple impulse response tails. In this new variant of BPOD, we use DMD to estimate the slowly decaying eigenvectors that dominate the long-time behavior (the “tails”) of the direct and adjoint impulse responses. We then formulate analytic expressions for the contribution of these eigenvectors to the controllability and observability Gramians. These contributions can be accounted for in the BPOD algorithm by simply appending the impulse response snapshot matrices (direct and adjoint, respectively) with particular linear combinations of the slow eigenvectors. Aside from these additions to the snapshot matrices, the algorithm remains unchanged. By treating the tails analytically, we eliminate the need to run long impulse response simulations, lowering storage requirements and speeding up ensuing computations. To demonstrate its effectiveness, we apply this method to two examples: the linearized, complex Ginzburg-Landau (CGL) equation and the two-dimensional fluid flow past a cylinder. As expected, reduced-order models computed using an analytic tail match or exceed the accuracy of those computed using the standard BPOD procedure, at a fraction of the cost.

The remainder of this chapter is organized as follows: Section 3.2 motivates the need for an improved BPOD algorithm and provides a brief introduction to empirical Gramians, balanced truncation, and the standard BPOD algorithm. Section 3.3 builds on this theory to develop the analytic tail method. Both a complex and real formulation are derived. In Section 3.4 we describe how DMD is used to estimate the eigenvectors and eigenvalues required to describe an impulse response tail. Finally, in Section 3.5 we demonstrate the effectiveness of the analytic tail method using a

number of examples. We note that this material is based on the article by Tu and Rowley [171], and to avoid confusion we use the notation found in that work.

3.1 Motivation

Model reduction is an increasingly common approach in numerical flow control studies. A typical discretization of the Navier–Stokes equations can produce a dynamical system with over 10^6 states, making standard control design procedures prohibitively expensive. However, the gross behavior of a fluid system can be much simpler than its state dimension would suggest. In such flows, a reduced-order model may be able to capture the dominant behavior using a relatively small number of states. For instance, the main features of vortex shedding behind a cylinder at low Reynolds numbers can be captured with a three-dimensional model [115]. These low-order models can then be used for control design, and in the course of their development important underlying physical mechanisms may be discovered.

Of the many model reduction techniques, balanced truncation is an especially well-suited choice for control-oriented applications. The resulting models balance the controllability and observability of a stable, linear system. (Unstable systems can be treated by decoupling the stable and unstable dynamics, as done by Barbagallo et al. [20] and Ahuja and Rowley [4].) Modes that are neither highly controllable nor highly observable are truncated. These modes are exactly those that cannot be easily affected by actuation or easily measured with sensors. In other words, they have little effect on the *input-output* dynamics of the system, and as such are not useful for control design.

Unfortunately, for very high-dimensional systems, the standard balanced truncation technique is impractical, requiring the solution of high-dimensional Lyapunov equations. A number of methods have been developed to iteratively solve such equations, including the classical Smith method [156] and its cyclic low-rank variant [126]. Alternatively, it is possible to avoid solving Lyapunov equations entirely. BPOD is a snapshot-based approximation to balanced truncation that takes this approach, making it suitable for large systems [137]. It has been used to great effect in a variety of flow control applications. For instance, Ilak and Rowley [84] used BPOD to accurately model the nonnormal transient growth in a linearized channel flow. Ahuja and Rowley [4] used BPOD to design estimator-based controllers that stabilized unstable steady states of the flow past a flat plate at a high angle of attack. BPOD-based controllers were also used by Bagheri et al. [15] and Semeraro et al. [150] to suppress the growth of perturbations in a boundary layer. Dergham et al. [43] used BPOD to model the flow over a backward-facing step, showing that a small number of input projection modes is able to capture the effect of arbitrarily placed localized actuators.

In balanced truncation, the product of the controllability and observability Gramians is used to find a transformation to a balanced coordinate system. In BPOD, a similar computation is performed. Impulse response simulations of the direct and adjoint systems are sampled and the resulting snapshots are collected into large matrices. The product of these snapshot matrices approximates the Hankel matrix, from which an approximate balancing transformation can be found.

Two obvious sources of error are inherent in approximating the Hankel matrix this way. The first is in discretely sampling the continuously varying impulse responses. The second is in truncating the impulse responses at a finite time. In practice, both of these are dealt with by using convergence tests, with respect to the sampling frequency and simulation lengths, respectively. However, such tests can be costly for large-scale simulations.

As an alternative, we propose a method for incorporating the effect of the truncated snapshots, based on analytic considerations. For a stable, linear system, the long-time behavior of the impulse response is dominated by the system's slowest eigenvectors alone. After enough time has elapsed, the contribution of all other eigenvectors to the state will have decayed to nearly zero. When the number of these slow eigenvectors is small, we say that the system has a *simple impulse response tail*. We use DMD to estimate these slow eigenvectors and eigenvalues. We then express the state at the beginning of the tail as a linear combination of the slow eigenvectors. The further evolution of the state is completely characterized by the corresponding eigenvalues, so no further snapshots need to be saved, reducing the required storage space and simulation time. The contribution of the tail to the Hankel matrix can then be computed analytically, as a function of the slow eigenvectors and eigenvalues.

3.2 Background

3.2.1 Empirical Gramians

Consider the stable, linear system

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} & \mathbf{x} &\in \mathbb{R}^n, \mathbf{u} \in \mathbb{R}^p \\ \mathbf{y} &= \mathbf{C}\mathbf{x} & \mathbf{y} &\in \mathbb{R}^q. \end{aligned} \tag{3.1}$$

The controllability and observability Gramians are given by

$$\mathbf{W}_c = \int_0^\infty e^{\mathbf{A}t} \mathbf{B} \mathbf{B}^* e^{\mathbf{A}^*t} dt \quad \mathbf{W}_o = \int_0^\infty e^{\mathbf{A}^*t} \mathbf{C}^* \mathbf{C} e^{\mathbf{A}t} dt,$$

where asterisks denote the conjugate transpose of a matrix. The controllability Gramian provides a measure of how easily a state is affected by actuation, while the observability Gramian describes how easily a state excites a sensor measurement.

Typically, the Gramians are computed by solving the Lyapunov equations

$$\mathbf{A}\mathbf{W}_c + \mathbf{W}_c\mathbf{A}^* + \mathbf{B}\mathbf{B}^* = 0 \quad \mathbf{A}^*\mathbf{W}_o + \mathbf{W}_o\mathbf{A} + \mathbf{C}^*\mathbf{C} = 0.$$

However, for very large systems, this can be numerically prohibitive. We can instead use data from numerical simulations to compute *empirical Gramians*. Suppose the system (3.1) has p inputs. Then we can write \mathbf{B} columnwise as

$$\mathbf{B} = [\mathbf{b}_1 \quad \cdots \quad \mathbf{b}_p],$$

and similarly,

$$\mathbf{u} = [u_1(t) \quad \cdots \quad u_p(t)]^T.$$

The response to a single impulsive input $u_j(t) = \delta(t)$ is then given by

$$\mathbf{x}_j(t) = e^{\mathbf{A}t}\mathbf{b}_j,$$

and the controllability Gramian can be rewritten as

$$\mathbf{W}_c = \int_0^\infty \sum_{j=1}^p \mathbf{x}_j(t)\mathbf{x}_j^*(t) dt. \quad (3.2)$$

To evaluate the right hand side, we run numerical simulations of the impulse responses, collecting snapshots of the state at discrete times t_1, t_2, \dots, t_m . We then scale each snapshot $\mathbf{x}_j(t_k)$ by an appropriate quadrature weight δ_k and collect the scaled snapshots in a data matrix

$$\mathbf{X} = [\mathbf{x}_1(t_1)\sqrt{\delta_1} \quad \cdots \quad \mathbf{x}_1(t_m)\sqrt{\delta_m} \quad \cdots \quad \mathbf{x}_p(t_1)\sqrt{\delta_1} \quad \cdots \quad \mathbf{x}_p(t_m)\sqrt{\delta_m}].$$

The integral (3.2) can then be approximated by a quadrature sum:

$$\mathbf{W}_c \approx \mathbf{X}\mathbf{X}^*. \quad (3.3)$$

We follow a similar procedure to compute the observability Gramian. Defining the adjoint system as

$$\dot{\mathbf{z}} = \mathbf{A}^*\mathbf{z} + \mathbf{C}^*\mathbf{w}, \quad (3.4)$$

we again sample impulse response simulations, scale the snapshots by quadrature weights, and form a data matrix \mathbf{Y} . The observability Gramian is then approximated by the quadrature sum $\mathbf{W}_o \approx \mathbf{Y}\mathbf{Y}^*$.

In approximating a Gramian this way, there are two clear sources of error. First, we are sampling the continuously varying impulse response at discrete points in time. However, if the sampling rate is sufficiently fast with respect to the dynamics of the system, this error should be minimal and can be further mitigated by using appropriate quadrature weights. The second source of error comes from truncating the impulse response at t_m , when the integral in (3.2) is evaluated to $t \rightarrow \infty$. For a stable system, the impulse response must eventually decay to zero. Thus if t_m is large enough, the contribution of the truncated snapshots to the Gramian will be negligible. However, it is unclear how to determine an appropriate truncation point given some a priori bound on the desired accuracy of the empirical Gramian. Furthermore, any such guideline would likely require knowledge about the eigenvalues (and possibly eigenvectors) of the system. For a large system, these may not be known, and can be expensive to compute (e.g., using an Arnoldi iteration).

3.2.2 Balanced truncation

Balanced truncation was developed by Moore [108] as a model reduction technique for stable, linear systems. For control applications, we are interested in the *input-output* dynamics of a system. As such, if a mode is difficult to affect with actuation (inputs) or hard to measure using sensors (outputs), then it is not particularly useful for control. Balanced truncation builds upon this simple idea by seeking a *balanced realization* of the system (3.1), in which the most controllable states are also the most observable. To get a reduced-order model, we then truncate those

states that are neither highly controllable nor observable. It is a standard result that if a system is both controllable and observable, then such a realization always exists (for example, see the standard references by Dullerud and Paganini [49] or Antoulas [9]).

In performing balanced truncation, we compute a coordinate transformation $\mathbf{x} = \mathbf{T}\tilde{\mathbf{x}}$ that balances the Gramians. Under this transformation, the Gramians become

$$\tilde{\mathbf{W}}_{\mathbf{c}} = \mathbf{T}^{-1}\mathbf{W}_{\mathbf{c}}(\mathbf{T}^{-1})^* \quad \tilde{\mathbf{W}}_{\mathbf{o}} = \mathbf{T}^*\mathbf{W}_{\mathbf{o}}\mathbf{T},$$

and in particular are equal and diagonal:

$$\tilde{\mathbf{W}}_{\mathbf{c}} = \tilde{\mathbf{W}}_{\mathbf{o}} = \boldsymbol{\Sigma}.$$

The non-zero elements σ_i of the diagonal matrix $\boldsymbol{\Sigma}$ satisfy $\sigma_1 \geq \dots \geq \sigma_n \geq 0$, and are known as the *Hankel singular values*.

The Hankel singular values can be used to compute a priori bounds on the error in approximating the system (3.1) with a reduced-order model. Let $\mathbf{G}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}$ be the transfer function of the original system, and $\mathbf{G}_r(s)$ be that of the reduced-order system of order r . Then the error is bounded below by the first truncated Hankel singular value:

$$\|\mathbf{G}(s) - \mathbf{G}_r(s)\|_{\infty} > \sigma_{r+1}. \quad (3.5)$$

This is a lower bound for *any* reduced-order approximation of $\mathbf{G}(s)$. For a balanced truncation model, we also have an upper bound given by

$$\|\mathbf{G}(s) - \mathbf{G}_r(s)\|_{\infty} < 2 \sum_{j=r+1}^n \sigma_j. \quad (3.6)$$

(These error bounds are standard results and can be found in Dullerud and Paganini [49], Antoulas [9], or other standard texts.)

3.2.3 BPOD

BPOD was developed by Rowley [137] as an approximation to balanced truncation. It is a snapshot-based method that avoids computation of the true Gramians $\mathbf{W}_{\mathbf{c}}$ and $\mathbf{W}_{\mathbf{o}}$. Instead, it makes use of the factors \mathbf{X} and \mathbf{Y} of the empirical Gramians in analyzing the Hankel matrix $\mathbf{H} = \mathbf{Y}^*\mathbf{X}$. This makes BPOD suitable for very high-dimensional systems, whereas balanced truncation is not. If we compute the singular value decomposition (SVD) of the Hankel matrix and write it as

$$\mathbf{H} = [\mathbf{U}_{\mathbf{H}} \quad \dots] \begin{bmatrix} \boldsymbol{\Sigma}_{\mathbf{H}} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{W}_{\mathbf{H}}^* \\ \vdots \end{bmatrix},$$

then the direct BPOD modes are then given by

$$\boldsymbol{\Phi} = \mathbf{X}\mathbf{W}_{\mathbf{H}}\boldsymbol{\Sigma}_{\mathbf{H}}^{-1/2}$$

and the adjoint BPOD modes by

$$\Psi = \mathbf{Y}\mathbf{U}_H\mathbf{\Sigma}_H^{-1/2}.$$

To get a reduced-order model of order r , we project the system (3.1) onto the span of the BPOD modes:

$$\begin{aligned} \dot{\mathbf{x}}_r &= \mathbf{A}_r\mathbf{x}_r + \mathbf{B}_r\mathbf{u} & \mathbf{A}_r &= \Psi_r^*\mathbf{A}\Phi_r \\ \mathbf{y}_r &= \mathbf{C}_r\mathbf{x}_r & \mathbf{B}_r &= \Psi_r^*\mathbf{B} \\ & & \mathbf{C}_r &= \mathbf{C}\Phi_r, \end{aligned}$$

where Φ_r and Ψ_r contain only the first r columns of Φ and Ψ , respectively. The entries of the diagonal matrix $\mathbf{\Sigma}_H$ provide an approximation to the Hankel singular values of the system (3.1) and can be used to estimate the error bounds given by (3.5) and (3.6). However, even if the true Hankel singular values are known, a BPOD-based model may not satisfy the theoretical upper error bound, as BPOD is only an approximation of balanced truncation. (The approximation comes in taking \mathbf{X} and \mathbf{Y} to be factors of the empirical, rather than true, Gramians. As such, the truncation and discrete sampling of the impulse responses are again to blame.)

3.3 Analytic tail method

3.3.1 Motivation

Many of the stable, linear systems studied in fluid dynamics exhibit what we refer to as a *simple impulse response tail*. For such a system, the long-time behavior of the impulse response is dominated by a small set of slowly decaying eigenvectors. If we can estimate these eigenvectors and their corresponding eigenvalues, then we can approximate the further evolution of the impulse response *analytically*, neglecting the fast eigenvectors whose contributions have already decayed to nearly zero. This reduces the required storage space for snapshots, a key consideration when dealing with large datasets. Furthermore, we can use this analytic expression to evaluate the contribution of the tail to a Gramian or to the Hankel matrix. This minimizes the error due to truncation by accounting for the effect of the impulse response(s) past the truncation point. (See Section 3.2.1 for a brief discussion of truncation error.)

3.3.2 Complex formulation

Consider the stable, linear system (3.1). Without loss of generality we assume a single-input system. If there are multiple inputs, the following procedure can be applied to each independently. Let $\mathbf{x}(t)$ be the response to an impulse in the input $u(t)$. Suppose that at some time T , we can approximate the state as a linear combination of M slow eigenvectors:

$$\mathbf{x}(T) = \sum_{j=1}^M \mathbf{v}_j,$$

where $\mathbf{A}\mathbf{v}_j = \lambda_j\mathbf{v}_j$ and we scale the eigenvectors \mathbf{v}_j to subsume any multiplicative constants. Then for $t \geq T$, the state is given by

$$\mathbf{x}(t) = \sum_{j=1}^M e^{\lambda_j(t-T)} \mathbf{v}_j,$$

or in matrix notation,

$$\mathbf{x}(t) = [\mathbf{v}_1 \ \cdots \ \mathbf{v}_M] \begin{bmatrix} e^{\lambda_1(t-T)} \\ \vdots \\ e^{\lambda_M(t-T)} \end{bmatrix}. \quad (3.7)$$

Suppose we want to compute the empirical controllability Gramian, as in (3.2). For our single-input system, we have

$$\mathbf{W}_c = \int_0^\infty \mathbf{x}(t)\mathbf{x}^*(t) dt.$$

For $t \geq T$, we can substitute for $\mathbf{x}(t)$ from (3.7), yielding

$$\mathbf{x}(t)\mathbf{x}^*(t) = \mathbf{V}_M \mathbf{M}(t) \mathbf{V}_M^*,$$

where

$$\mathbf{V}_M = [\mathbf{v}_1 \ \cdots \ \mathbf{v}_M]$$

and the elements of $\mathbf{M}(t)$ are given by

$$\mathbf{M}_{j,k}(t) = e^{(\lambda_j + \bar{\lambda}_k)(t-T)},$$

where $\bar{\lambda}_k$ is the complex conjugate of λ_k .

Splitting the integral at $t = T$, we can rewrite the controllability Gramian using our simple tail approximation:

$$\mathbf{W}_c = \int_0^T \mathbf{x}(t)\mathbf{x}^*(t) dt + \mathbf{V}_M \left(\int_T^\infty \mathbf{M}(t) dt \right) \mathbf{V}_M^*.$$

The integral of $\mathbf{M}(t)$ can be performed element-wise, recalling that the eigenvalues of \mathbf{A} all have negative real part:

$$\begin{aligned} \int_0^\infty \mathbf{M}_{j,k}(t) dt &= \int_0^\infty e^{(\lambda_j + \bar{\lambda}_k)(t-T)} dt \\ &= \lim_{t_f \rightarrow \infty} \frac{e^{(\lambda_j + \bar{\lambda}_k)(t-T)} \Big|_T^{t_f}}{\lambda_j + \bar{\lambda}_k} \\ &= -\frac{1}{\lambda_j + \bar{\lambda}_k}. \end{aligned}$$

Then we can write

$$\mathbf{W}_c = \int_0^T \mathbf{x}(t)\mathbf{x}^*(t) dt + \mathbf{V}_M \mathbf{N} \mathbf{V}_M^*, \quad (3.8)$$

where the elements of \mathbf{N} are given by

$$\mathbf{N}_{j,k} = -\frac{1}{\lambda_j + \bar{\lambda}_k}. \quad (3.9)$$

We wish to express (3.8) in a form that lends itself to the snapshot-based formulation. In other words, we seek an expression $\mathbf{W}_c = \mathbf{X}\mathbf{X}^*$ for some data matrix \mathbf{X} . If we collect impulse response snapshots at discrete times $t_1, \dots, t_m = T$, then the integral in (3.8) is given by

$$\int_0^T \mathbf{x}(t)\mathbf{x}^*(t) dt = \mathbf{X}_T\mathbf{X}_T^*,$$

where

$$\mathbf{X}_T = [\mathbf{x}(t_1)\sqrt{\delta_1} \quad \dots \quad \mathbf{x}(T)\sqrt{\delta_m}].$$

Then

$$\mathbf{W}_c = \mathbf{X}_T\mathbf{X}_T^* + \mathbf{V}_M\mathbf{N}\mathbf{V}_M^*. \quad (3.10)$$

We observe that as the product of a matrix with its transpose, $\mathbf{M}(t)$ is positive semi-definite. Since \mathbf{N} is just the integral of $\mathbf{M}(t)$, it too is positive semi-definite, and can be factored using a Cholesky decomposition as

$$\mathbf{N} = \mathbf{\Gamma}\mathbf{\Gamma}^*,$$

where $\mathbf{\Gamma}$ is lower-triangular.¹ We can then write

$$\mathbf{V}_M\mathbf{N}\mathbf{V}_M^* = \mathbf{V}_M\mathbf{\Gamma}\mathbf{\Gamma}^*\mathbf{V}_M^*,$$

allowing us to rewrite (3.10) as

$$\mathbf{W}_c = [\mathbf{X}_T \quad \mathbf{V}_M\mathbf{\Gamma}] [\mathbf{X}_T \quad \mathbf{V}_M\mathbf{\Gamma}]^*. \quad (3.11)$$

This procedure can be applied in the same way to an impulse response of the adjoint system, yielding an improved approximation of the observability Gramian.

3.3.3 Real formulation

In some applications, it may not be desirable to append the snapshot matrix with complex-valued vectors, as is done in (3.11). For instance, in an application where the state is always real-valued (often the case in numerical simulations), post-processing codes for computing empirical Gramians or BPOD modes may already exist, but may not be equipped to deal with complex-valued vectors. While we must typically consider complex-valued vectors when computing the eigenvector matrix \mathbf{V}_M , this process will in general be handled by a different code than the one that computes Gramians or BPOD modes. As such, a real factorization of $\mathbf{V}_M\mathbf{N}\mathbf{V}_M^*$ may be desirable. (In general, the elements of \mathbf{V}_M and \mathbf{N} are complex.)

¹Certain matrix libraries (e.g., the one used by Matlab) will not compute the Cholesky factorization of \mathbf{N} if it is not positive definite. In this case, one can compute unitary diagonalization $\mathbf{N} = \mathbf{U}_N\mathbf{\Lambda}_N\mathbf{U}_N^*$ and let $\mathbf{\Gamma} = \mathbf{U}_N\mathbf{\Lambda}_N^{1/2}$.

We break \mathbf{V}_M and \mathbf{N} into their real and imaginary parts:

$$\begin{aligned}\mathbf{V}_M &= \mathbf{V}_M^{\text{Re}} + i\mathbf{V}_M^{\text{Im}} \\ \mathbf{N} &= \mathbf{N}^{\text{Re}} + i\mathbf{N}^{\text{Im}}.\end{aligned}$$

For a real-valued system, the product $\mathbf{V}_M\mathbf{N}\mathbf{V}_M^*$ must also be real-valued, so we can simply collect the real terms in computing

$$\begin{aligned}\mathbf{V}_M\mathbf{N}\mathbf{V}_M^* &= (\mathbf{V}_M^{\text{Re}} + i\mathbf{V}_M^{\text{Im}}) (\mathbf{N}^{\text{Re}} + i\mathbf{N}^{\text{Im}}) (\mathbf{V}_M^{\text{Re}} - i\mathbf{V}_M^{\text{Im}})^T \\ &= \mathbf{V}_M^{\text{Re}}\mathbf{N}^{\text{Re}}(\mathbf{V}_M^{\text{Re}})^T + \mathbf{V}_M^{\text{Re}}\mathbf{N}^{\text{Im}}(\mathbf{V}_M^{\text{Im}})^T - \mathbf{V}_M^{\text{Im}}\mathbf{N}^{\text{Im}}(\mathbf{V}_M^{\text{Re}})^T + \mathbf{V}_M^{\text{Im}}\mathbf{N}^{\text{Re}}(\mathbf{V}_M^{\text{Im}})^T \\ &= \begin{bmatrix} \mathbf{V}_M^{\text{Re}} & \mathbf{V}_M^{\text{Im}} \end{bmatrix} \begin{bmatrix} \mathbf{N}^{\text{Re}} & \mathbf{N}^{\text{Im}} \\ -\mathbf{N}^{\text{Im}} & \mathbf{N}^{\text{Re}} \end{bmatrix} \begin{bmatrix} \mathbf{V}_M^{\text{Re}} & \mathbf{V}_M^{\text{Im}} \end{bmatrix}^T.\end{aligned}\quad (3.12)$$

(The imaginary terms can be shown to equal the zero matrix individually, if one considers the form of \mathbf{N} itself, as well as the fact that the columns of \mathbf{V}_M come in conjugate pairs for a real-valued system.)

We recall from (3.9) that

$$\mathbf{N}_{j,k} = -\frac{1}{\lambda_j + \bar{\lambda}_k}.$$

Since (3.1) is a stable system, we let $\lambda_j = -\alpha_j + i\beta_j$, with $\alpha_j > 0$. Then

$$\mathbf{N}_{j,k} = \frac{\alpha_j + \alpha_k}{(\alpha_j + \alpha_k)^2 + (\beta_j - \beta_k)^2} + i\frac{\beta_j - \beta_k}{(\alpha_j + \alpha_k)^2 + (\beta_j - \beta_k)^2},$$

giving us

$$\begin{aligned}\mathbf{N}_{j,k}^{\text{Re}} &= \frac{\alpha_j + \alpha_k}{(\alpha_j + \alpha_k)^2 + (\beta_j - \beta_k)^2} \\ \mathbf{N}_{j,k}^{\text{Im}} &= \frac{\beta_j - \beta_k}{(\alpha_j + \alpha_k)^2 + (\beta_j - \beta_k)^2}.\end{aligned}$$

From this we see that \mathbf{N}^{Re} is symmetric and \mathbf{N}^{Im} is skew-symmetric, making

$$\mathbf{Q} = \begin{bmatrix} \mathbf{N}^{\text{Re}} & \mathbf{N}^{\text{Im}} \\ -\mathbf{N}^{\text{Im}} & \mathbf{N}^{\text{Re}} \end{bmatrix} = \begin{bmatrix} \mathbf{N}^{\text{Re}} & \mathbf{N}^{\text{Im}} \\ (\mathbf{N}^{\text{Im}})^T & \mathbf{N}^{\text{Re}} \end{bmatrix}$$

a real, symmetric matrix. One can show that if $[(\mathbf{v}^{\text{Re}})^T \ (\mathbf{v}^{\text{Im}})^T]^T$ is an eigenvector of \mathbf{Q} , then $\mathbf{v}^{\text{Re}} - i\mathbf{v}^{\text{Im}}$ is an eigenvector of \mathbf{N} with the same eigenvalue. Thus the eigenvalues of \mathbf{Q} are a subset of those of \mathbf{N} , which are all real and non-negative, and \mathbf{Q} is positive semi-definite. Letting $\mathbf{Q} = \mathbf{R}\mathbf{R}^T$ be the Cholesky factorization of \mathbf{Q} ,² (3.12) can be rewritten as

$$\mathbf{V}_M\mathbf{N}\mathbf{V}_M^* = \left(\begin{bmatrix} \mathbf{V}_M^{\text{Re}} & \mathbf{V}_M^{\text{Im}} \end{bmatrix} \mathbf{R} \right) \left(\begin{bmatrix} \mathbf{V}_M^{\text{Re}} & \mathbf{V}_M^{\text{Im}} \end{bmatrix} \mathbf{R} \right)^T,$$

²As with \mathbf{N} , if necessary, a unitary diagonalization can be used to factor \mathbf{Q} instead of a Cholesky factorization.

and the controllability Gramian (see (3.10)) as

$$\mathbf{W}_c = \begin{bmatrix} \mathbf{X}_T & [\mathbf{V}_M^{\text{Re}} & \mathbf{V}_M^{\text{Im}}] \mathbf{R} \end{bmatrix} \begin{bmatrix} \mathbf{X}_T & [\mathbf{V}_M^{\text{Re}} & \mathbf{V}_M^{\text{Im}}] \mathbf{R} \end{bmatrix}^T, \quad (3.13)$$

a product of real matrices.

3.3.4 Application to BPOD

While (3.11) was derived for an impulse response of the direct system (3.1), the same method can be applied to an impulse response of the adjoint system (3.4). Applying the analytic tail method to both sets of impulse responses, we can factor the controllability and observability Gramians as

$$\begin{aligned} \mathbf{W}_c &= [\mathbf{X}_T \quad \mathbf{V}_M^c \Gamma^c] [\mathbf{X}_T \quad \mathbf{V}_M^c \Gamma^c]^* \\ \mathbf{W}_o &= [\mathbf{Y}_T \quad \mathbf{V}_M^o \Gamma^o] [\mathbf{Y}_T \quad \mathbf{V}_M^o \Gamma^o]^*. \end{aligned}$$

For BPOD, we construct the Hankel matrix by multiplying factors of the controllability and observability matrices. With the analytic tail, this gives us

$$\mathbf{H} = [\mathbf{Y}_T \quad \mathbf{V}_M^o \Gamma^o]^* [\mathbf{X}_T \quad \mathbf{V}_M^c \Gamma^c]. \quad (3.14)$$

From here, the rest of the BPOD algorithm is the same as that described in Section 3.2.3.

3.4 Snapshot-based eigenvector estimation using DMD

The analytic tail method described in Section 3.3 requires a knowledge of certain eigenvalues and eigenvectors of \mathbf{A} . For a very large system, computing the eigenvectors of \mathbf{A} directly may be numerically intractable. In some cases, for instance in fluid simulations, an explicit representation of \mathbf{A} is not even available. Iterative methods such as the Arnoldi algorithm provide a means for estimating the eigenvalues and eigenvectors of large systems, taking a “black box” approach that requires only the ability to evaluate the matrix-vector product $\mathbf{A}\mathbf{x}$. However, for our purposes Arnoldi-like methods are less than ideal. In addition to requiring additional simulations, which may be expensive, they typically estimate the eigenvectors of \mathbf{A} whose corresponding eigenvalues lie on the periphery of the spectrum [166], whereas we are only interested in those that dominate the impulse response tail. For instance, the slowest eigenvalue of \mathbf{A} may correspond to an eigenvector that is not excited by the impulse at all.

Instead, we turn to DMD, discussed in detail in Chapter 2. Since our data are generated by a linear system, the DMD modes and eigenvalues will approximate the eigenvectors and eigenvalues of the underlying dynamics. For the analytic tail method, we can run an impulse response simulation until a small number of eigenvectors begins to dominate the state, at which point we stop the simulation. (This cut-off can be detected, for example, by plotting the norm of the state and waiting until only a few frequencies dominate the signal.) DMD modes can then be computed from a small number of snapshots collected at the end of the impulse response.

By applying DMD to these later snapshots, we eliminate the need for any additional simulations and guarantee that only those eigenvectors with a measurable presence in the tail are estimated.

The number of snapshots necessary for such a DMD computation depends on the number of eigenvectors that are active in the impulse response tail. At minimum, the rank of the snapshot set must be equal the number of eigenvectors to be estimated. Since this number is assumed to be small (a simple tail is assumed), the DMD computation is quite cheap. Furthermore, there is no benefit in using additional snapshots if they do not increase the rank of the snapshot set. In a simple impulse response tail, all snapshots will be linear combinations of the same few eigenvectors, so there is no need to extend the impulse response past the beginning of the tail.

It was shown by Rowley et al. [138] that if one uses DMD to estimate eigenvalues λ_j and eigenvectors \mathbf{v}_j from a set of linearly independent snapshots $\{\mathbf{k}_j\}_{j=0}^N$, the modes can be scaled such that

$$\mathbf{k}_j = \sum_{k=1}^N \lambda_k^j \mathbf{v}_k \quad j = 0, \dots, N-1. \quad (3.15)$$

(For more details, see Section 2.5.) Thus the norm of each mode gives some indication of its contribution to a given snapshot. For example, the first snapshot is simply equal to the sum of the DMD modes:

$$\mathbf{k}_0 = \sum_{k=1}^N \mathbf{v}_k. \quad (3.16)$$

As such, in addition to an estimate of the eigenvectors and eigenvalues that dominate the impulse response tail, DMD analysis also provides us with a way to quantify the relative importance of each eigenvector/eigenvalue pair, based on the norm $\|\mathbf{v}_k\|$. This can be used to determine how many eigenvectors are necessary to characterize an impulse response tail.

3.5 Results and discussion

3.5.1 Computing the controllability Gramian

Here we present two examples that demonstrate the effectiveness of the analytic tail method in computing empirical Gramians. In each, we compute the impulse response of a real system $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$, collecting snapshots of the state \mathbf{x} every $\Delta t = 0.01$. The empirical controllability Gramian is first computed using what we will refer to as the “standard” method. For varying T , we stack snapshots spanning the interval $t = [0, T]$ as columns of a matrix, using a uniform quadrature weight $\sqrt{\Delta t}$:

$$\mathbf{X}_T = [\mathbf{x}(0) \quad \mathbf{x}(\Delta t) \quad \dots \quad \mathbf{x}(T)] \sqrt{\Delta t}.$$

The empirical Gramian is then given by $\mathbf{W}_c = \mathbf{X}_T^* \mathbf{X}_T$. For the analytic tail method, we use DMD to compute the slow eigenvalues and eigenvectors and form the matrices

\mathbf{V} and $\mathbf{\Gamma}$ as in (3.11). The modified snapshot matrix is then

$$\mathbf{X} = [\mathbf{X}_T \quad \mathbf{V}\mathbf{\Gamma}]$$

and the controllability Gramian is given by $\mathbf{W}_c = \mathbf{X}^*\mathbf{X}$. We compare each of these computations against the controllability Gramian as computed using Matlab. The error is measured using the Frobenius matrix norm:

$$\|\Delta\mathbf{W}_c\|_2 = \left[\sum_j \sum_k \left(\mathbf{W}_{c_{j,k}}^{(\text{matlab})} - \mathbf{W}_{c_{j,k}}^{(\text{empirical})} \right)^2 \right]^{1/2}. \quad (3.17)$$

Non-normal 3×3 system

In our first example, we consider the system (3.1) with

$$\mathbf{A} = \begin{bmatrix} -1 & 0 & 100 \\ 0 & -2 & 100 \\ 0 & 0 & -5 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}. \quad (3.18)$$

(This system was first used as a test for model reduction techniques by Holmes et al. [78, Section 5.6.1].) Though this system is stable, it exhibits non-normal transient growth before undergoing exponential decay (Figure 3.1, left). This non-normality is caused by the fact that the fast-decaying eigenvector $[-0.6 \quad -0.8 \quad 0.02]^T$ is nearly parallel to the span of the other two eigenvectors, $[1 \quad 0 \quad 0]^T$ and $[0 \quad 1 \quad 0]^T$. Using the standard method, we must sample the impulse response to $T = 6$ before \mathbf{W}_c converges to its final value (Figure 3.1, right). In contrast, if we estimate the slow eigenvector and treat the tail analytically, we observe convergence in \mathbf{W}_c by $T = 4$. (For a given simulation length T , the last four snapshots are used for the DMD computation.)

In addition to the 33 % reduction in required storage space for snapshots, we see that the analytic tail method produces a more accurate controllability Gramian for all T , up to the point where both methods have converged. That the same amount of

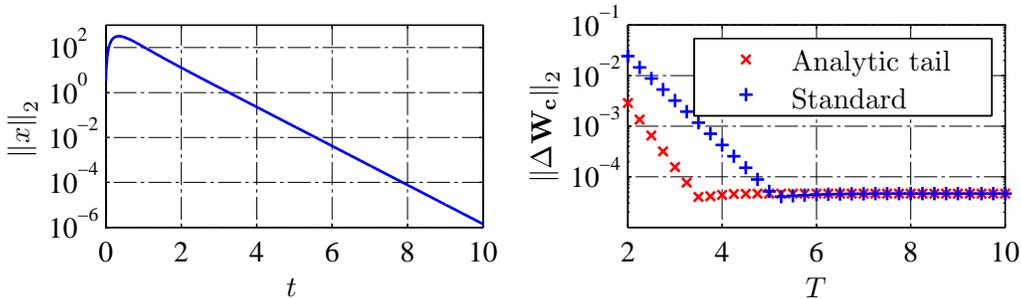


Figure 3.1: (Left) Non-normal transient growth in a 3×3 linear system. The slow decay is the result of a single slow eigenvector. (Right) Error in computing the controllability Gramian empirically. With the analytic tail method, convergence is achieved with approximately 33% fewer snapshots than is required for the standard method.

error is eventually observed for both methods is expected and encouraging. Though the analytic tail method should increase accuracy by accounting for the truncated snapshots, for a stable impulse response the contribution of those snapshots will eventually be negligible. Thus for T large enough, the two methods should produce nearly identical results. The observed agreement suggests that the analytic tail method is indeed enhancing accuracy in the manner intended. The remaining error in computing \mathbf{W}_c is due to the fact that we are using snapshots to compute the Gramian empirically, and can be reduced by sampling the impulse response faster and/or using higher-order quadrature weights.

Pseudorandom 100×100 system

For a larger example, we construct a pseudorandom, 100×100 matrix \mathbf{A} using the following procedure:

1. Start with a matrix of zeroes. Place ten stable, slowly decaying oscillators of the form

$$\begin{bmatrix} -\alpha & \beta \\ -\beta & -\alpha \end{bmatrix}$$

along the diagonal of \mathbf{A} . The values α and β are chosen randomly subject to the restrictions $\alpha \in (0, 1]$ and $\beta \in [0, 10]$.

2. Place up to 40 stable, fast-decaying oscillators (same form as above) along the diagonal of \mathbf{A} . The number of fast-decaying oscillators is chosen randomly, modulo the restrictions $\alpha \in (1, 5]$ and $\beta \in [0, 10]$.
3. Place ten slowly decaying, stable, real eigenvalues along the diagonal of \mathbf{A} . These eigenvalues are of the form $\lambda = -\alpha$ with $\alpha \in (0, 1]$.
4. The rest of the entries on the diagonal are filled with fast-decaying, real eigenvalues $\lambda = -\alpha$ with $\alpha \in (1, 5]$.
5. Fill in the upper triangular portion of \mathbf{A} with random values lying in the interval $[0, 0.25]$.

By constructing \mathbf{A} in this way, we are able to specify its eigenvalues, guaranteeing a stable system with oscillatory dynamics and multiple timescales of interest.

Here we consider a particular choice of \mathbf{A} with a much more complex impulse response than was seen in the 3×3 example discussed previously. After an initial period of non-normal transient growth, the system simultaneously decays and oscillates (Figure 3.2). The decay rate is fairly constant from $t = 20$ to $t = 100$, though there is growing evidence of multifrequency interaction, in the form of beating (amplitude modulation). Around $t = 100$, the decay rate begins to slow down and the presence of beating is clear. The beating behavior begins to fade as the decay rate slows down to its final value, and by $t = 200$, it appears that we have returned to a constant decay rate and oscillation at a single, fixed frequency.

Using DMD analysis, we can corroborate this behavior. We consider impulse responses ending at $T = 50$, $T = 150$ and $T = 250$. For each case, we use the last 20 snapshots of the simulation for DMD. At $T = 50$, the spectrum is dominated by a real eigenvector with a decay rate $\alpha = 0.046$ (Figure 3.3, left). The beating,

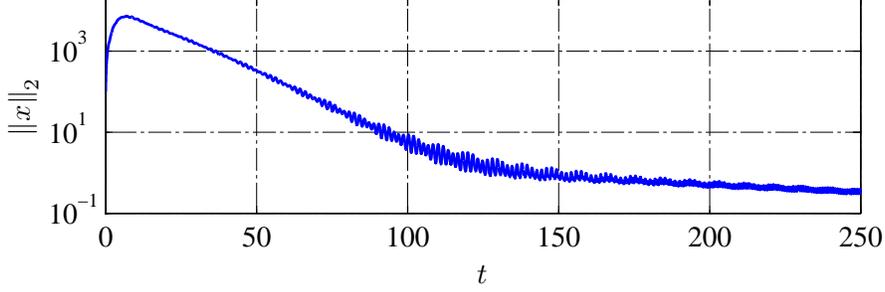


Figure 3.2: Impulse response of a pseudorandom 100×100 system. Non-normal transient growth is followed by simultaneous oscillation and decay. Around $t = 120$ there is a clear change in the decay rate, as well as evidence of multifrequency interaction, in the form of beating (amplitude modulation). Eventually, the effect of other frequencies decays quickly, with only minimal evidence of beating past $t = 200$.

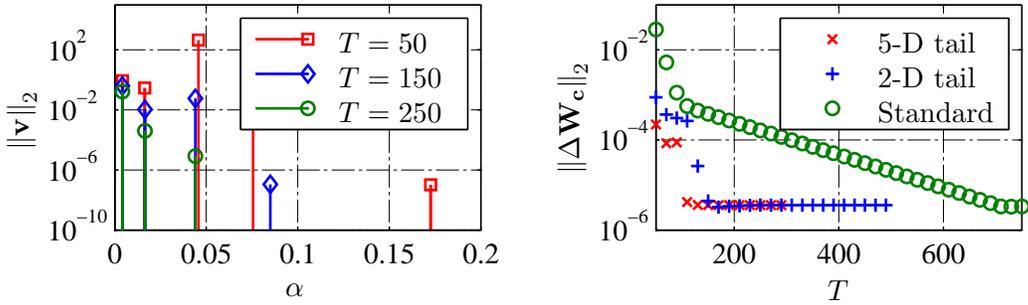


Figure 3.3: (Left) DMD spectra for the impulse response of a pseudorandom 100×100 system. The norms of the estimated eigenvectors are plotted against the corresponding decay rates. For $T = 50$ the dominant decay rate corresponds to a real eigenvector at $\alpha = 0.046$ whereas at $T = 150$ it corresponds to a complex conjugate pair at $\alpha = 0.004$. The beating observed in the impulse response results from the interaction of the $\alpha = 0.004$ pair with another complex conjugate eigenvector pair at $\alpha = 0.016$. (Right) Error in computing the controllability Gramian empirically for a pseudorandom 100×100 system. As predicted by DMD analysis, a five-dimensional analytic tail leads to convergence by $T = 150$, while a two-dimensional tail converges at $T = 250$. In contrast, without an analytic tail, the empirical controllability Gramian does not converge until $T > 700$. By applying the analytic tail method we achieve a savings of 65% (two-dimensional tail), or even 79% (five-dimensional tail), in storage space for snapshots.

oscillatory behavior is caused by the interaction of two pairs of complex conjugate eigenvectors, at $\alpha = 0.004$ and $\alpha = 0.016$. For $T = 150$, the spectrum is instead dominated by the complex conjugate pair at $\alpha = 0.004$, corresponding to the change in decay rate discussed previously. The eigenvector pair at $\alpha = 0.016$ still has a significant, though reduced, norm here, corresponding to the reduced evidence of beating. Once we reach $T = 250$, the slow eigenvector pair at $\alpha = 0.004$ completely dominates the DMD spectrum, corresponding to the constant exponential decay and single frequency oscillation observed at the end of the impulse response.

Motivated by this DMD analysis, we compute the empirical controllability Gramian using a five-dimensional and two-dimensional analytic tail. (The DMD spectra suggest that five and two eigenvectors should accurately describe the impulse tails at $T = 150$ and $T = 250$, respectively.) Indeed, we see that for the five-dimensional tail, the controllability Gramian converges by $T = 150$ (Figure 3.3, right). With a two-dimensional tail, the computation converges by $T = 250$. Surprisingly, a standard computation of \mathbf{W}_c does not converge until we pass $T = 700$. This is despite the fact that past $T = 250$ the impulse response is dominated by a single pair of eigenvectors. The surprisingly slow convergence of the standard method highlights the fact that for many systems, an impulse response must be sampled well into the tail before convergence is achieved, even if by that point the dynamics are very simple.

3.5.2 Model reduction

The analytic tail method is especially useful for BPOD, as there are tails associated with both the direct and adjoint impulse responses. We present two examples demonstrating the benefits of the method. First, we consider the complex Ginzburg-Landau (CGL) equation, commonly used as a model for fluid flows. A discretization of the CGL dynamics yields a system that can be analyzed directly using numerical packages such as Matlab, allowing us to compare the models derived using BPOD against those generated from exact balanced truncation. We then consider the two-dimensional flow past a cylinder, computed using an immersed boundary method (see Section 2.4.3 for details) [36, 158]. This is a much larger computation and clearly demonstrates the savings achieved with the analytic tail method, as well as its applicability for the types of large systems that are likely to be encountered in practice. Unfortunately, due to the size of the problem, exact balanced truncation cannot be performed, and as such, we use convergence tests to compare the results of BPOD with and without analytic tails. This is in contrast to the CGL system, for which a direct comparison to balanced truncation is done.

Linearized complex Ginzburg-Landau (CGL) system

The linearized CGL equation is given by

$$\dot{q} = -\nu \frac{\partial q}{\partial x} + \mu(x)q + \gamma \frac{\partial^2 q}{\partial x^2}. \quad (3.19)$$

The evolution of q can be thought of as a model for the growth and decay of a velocity perturbation in a fluid flow [32, 37]. (For a control-oriented review of the CGL equation, see [16].) To put (3.19) in the state-space form (3.1), we discretize as described by Chen and Rowley [30]. We choose a state dimension $n = 100$, which is large enough to accurately represent (3.19) but small enough to perform exact balanced truncation (using Matlab), which we use as a reference for our empirical methods. We choose a subcritical value $\mu_0 = 0.38$ (for which the linearized dynamics are stable), place a single actuator at $x = -1$, and place a single sensor at $x = 1$. All other parameters are set to the default values used by Chen and Rowley [30], which were drawn from Bagheri et al. [16].

The direct impulse response initially decays before undergoing non-normal transient growth, with $\|x\|_2$ reaching a peak around $t = 11$ (Figure 3.4, left). Past this point, there is exponential decay at a constant rate. Looking at the real part of the state, we can see that the state also oscillates, with a single, fixed frequency. As such, we use a single, complex eigenvector to describe the direct impulse response tail. We assume that the same can be done for the adjoint impulse response.

To form reduced-order models of (3.19), we collect snapshots of the direct and adjoint impulse responses every $\Delta t = 0.01$. We vary the truncation point T from 20 to 100, at each point using the last 20 snapshots for a DMD computation of the slow eigenvector and eigenvalue, direct and adjoint respectively. The snapshots are scaled with fourth-order quadrature weights [130] so that the quadrature sum (3.3) more accurately approximates the integral expression (3.2). BPOD modes are then computed, both with and without an analytic tail. We project the dynamics (3.19) onto these modes to get reduced-order models.

Figure 3.4 (right) shows the error in computing 10-state reduced-order models of the linearized CGL equation, as a function of T . In addition to comparing the models to each other, we also compare the transfer function errors to the analytic bounds given by (3.5) and (3.6). We recall that because BPOD is only an approximation of balanced truncation, the theoretical upper bound (3.6) may not be satisfied. This is indeed the case for the standard BPOD models, up to $T = 85$. In contrast, the analytic tail models meet this criterion as early as $T = 25$. To check that this is not a peculiarity for a model order $r = 10$, we fix T and plot the transfer function error as a function of the model order r . Figure 3.5 shows that with $T = 25$, the analytic tail models meet the theoretical upper bound for all model orders,

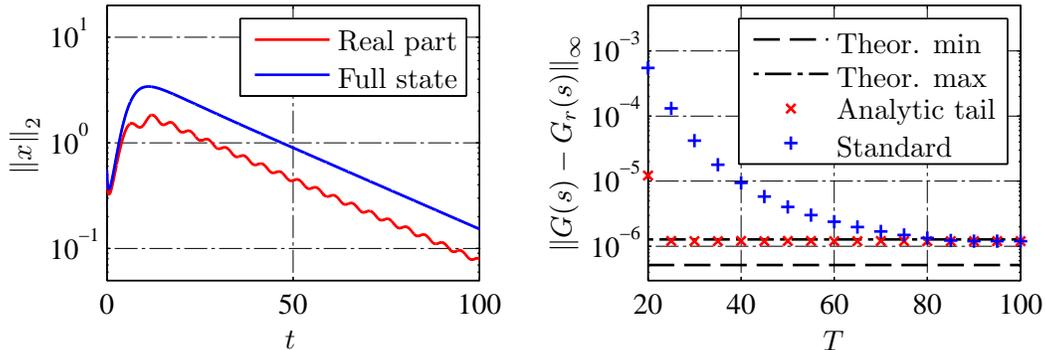


Figure 3.4: (Left) Impulse response of the CGL equation. An initial decay in the energy of the system is followed by non-normal transient growth and an eventual exponential decay. The real part of the state shows similar behavior, with oscillation at a single, fixed frequency during the exponential decay phase. This suggests that a single, complex eigenvector dominates the tail. (Right) Transfer function error for 10-state reduced-order models of the CGL equation, as a function of impulse response simulation length. All models are computed using BPOD. Without an analytic tail, the models do not converge until $T = 85$. With an analytic tail method, convergence is achieved at $T = 25$, resulting in a drastic reduction in both storage space and computation time.

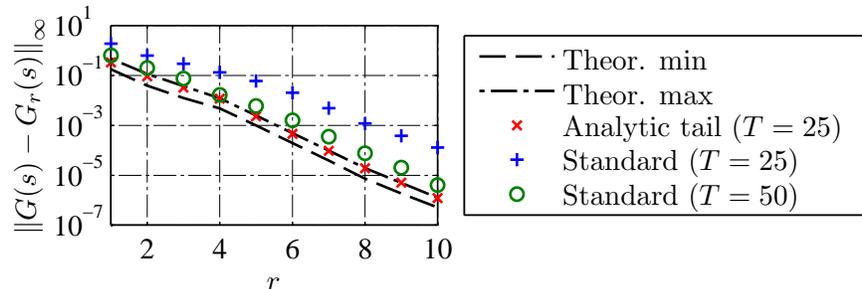


Figure 3.5: Transfer function error as a function of model order. All models are computed using BPOD. With an analytic tail applied at $T = 25$, the theoretical error bounds for balanced truncation are met for all model orders. With the same T , standard BPOD models fail to meet the error bounds for any order. As we increase T to 50, we see that the performance of the standard models begins to approach that of the analytic tail models, as expected.

while the standard BPOD models fail to meet the theoretical upper bound for any. However, as expected, with increasing T the performance of the standard BPOD models (for all model orders) begins to approach that of the analytic tail models. For $T = 100$, the errors are nearly indistinguishable (not pictured, for clarity) and closely approximate the error for exact balanced truncation (within 8 % for the cases shown).

Linearized two-dimensional cylinder flow

To investigate the flow past a two-dimensional cylinder, we use the numerical method and flow configuration described in Section 2.4.3. With a cylinder diameter of 1 and a Reynolds number of 100, the flow is globally unstable. This instability leads to an oscillatory wake, where vortices alternately shed from the upper and lower shear layers, yielding the familiar Kármán vortex street. As the vortices shed, they generate unsteady forces on the cylinder, which can be undesirable. To eliminate these oscillations, we can design and implement feedback controllers based on reduced-order models. Here we will investigate the benefits of using the analytic tail method in constructing such models using BPOD.

The BPOD computation requires simulations of the direct and adjoint dynamics, which are based on a linearization of (2.16), the immersed boundary formulation of the Navier–Stokes equations. (For details on this linearization, see the article by Ahuja and Rowley [4].) To simulate these dynamics, we must first identify the unstable equilibrium. We do so using selective frequency damping [6], yielding the steady solution shown in Figure 3.6. Furthermore, because BPOD can be applied only to stable systems, we must also decouple the stable and unstable dynamics. For the cylinder flow at $\text{Re} = 100$, the direct and adjoint systems each have a single pair of unstable global modes, which we compute using a standard Arnoldi iteration [166]. These global modes are shown in Figure 3.7 and are used to project the linearized and adjoint dynamics onto their stable subspaces, respectively. It is these restricted, stable dynamics on which we perform BPOD.

To control the cylinder wake, we actuate the flow using a vertical velocity perturbation supported on a disk located downstream of the cylinder. The forcing covers

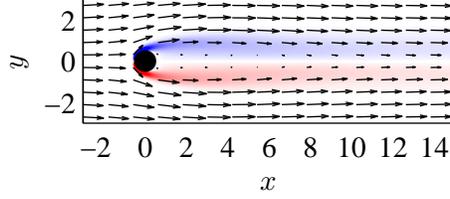


Figure 3.6: Unstable equilibrium for the flow past a cylinder at $Re = 100$. The flow field is depicted using contours of vorticity overlaid with velocity vectors.

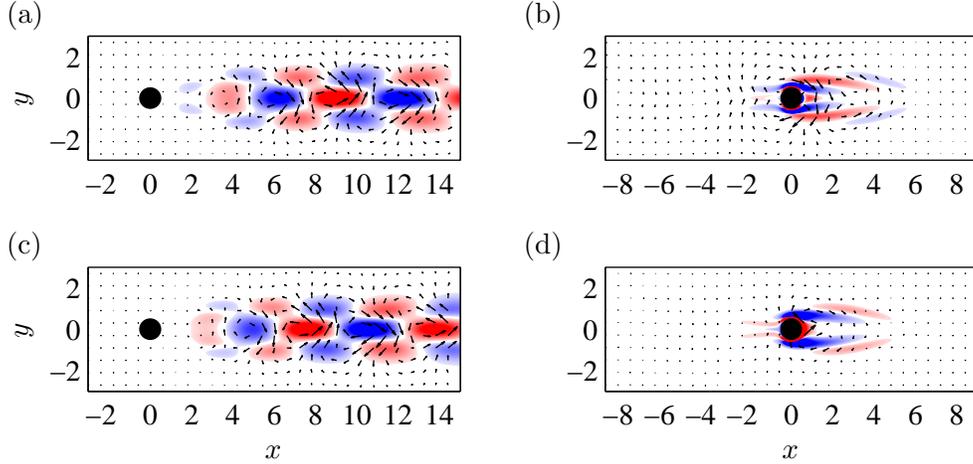


Figure 3.7: Unstable global modes for the two-dimensional cylinder flow at $Re = 100$. Flow fields are depicted using contours of vorticity overlaid with velocity vectors. (a) Direct system, real part; (b) adjoint system, real part; (c) direct system, imaginary part; (d) adjoint system, imaginary part.

a spatial region equal in size to the body, and is placed two diameters downstream (Figure 3.8). This choice of actuation is based on the work of Noack et al. [116], which can be used as a benchmark for cylinder control applications. Though it does not model a physical actuator, the volume forcing is a convenient choice for this example as it is easy to implement and has an obvious effect on the wake. The output signals used for feedback control are collected using point sensors placed at $x = 2, 3$, and 6 . Each sensor measures only the vertical component of the velocity. (This flow configuration is the same as the one described in Section 2.4.3.)

The direct impulse response for this system, restricted to the stable subspace, is qualitatively similar to that of the CGL system. Figure 3.9 shows that there is an initial period of non-normal growth during which the norm of the state grows by over four orders of magnitude. This is followed by a relatively slow decay. After 1200 convective times (60,000 time steps) the state is still over six times as energetic as the initial condition. During the initial period of decay ($t \in [100, 300]$), there are slow oscillations in the kinetic energy. While the slow oscillations die out, there are also fast oscillations that are present through the end of the impulse response (see the enlarged inset in Figure 3.9). By $t = 500$, this fast frequency is the only oscillatory behavior that can be observed. All other oscillatory behavior has died away. This,

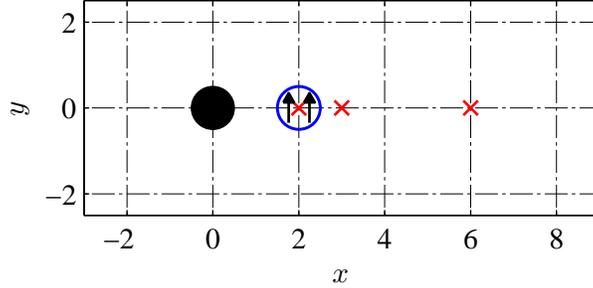


Figure 3.8: Schematic of input and output for two-dimensional cylinder flow. Actuation is implemented as a disk of vertical force two cylinder diameters downstream of the body (blue, \circ). Point sensors measuring the vertical velocity are placed at $x = 2, 3,$ and 6 (red, \times).

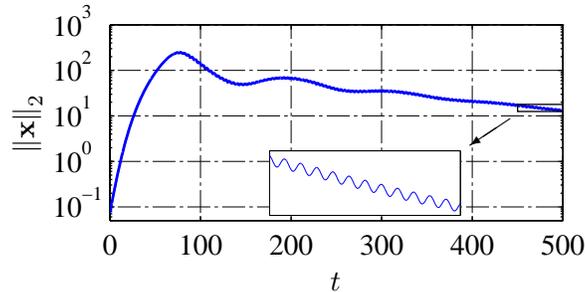


Figure 3.9: Kinetic energy in stable impulse response for two-dimensional cylinder flow. There is an initial period of non-normal growth followed by a slow decay. By $t = 500$, the decay rate is linear (on a log scale) with a single oscillation frequency, suggesting that only one complex-conjugate pair of eigenvectors is active.

in addition to the fact that the decay rate is perfectly logarithmic, suggests that the remainder of the impulse response can be modeled using an analytic tail.

To test this hypothesis, we compute a series of reduced order models using BPOD, with and without an analytic tail. For such a high-dimensional system, we cannot compute an exact balanced truncation, so we instead check for convergence. As more snapshots are used in the standard BPOD computations, more and more of the long-time behavior is captured, and the models should converge. If the analytic tail method is correctly capturing the long-time behavior, then the models computed using an analytic tail will converge to the same answer, but using fewer snapshots.

We run our direct and adjoint impulse response simulations to $t = 1200$, collecting a snapshot every 50 timesteps (once every convective time unit). (Collecting snapshots at this rate resolves the fastest frequency observed in the impulse responses.) All of these snapshots are used to compute a BPOD model of order 16, using Riemann sum approximations for all integrals. We take this model as the best approximation of exact balanced truncation. Figure 3.10 (left) shows that the output predicted by this model does in fact match the output from the full simulation, validating this approximation. (A close inspection reveals small discrepancies between simulation and model outputs. However, these are expected and result from the fact that in this multi-domain scheme, the Laplacian operator is not self-adjoint to numerical precision, as described by Ahuja and Rowley [4].)

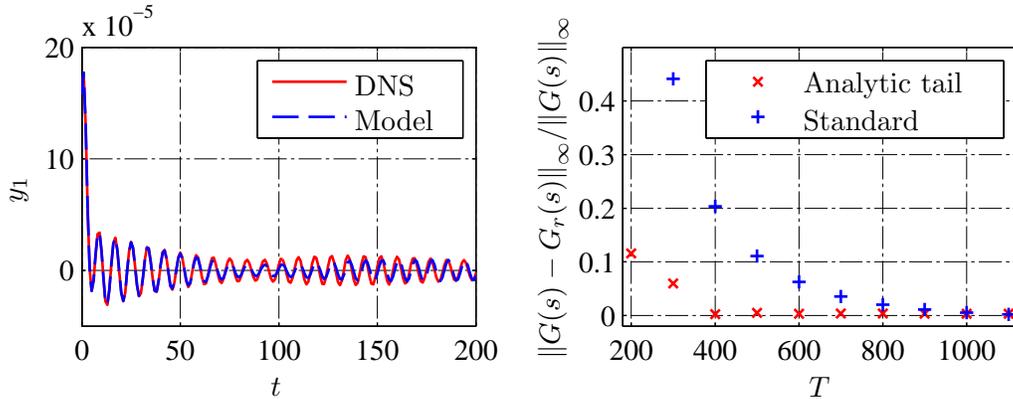


Figure 3.10: (Left) Comparison of true and predicted impulse response outputs. The vertical velocity at $x = 2$ is measured in the full simulation and compared to the output predicted by a BPOD model computed using snapshots collected up to $T = 1200$. The agreement shows that the $T = 1200$ model is converged and accurately captures the physics of the flow. (Right) Transfer function error for 16-order models of the two-dimensional cylinder flow, as a function of impulse response simulation length. Convergence is checked against the $T = 1200$ BPOD model. The standard BPOD models converge relatively slowly, whereas the models computed using a two-dimensional analytic tail have converged to the $T = 1200$ solution as soon as $T = 400$. (For $T = 200$, the error in the standard computation is 4.26. This point is omitted from the plot for clarity.)

We then compute 16-state models with direct and adjoint impulse responses truncated at various $T < 1200$. Each of these models is compared to the $T = 1200$ model to check for convergence. The results of this analysis are shown in Figure 3.10 (right). As before, for each choice of T , using the analytic tail method improves the accuracy of the model. Furthermore, we see that the models computed with an analytic tail converge much faster than those computed without. With an analytic tail of dimension two, snapshots only need to be collected up to $T = 400$. The computation of the eigenvectors dominating the tail is fairly cheap, requiring a DMD computation using only the last seven available snapshots. Further savings could potentially be achieved by considering more vectors in the tail, at little additional cost.

Table 3.1 gives a quantitative summary of the savings achieved by implementing the analytic tail method. The computation time is dominated by the impulse responses (one direct, three adjoint), which are each done in serial. Using analytic tails, we get a linear speedup in the simulation time (67 % savings), which for this particular computation corresponds to a savings of nearly 300 CPU hours. In computing the Hankel matrix, we achieve a savings of nearly 85 %, or about 70 CPU hours. While the absolute savings in this step is smaller, it scales roughly quadratically. This is critical, as constructing the Hankel matrix can easily dominate the computation time. For instance, using a parallel solver could decrease the simulation time, while large datasets and/or larger snapshot ensembles would increase the cost of assembling the Hankel matrix. Computing the SVD of the Hankel matrix (97 %

Table 3.1: BPOD costs for two-dimensional cylinder flow (in CPU hours).

Task	Standard	Analytic tail	Savings
Impulse response simulations *	447.6	148.0	67.0%
DMD for analytic tail	—	0.24	—
Constructing Hankel matrix	83.57	13.24	84.2%
SVD of Hankel matrix	0.093	0.003	97.0%
Constructing modes	34.85	11.76	66.3%
Total	566.1	173.2	69.4%

* Simulations run to $T = 1200$ for standard method, $T = 400$ with analytic tail.

savings) scales cubically, but the SVD time is such a small part of the total cost that this savings is insignificant. Finally, we also achieve a linear speedup (66 %, 33 CPU hours) in constructing the BPOD modes, which is a linear operation. In total, by implementing the analytic tail method, we save nearly 400 CPU hours without any sacrifice in model accuracy.

3.6 Conclusions

There are many features of DMD that can be leveraged for gain, depending on the particular application. In this chapter, we focused on the fact that DMD is a snapshot-based algorithm. We used this fact to efficiently compute eigendecompositions of linear systems. Specifically, we presented a method for analytically treating the tail of an impulse response, improving accuracy and efficiency when computing empirical Gramians or using BPOD to compute reduced-order models. When the long-term behavior of an impulse response is governed by a small number of eigenvectors, we can account for the effect of these eigenvectors on the empirical Gramian analytically. In doing so, we no longer need to sample the impulse response past the beginning of the tail. This lowers the storage requirement for snapshots and speeds up ensuing computations. These effects are especially useful for BPOD, as benefits are gained in treating both the direct and adjoint impulse responses this way. By using DMD to estimate the eigenvectors that dominate the tail, we minimize the additional cost in applying the analytic tail method, requiring no additional simulations.

These methods were applied to number of examples, demonstrating their effectiveness. For two linear systems, the analytic tail method was used to aid in computing the controllability Gramian empirically. It was also used to more efficiently compute reduced-order models of the linearized, complex Ginzburg-Landau equation and the linearized flow past a two-dimensional cylinder at a Reynolds number of 100. In all cases, the use of an analytic tail produced highly accurate results with significantly fewer snapshots than would be required otherwise. The controllability Gramians and BPOD-based models, respectively, converged to the values that were obtained when the impulse responses were sampled far into their tails. These examples verify that the analytic tail method correctly accounts for the long-term behavior of the impulse response tails with little additional cost.

In a sense, computing approximate eigendecompositions is the most direct and well-understood application of DMD. As an eigendecomposition tool applied to sequential snapshots, DMD can be treated using the theory of Krylov subspaces, much like the Arnoldi iteration [138, 141, 166]. This requires very little of the more general framework discussed in Chapter 2. The work described in the present chapter shows that even this restriction of DMD to linear systems can be quite useful in practice. In the next chapter, we explore the application of DMD to nonlinear systems, making use of its full generality.

Chapter 4

Application to oscillatory fluid flows

Dynamic mode decomposition (DMD) is often said to identify modes based on their frequency content. This is because each DMD mode is associated with a corresponding DMD eigenvalue, and thus a growth/decay rate and possibly an oscillation frequency too (if the eigenvalue is complex). When DMD is applied to analyze the time evolution of fluid flows, the modes are spatial structures whose temporal dynamics are determined by their corresponding eigenvalues. As such, the flow is decomposed into spatial structures that each oscillate at a single temporal frequency.

This makes DMD a useful tool for analyzing fluid flows that exhibit oscillatory dynamics. For such flows, DMD identifies spatial structures that correspond to characteristic flow frequencies. The structures often highlight coherent structures of interest, which may elucidate the underlying dynamics of the flow. In this chapter we present two such applications of DMD. First, we analyze a model for separated airfoil flows, which are characterized by a complex interaction between multiple flow frequencies. Then we discuss DMD analysis of a shock-turbulent boundary layer interaction (STBLI), which is characterized by a low-frequency oscillation of the shock and a corresponding “breathing” in the separation region.

4.1 A canonical separated flow

The separated flow over an airfoil is characterized by up to three distinct natural frequencies: those of the shear layer, separation bubble, and wake. Previous work has shown that open-loop forcing at sub- and super- harmonics of these frequencies can be especially effective in controlling the extent of the separation bubble. Unfortunately, an understanding of the mechanisms driving this behavior is far from complete. As a model for separated airfoil flows, we analyze the high Reynolds number flow past a finite-thickness flat plate with an elliptical leading edge, inducing separation via a blowing and suction boundary condition rather than angle of attack. We refer to this configuration as the *canonical separated flow*. Three-dimensional large eddy simulations (LES) of the flow are analyzed using DMD and proper orthogonal decomposition (POD). We compare the results of DMD and POD for a variety of forcing frequencies, looking for trends in the modal decompositions

that correspond to control effectiveness. We find that forcing is most effective when lock-on is achieved in the most energetic modes.

This material presented in this section is the result of joint work with Ehsan Aram (Johns Hopkins University) and Professor Rajat Mital (John Hopkins University). Ehsan was responsible for the setup and execution of the numerical simulations, as well as the computation of time-averaged properties of the flow, such as the mean separation bubble height. The LES code used for those simulations was originally written by Professor Rajat Mittal [107].

4.1.1 Motivation

With ever-increasing demands for improved performance and efficiency, the control of separated flows has drawn considerable attention. In this work, we are concerned with the flow over an airfoil, wherein decreased separation can improve the lift-to-drag ratio [162]. Such flows display highly complex behavior, characterized by shear layer instability and vortex shedding in the wake, and in some cases periodic shedding of the separation bubble. In spite of this, much progress has been made in controlling separation, for instance by implementing zero-net-mass-flux (ZNMF) actuators. These devices, also known as “synthetic jets,” are favored due to their relatively simple design. Furthermore, the periodic excitation they produce has been shown to be more effective than steady forcing [149].

ZNMF actuators are also versatile, in that the forcing frequency can be adjusted to suit a particular flow. Much of the research related to ZNMF actuation is concerned with finding an optimal open-loop forcing frequency. A nondimensional frequency, F_{jet}^+ , can be defined using length and velocity scales associated with some natural frequency in the flow. (The velocity scale is often taken to be U_∞ , the freestream velocity.) For studies where the length is scaled by the chord length c , a range of values $0.55 < F_{\text{jet}}^+ < 5.5$ has been reported to be optimal [17, 40, 56, 101, 135, 148, 185]. For the length scales X_{TE} (distance from the actuator to the trailing edge) and L_{sep} (length of the separation bubble), optimal ranges of $0.50 < F_{\text{jet}}^+ < 2.0$ [68, 119, 147] and $0.75 < F_{\text{jet}}^+ < 2.5$ [62, 118, 149] have been reported, respectively. As an extreme example, Amitay et al. [7] found that for an unconventional airfoil, forcing at $F_{\text{jet}}^+ > 10$ outperformed configurations with $F_{\text{jet}}^+ < 4$. The magnitude of these discrepancies may seem insignificant, but Seifert et al. [148] reported a corresponding 25 % change in the lift coefficient C_L when F_{jet}^+ was varied between 0.25 and 1.5. Similarly, Wygnanski [185] found a 400 % increase in the required momentum coefficient C_μ over the same range of F_{jet}^+ .

Even taking into account the varying definitions of F_{jet}^+ , it is clear that there is no consensus on its optimal value. This is not a complete surprise, given the highly complex behavior of these flows. For instance, in a separated airfoil flow where the mean separation bubble remains detached, there are two dominant frequencies: those of the shear layer (caused by the separation of the boundary layer from the body) and the wake. This is typical of a bluff body flow, and studies in this field typically focus on the latter frequency and not the former [28, 76, 105]. Only Wu et al. [183] consider both parameters. When the mean flow reattaches, there can be a third natural frequency, associated with the shedding of the separation bubble. The lack of agreement on an optimal F_{jet}^+ value may indicate a need to consider all of these frequencies, as well any coupling between them. Certainly it suggests

a need for an increased appreciation of the rich, nonlinear dynamics driving these flows. For a nice overview of these issues, see the article by Mittal et al. [106]

Though separated airfoil flows are characterized by up to three natural frequencies, in certain flow configurations some of these may take on the same value. We refer to such conditions as *lock-on states*. Kotapati et al. [90] simulated a configuration in which the unforced flow locks onto a single frequency: that is, the shear layer, separation bubble, and wake frequencies are equal. For this system, actuation at the natural frequency or its first subharmonic results in a coupling of the separation bubble frequency with a superharmonic of the forcing frequency. On the other hand, forcing at superharmonics causes the wake and separation bubble to lock onto a subharmonic of the forcing frequency. It was observed that forcing at the first superharmonic significantly improves the lift-to-drag ratio, while forcing at further superharmonics is detrimental. Given the rich system dynamics, it is likely that more of these lock-on states exist, some of which may have desirable lift and/or drag properties. By increasing our understanding of the complex interactions between the shear layer, separation bubble, and wake, we may improve our ability to exploit these states for control purposes.

4.1.2 Flow configuration and solver

Rather than directly analyze the separated flow over an airfoil, we consider a model flow we refer to as the *canonical separated flow*. In this configuration, we place a finite-thickness flat plate with an elliptical leading edge in a uniform flow, at an angle of attack of zero (Figure 4.1). Separation is induced through a steady blowing/suction boundary condition applied at the top of the computational domain. By using a flat plate geometry, we eliminate the influence of curvature, which has been shown to significantly affect the nature of the boundary layer [39], as well as the receptivity of the flow to actuation [69, 70]. Furthermore, by varying the location of the plate and the amplitude of the blowing/suction boundary condition, we can specify both the location and extent of the separation bubble, something not possible by setting the angle of attack and freestream velocity alone [106]. Another advantage of the canonical separated flow is that it is amenable to both experimental [72] and computational [10, 90, 106, 172] studies; here we focus on the results of three-dimensional LES.¹

The specific geometry used in this work is defined by a 4:1 elliptical leading edge and a thickness-to-chord ratio $t/c = 0.095$. We place the upper surface of the body $0.15c$ from the upper wall and the leading edge $0.5c$ downstream of the uniform inflow. The body extends to both ends of the domain in the spanwise direction, with no spanwise variation in its geometry. We consider a domain of size $2.0c \times 0.76c \times 0.095c$ (in x , y , and z), discretized using a $512 \times 256 \times 32$ mesh. The mesh is Cartesian but non-uniform in x and y ; smaller cells are placed near the body and in the wake to resolve the dynamics of interest. Uniform spacing is used in the spanwise (z) direction. Grid dependence studies show that neither increasing the grid resolution nor increasing the size of the domain leads to significant changes in the simulation results.

Boundary conditions are chosen to allow for experimental implementations of the canonical separated flow. For instance, we enforce no-slip at the upper and lower

¹All LES computations were performed by Ehsan Aram (Johns Hopkins University).

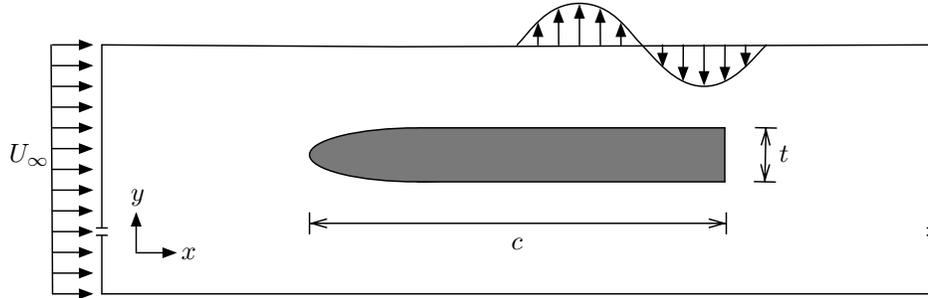


Figure 4.1: Schematic of the canonical separated flow (not to scale). A finite-thickness flat plate with an elliptical leading edge is subjected to uniform incoming flow. A separation bubble is induced by a steady blowing/suction boundary condition applied along the upper boundary of the computational domain.

boundaries of the domain and on the surface of the body, as if the body were placed in a wind tunnel. This of course excludes the portion of the upper boundary over which blowing/suction is applied. In this region, which stretches from $x/c = 0.75$ to $x/c = 1.25$ (and spans the entire domain in z), the vertical velocity v varies roughly sinusoidally with respect to x , with an amplitude of $0.65U_\infty$. (For an exact specification of this boundary condition, see the article by Kotapati et al. [90].) The inflow (freestream) velocity is chosen such that $\text{Re}_c = 100,000$. Periodic boundary conditions are applied in the spanwise direction. At the downstream boundary, a convective outflow condition minimizes the reflections that can occur when vortices leave the computational domain [87]; the convection speed is given by the average streamwise velocity along the boundary. This is the same configuration analyzed by Aram et al. [10].

The flow is actuated via ZNMF forcing, with a synthetic jet placed on the upper surface of the body at $x/c = 0.6$. We use a “slot-only” model for the jet [134], with a slot width and depth of $0.01c$. (In this model the jet cavity is not modeled at all.) The jet generates a uniform vertical velocity across the bottom of the slot. The slot velocity varies sinusoidally in time, with an amplitude such that the mean jet velocity (during expulsion) is $\bar{v}_j = 0.15U_\infty$. This results in a momentum coefficient $C_\mu = 2.25 \times 10^{-4}$.

The dynamics of this flow are governed by the incompressible Navier–Stokes equations, which we solve using a fractional-step method. To advance in time, we employ a second-order Adams–Bashforth scheme for the convective terms and an implicit Crank–Nicolson scheme for the diffusive terms. Spatial derivatives are computed using second-order finite differences. We account for the effect of solid bodies using a sharp-interface immersed boundary method [107] and model turbulence using the global dynamic coefficient LES approach introduced by Vreman [178]. This code has been validated extensively through comparisons with published numerical and experimental data [107].

4.1.3 Results

To investigate the effect of ZNMF forcing on the canonical separated flow, we simulate the flow for various nondimensional forcing frequencies $F_{\text{jet}}^+ = fc/U_\infty$. Based on a preliminary analysis, we consider the following ZNMF forcing frequencies: $F_{\text{jet}}^+ = 0.00, 2.24, 4.40$ and 6.90 . These correspond loosely to no forcing (uncontrolled flow), forcing at the wake frequency, forcing at the separation bubble frequency, and forcing at the shear layer frequency. Each case is analyzed using both POD and DMD. We then compare the results, looking for features in the modal decompositions that correspond to control effectiveness. Specifically, we look for signs of lock-on involving combinations of the wake, separation bubble, and shear layer. Such features may provide insight into the underlying physics that govern the flow.

Separation bubble height

In the canonical separated flow, the goal of ZNMF forcing is to reduce the effects of separation. There are many ways to measure this effect. Table 4.1 summarizes the effect of ZNMF forcing on the time-averaged separation bubble, in terms of its height δ^* (normalized with respect to the synthetic jet slot width d) and length L_{sep} (normalized with respect to the chord c). (These values are computed as in [90].²) For the remainder of this discussion, we measure control effectiveness in terms of reduction in the mean separation bubble height.

Energy distribution (POD analysis)

Recall that POD analysis takes a set of velocity fields and from them generates spatial modes that optimally capture the kinetic energy in the flow. (For more details on POD, see Appendix A.) Figure 4.2 shows the dominant POD mode for each of the four forcing frequencies. (Due to the traveling wave nature of the flow, these modes come in pairs, only one of which is illustrated.) We see that in three of the four cases (all but $F_{\text{jet}}^+ = 4.40$), the dominant mode displays the same general structure. The contours of v (vertical velocity) are large elliptical surfaces, extending across the centerline of the body in a roughly symmetric manner. These occur only in the region downstream of the body, with no support elsewhere. This is a commonly

²These values were computed by Ehsan Aram (Johns Hopkins University).

Table 4.1: Separation bubble size for various ZNMF forcing frequencies.

F_{jet}^+	Height (δ^*/d)	Length (L_{sep}/c)
0.00	5.4	0.47
2.24	3.0	0.15
4.70*	1.8	0.13
6.92	2.4	0.05

* Data for $F_{\text{jet}}^+ = 4.40$ are not available, but the behavior is very similar to that of $F_{\text{jet}}^+ = 4.70$.

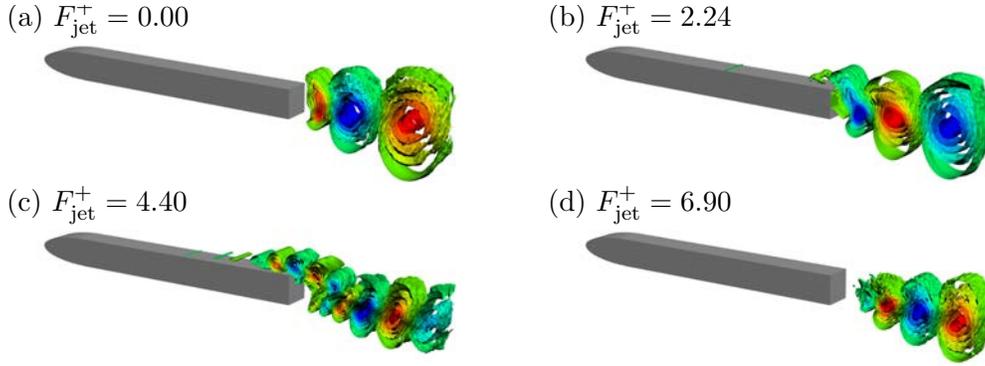


Figure 4.2: Comparison of dominant POD modes, illustrated using contours of vertical velocity v . All cases except for $F_{\text{jet}}^+ = 4.40$ show the same general structure, characteristic of a von Kármán vortex street.

observed modal structure in bluff-body wakes and is associated with von Kármán vortex streets. As such, we can say that in each of these cases, the flow is dominated by von Kármán vortex shedding.

In contrast, for $F_{\text{jet}}^+ = 4.40$, the dominant mode displays none of these characteristics (Figure 4.2 (c)). There is clear support in the separation bubble region, over the upper surface of the trailing edge. In the near wake, the contours are smaller than for the other forcing frequencies and originate from the flow over the upper surface of the body. Even in the far wake, where these surfaces grow and extend across the centerline of the body, they are not nearly as large as in the wake modes that dominate for the other forcing frequencies. Furthermore, the contours of v never show the same top-bottom symmetry seen in the other cases.

If we consider the secondary POD modes, we find a mode for $F_{\text{jet}}^+ = 4.40$ that resembles a wake mode (Figure 4.3 (c)). However, the wake structures are much more irregularly shaped than those found in the wake modes for other forcing frequencies. In addition, for $F_{\text{jet}}^+ = 4.40$, we see from Figure 4.4 that the dominant and secondary mode pairs contain roughly the same amount of energy (about 15 % and 12 % per mode, respectively). This differs from the other cases, where the dominant modes contain at least twice the energy of the secondary ones, if not more. In fact, comparing to Table 4.1, we see that the reduction in separation bubble height correlates with the evenness of the POD energy distribution. Thus it appears that ZNMF forcing is most effective when it leads to a more equitable distribution of energy in the various modes of the flow.

Lock-on modes (DMD analysis)

Recall from Chapter 2 that DMD analysis decomposes a set of velocity fields into spatial modes that each oscillate with a single temporal frequency. This can be used to identify the characteristic frequencies of a flow, as well as the corresponding spatial structures. In addition, the support observed in a DMD mode can be used as evidence of lock-on. Because each DMD mode oscillates at a single frequency, if a DMD mode shows support in two regions of the flow, then those two regions

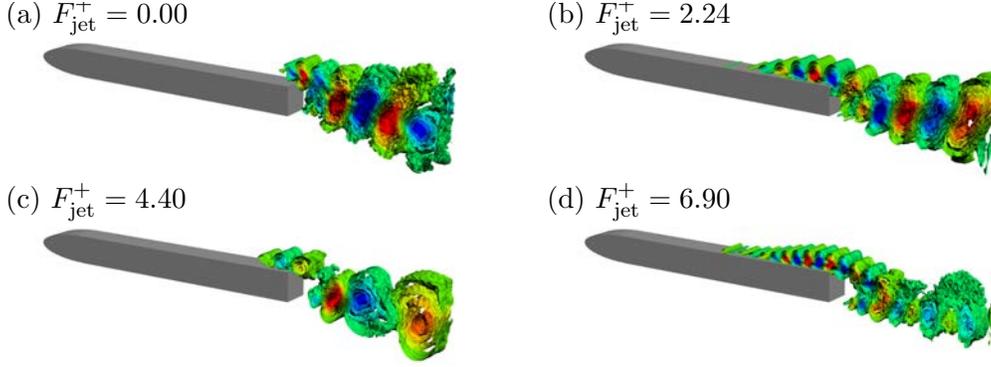


Figure 4.3: Comparison of secondary POD modes, illustrated using contours of vertical velocity v . The mode for $F_{\text{jet}}^+ = 4.40$ contains large contours similar to those observed in Figure 4.2 for other frequencies.

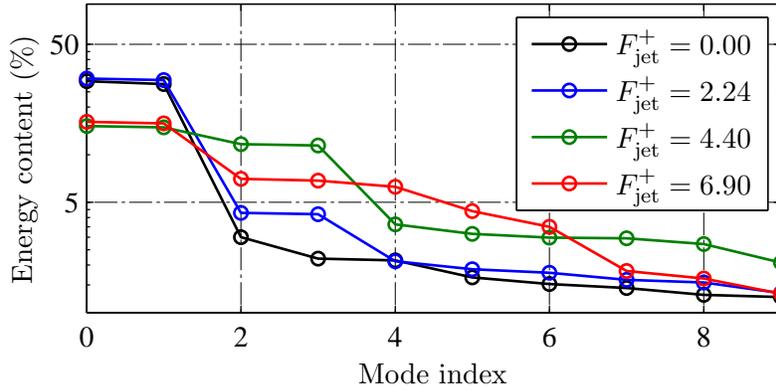


Figure 4.4: POD energy content. A more even distribution of energy among the first few POD modes correlates with a decreased mean separation bubble height (see Table 4.1).

must exhibit oscillations at the same frequency. For instance, if a particular mode has support in both the wake region and the shear layer region, then we can say the wake and shear layer are locked-on.

Figure 4.5 shows a comparison of the DMD spectra for the canonical separated flow with various forcing frequencies applied. (We perform all DMD computations using sequential series of snapshots, as described in Section 2.3.1.) The separated flow with no forcing exhibits only approximately harmonic structure, with a fundamental forcing frequency corresponding to the wake shedding frequency (Figure 4.5 (a), blue peak), as evidenced by the structure of the corresponding mode (Figure 4.6 (a)). In each case where forcing is applied, the DMD spectrum becomes more strongly harmonic. For forcing at $F_{\text{jet}}^+ = 2.24$ (Figure 4.5 (b)), we observe harmonics of the forcing/wake frequency. When forcing is applied at $F_{\text{jet}}^+ = 4.40$ (Figure 4.5 (c)), we instead see harmonics of the forcing/separation bubble frequency. Forcing at $F_{\text{jet}}^+ = 6.90$ (Figure 4.5 (d)) produces harmonics of the wake frequency again.

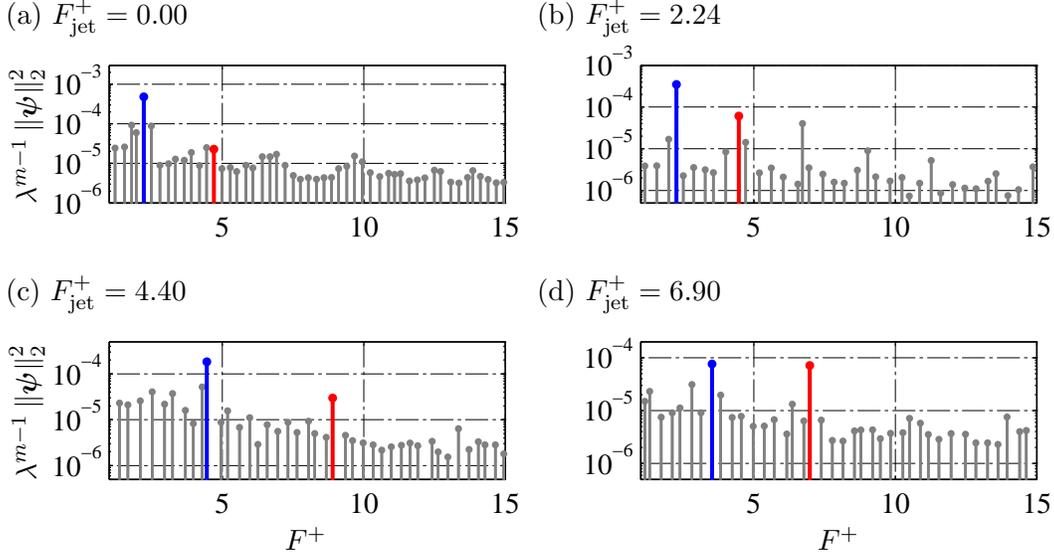


Figure 4.5: Comparison of DMD spectra. Spectral peaks corresponding to the modes depicted in Figures 4.6 and 4.7 are shown in blue and red, respectively. The spectrum for the unforced flow contains only approximate harmonics, whereas the spectra for the forced flows are characterized by near-exact harmonics. Only in the case of $F_{\text{jet}}^+ = 4.40$ is the fundamental frequency not associated with a wake mode (see Figure 4.6).

Looking at the corresponding modes (Figure 4.6), we see that just as in the POD analysis, all cases except $F_{\text{jet}}^+ = 4.40$ produce a wake mode. For each of these cases, this wake mode corresponds to the fundamental frequency of the system. In contrast, for $F_{\text{jet}}^+ = 4.40$ the fundamental frequency corresponds to a very different mode shape (Figure 4.6 (c)). This mode resembles the dominant mode observed in POD analysis (Figure 4.2 (c)), with support over the trailing edge and smaller elliptical contours in the far wake. Because these structures appear in a single DMD mode, we can say that these structures oscillate at the same frequency. As such, for $F_{\text{jet}}^+ = 4.40$, the flow is dominated by lock-on between the wake and the separation bubble (both as a fundamental frequency and in terms of energy content). For the other forcing frequencies, the flow is dominated by the wake mode.

If we look at the superharmonic DMD modes (corresponding to the red peaks in Figure 4.5), we observe lock-on for the other forcing frequencies (Figure 4.7). These often involve small structures extending from the mid-chord of the body all the way into the wake region (Figure 4.7 (b)–(d)). Such modes indicate lock-on between the shear layer and harmonics of the wake, rather than the dominant structures of the wake. Interestingly, for $F_{\text{jet}}^+ = 4.40$, there is no evidence of a wake mode at any frequency. This is unlike the POD results, which simply show that the wake mode has diminished energy content. Thus, for this forcing frequency, the wake mode is a high-energy structure, but not a temporally oscillatory one. In other words, the time evolution of the wake is not characterized by a steady oscillation of a von Kármán vortex street. Recall that the $F_{\text{jet}}^+ = 4.40$ case is the most effective in reducing the

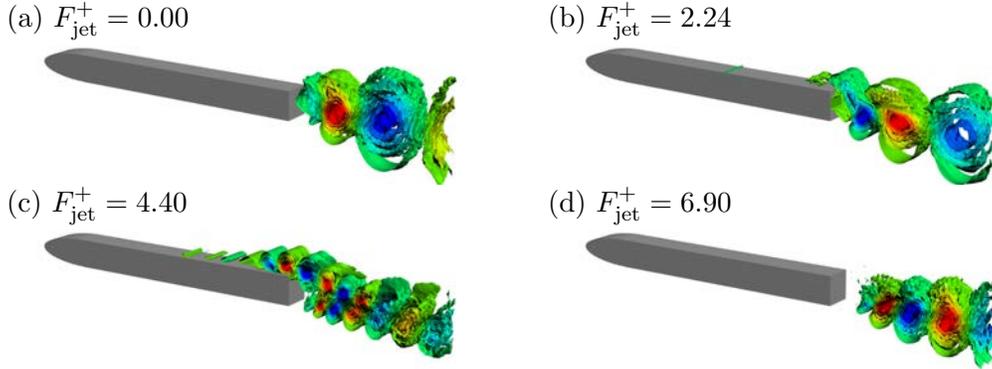


Figure 4.6: Comparison of fundamental DMD modes. Note the similarity to the dominant POD modes shown in Figure 4.2. All cases except $F_{\text{jet}}^+ = 4.40$ show the same general structure, characteristic of von Kármán vortex shedding. For $F_{\text{jet}}^+ = 4.40$, the location of the spatial support is evidence of lock-on between the separation bubble and wake.

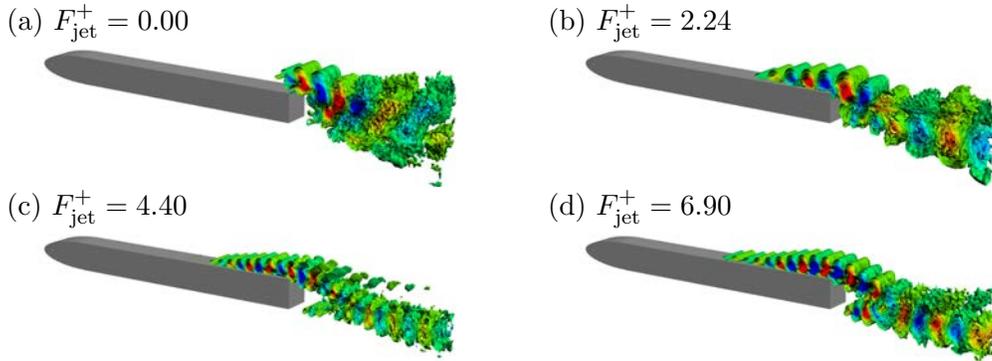


Figure 4.7: Comparison of secondary DMD modes. Except for the case of $F_{\text{jet}}^+ = 4.40$, these modes are similar to the secondary POD modes shown in Figure 4.3.

mean separation bubble height. We see here that it is also the most unique in terms of its modal characteristics.

4.1.4 Summary

In this section we applied modal decomposition techniques to analyze the canonical separated flow, a model for separated airfoil flows. Specifically, we used POD and DMD to decompose a flow into its high-energy and oscillatory components, respectively. We performed this analysis for a number of ZNMF forcing frequencies, looking for trends that corresponded to control effectiveness. We found that the most effective forcing frequency, $F_{\text{jet}}^+ = 4.40$, produced the most unique modal characteristics.

POD analysis showed that for $F_{\text{jet}}^+ = 4.40$, the energy distribution in the POD modes was more evenly distributed. In fact, the flatness of the energy distribution correlated directly with the separation height (the flatter, the smaller the bubble). The $F_{\text{jet}}^+ = 4.40$ case was the only one not dominated by a wake mode; the wake mode in this case appeared in the secondary POD modes. DMD analysis found that $F_{\text{jet}}^+ = 4.40$ was the only case whose spectrum was not dominated by superharmonics of the wake frequency. Instead, the fundamental frequency of this flow was the forcing/separation bubble frequency, and no wake mode could be found at all. This mode shows support in both the separation bubble and wake regions, indicating lock-on between the two. No other forcing frequency generated lock-on of this nature.

4.2 Shock-turbulent boundary layer interactions

It is well known that the interaction of a shock wave with a turbulent boundary layer leads to low-frequency, oscillatory shock motion. While some have argued that these oscillations are primarily correlated with upstream fluctuations, others have found evidence that they are primarily correlated with downstream fluctuations; their origin remains a question. In this section, we present preliminary work in which we apply DMD to analyze the Mach 2.9 flow over a 24° compression ramp. Our goal is to identify any DMD modes whose frequencies match that of the shock motion. The spatial support observed in such modes may provide insight into the physics driving the shock oscillation.

This material is the result of joint work with Stephan Priebe (Princeton University) and Professor Pino Martín (University of Maryland). It is an extension of the work presented in Priebe and Martín [132] and Priebe [131]. Stephan Priebe conducted the numerical simulations and provided assistance in non-DMD postprocessing (e.g., computing skin friction coefficient profiles) and visualization of the three-dimensional results.

4.2.1 Motivation

Shock-turbulent boundary layer interactions (STBLIs) occur in many engineering applications. For instance, a normal shock may form in the transonic flow over an airfoil. When the shock meets the airfoil surface, it will interact with the (typically turbulent) boundary layer, leading to a normal STBLI. STBLIs also occur due to shocks that form in the flow around high-speed (super- or hyper-sonic) vehicles. For instance, they can be caused by the deflection of control surfaces, which produces a compression ramp and an accompanying oblique shock.

The presence of STBLIs often leads to a degradation in performance. A common example is the engine of a scramjet, in which compression is achieved upstream of the combustion chamber via a series of shocks. Because each of these shocks terminates within the engine, a series of STBLIs results. This can lead to a number of undesirable effects, including large-scale unsteady separation and significant fluctuations in pressure and heat transfer at solid surfaces [33, 42, 44, 157]. Unsteady separation effects contaminate downstream flow, which may affect engine performance. Fluctuations in heat transfer have been known to cause damage to vehicle surfaces [8, pp. 396].

Unfortunately, the origin of STBLIs is not well-understood. A number of studies suggest that upstream influences may cause the low-frequency oscillations characteristic of STBLIs. Beresh et al. [22] found that the shock motion resulting from an STBLI is correlated with upstream fluctuations in the streamwise boundary layer velocity. Ganapathisubramani et al. [58, 59] found a correlation between spanwise-constant motion in the separation bubble and global changes in the upstream velocity and suggested that these “superstructures” in the incoming boundary layer could be responsible for the shock motion. Experimental results by Humble et al. [82] found similar correlations between unsteadiness in the upstream boundary layer and unsteadiness due to an STBLI. However, many studies also find that downstream influences may be important to STBLI dynamics. Using unsteady pressure measurements, many have shown that fluctuations near the foot of the shock correlate with fluctuations below the downstream separation region, in an out-of-phase manner [50, 53, 161]. Analogous results relating the shock motion to changes in the separation bubble have been found using particle image velocimetry (PIV) [128] and direct numerical simulation (DNS) [184]. Priebe and Martín [132] investigated the relative importance of upstream and downstream effects. They found that the shock motion correlates strongly with a breathing of the separation bubble and the associated flapping of the separated shear layer, whereas correlation with features of the upstream boundary layer is weaker. However, they also remark that this relationship may vary depending on the degree of separation.

There have also been numerous efforts to directly model the physics of STBLIs, rather than simply find correlations. For instance, Piponniau et al. [128] proposed an entrainment-based model for oscillations in the downstream separation bubble, which lead to oscillatory motion in the shock. Toubert and Sandham [165] used a linear stability analysis of the mean velocity field to identify an unstable global mode that may cause the observed oscillations. Using DNS, Priebe and Martín [132] found a C_f (coefficient of friction) signature that matches that of the unstable mode computed by Toubert and Sandham [165]. Here, we extend this work, looking to find further evidence of an unstable mode in DNS data. We do so using DMD, seeking modes whose frequency matches that of the shock motion. A match in the DMD and global stability modes would validate the hypothesis that the oscillations observed in STBLIs are caused by the saturation of a linear instability.

4.2.2 Flow configuration and solver

We consider the Mach 2.9 flow over a 24° compression ramp. (Other studies consider STBLIs generated by reflected shocks; the results are analogous.) Figure 4.8 shows the geometry of the computational domain. The domain is discretized with uniform spacing in the spanwise (y) direction. In the wall-normal (z) and streamwise (x) directions, grid points are clustered according to a hyperbolic sine transformation, with more points allocated near the wall and the compression corner, respectively. This results in a $1024 \times 160 \times 128$ mesh (in x , y , and z). For DMD, we downsample in the spanwise direction, leaving flow variables defined on a $1024 \times 40 \times 128$ grid. To avoid aliasing, we use following filter, with $n = 4$:

$$\bar{f}_i = \frac{1}{2n} \left(f_{i-n/2} + 2 \sum_{j=i-n/2+1}^{i+n/2-1} f_j + f_{i+n/2} \right),$$

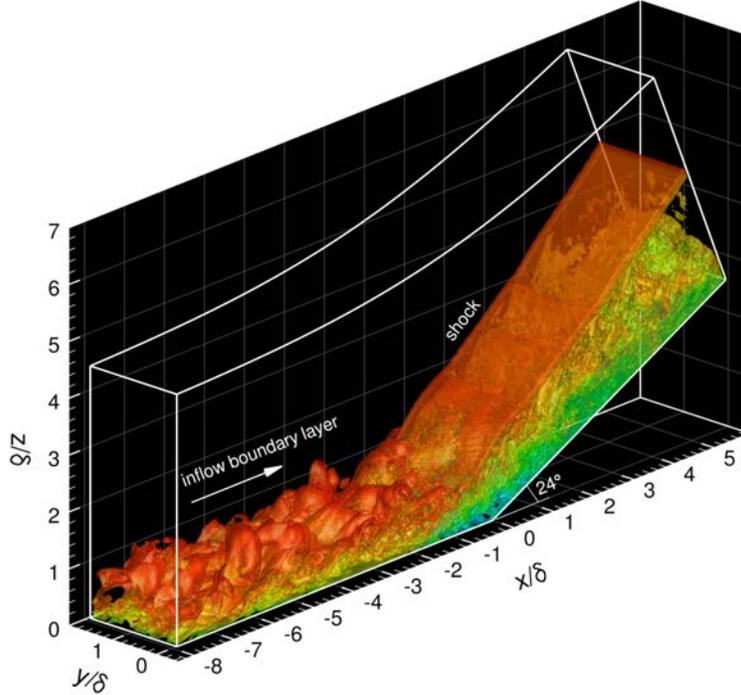


Figure 4.8: Schematic of domain used for DNS of STBLI. (Figure courtesy of Stephan Priebe.)

where \bar{f} denotes filtered values and f denotes the original values. On this coarse grid, we are able to resolve features on the scales found in large eddy simulations (LES).

The flow is solved using a discretization of the three-dimensional, compressible Navier–Stokes equations. (I.e., we simulate the flow using three-dimensional DNS.³) A perfect gas is assumed. Inviscid fluxes are discretized using a fourth-order weighted essentially non-oscillatory (WENO) scheme. Viscous fluxes are discretized using fourth-order finite differences. We step the flow forward in time using a third-order, low-storage Runge–Kutta method [181]. A no-slip, isothermal boundary condition is applied at the wall. Supersonic outflow conditions are imposed at the lid and outlet of the domain. In the spanwise direction, we use periodic boundary conditions. The inflow boundary condition is determined using the results of an auxiliary DNS. For further details on the computational method, see Wu and Martín [184] and Priebe and Martín [132].

4.2.3 Preliminary results

Based on the work of Touber and Sandham [165] and Priebe and Martín [132], we expect the dominant features of the flow near the shock foot to be two-dimensional. As such, we first consider a DMD analysis of the spanwise-averaged flow field. (We apply DMD to sequential snapshots of the velocity field, as described in 2.3.1.) Figure 4.9 shows that while the DMD spectrum is fairly broadband, it does identify

³All DNS computations were performed by Stephan Priebe (Princeton University), as were all visualizations of three-dimensional data.

a mode with non-dimensional frequency $f^+ = 0.066$; this frequency remains nearly constant even as the snapshot set is varied. (f^+ is non-dimensionalized with respect to the mean separation length and the freestream velocity.) The corresponding mode is illustrated in Figure 4.10. We observe that along the wall, the sign of the streamwise velocity changes at and downstream of the shock corner. For instance, in the imaginary part of the mode, as we move along the wall we see a strong region of positive velocity perturbation near the mean separation point, weaker negative velocity perturbations at the corner, and strong positive perturbations downstream of the corner. This general behavior matches the global stability mode identified by Touber and Sandham [165].

To better illustrate the contribution of this DMD mode to the flow, we recall that DMD modes can be considered extensions of global stability modes to nonlinear dynamics. As such, the delineation between the real and imaginary parts of the mode is somewhat arbitrary. The contribution of the mode will evolve in time, transitioning from the real part to the imaginary part and back again, just as the eigenvector of a linear system would. Figure 4.11 compares the ΔC_f profile from a particular point in this transition⁴ to the mean-subtracted profile found by Priebe and Martín [132], who computed the profile using a conditional averaging approach. We observe that DMD confirms the key features of the profile, including a large

⁴The ΔC_f profiles were computed by Stephan Priebe (University of Maryland).

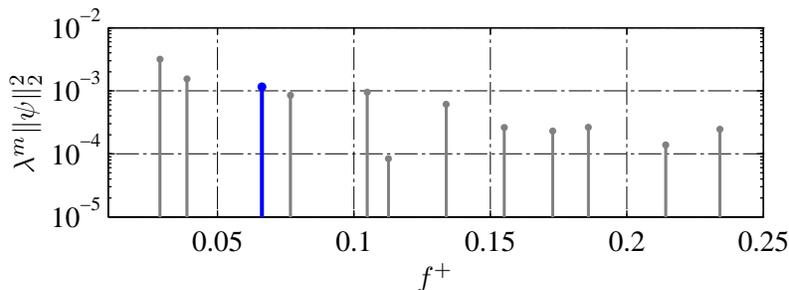


Figure 4.9: DMD spectrum computed from spanwise-averaged data. The spectral peak corresponding to the mode depicted in Figure 4.10 is shown in blue. Though the spectrum is fairly broadband, there is a peak at the expected frequency.

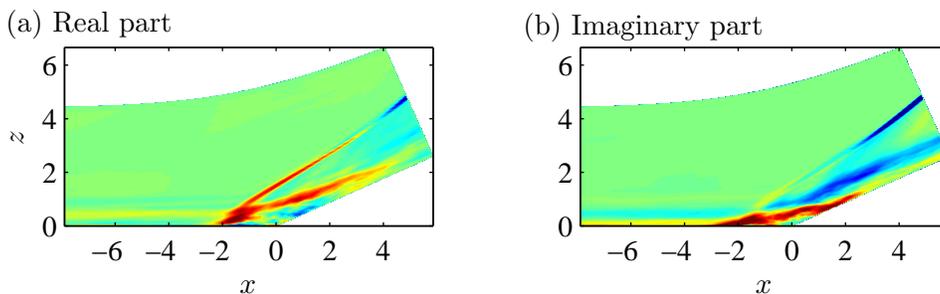


Figure 4.10: DMD mode computed from spanwise-averaged data, illustrated using contours of streamwise velocity u . Changes in the sign of the velocity along the wall match those observed in linear stability modes computed by Touber and Sandham [165]. (Positive perturbations are shown in red; negative ones are shown in blue.)

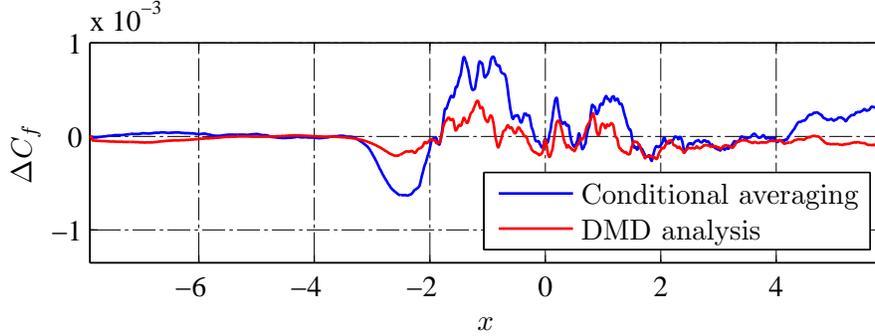


Figure 4.11: Comparison of ΔC_f profiles. The contribution of the identified DMD mode (red) matches the mean-subtracted profile computed by Priebe and Martín [132] using conditional averaging (blue).

negative region followed by a large positive region, and smaller positive regions further downstream. These features are also observed in the global stability mode computed by Touber and Sandham [165].

Initial computations using the full three-dimensional flow show the same general features. Again, the DMD spectrum is fairly broadband (Figure 4.12). However, there is a peak at $f^+ = 0.060$, near the expected frequency and close to the value $f^+ = 0.066$ identified using spanwise-averaged data (Figure 4.9). Figure 4.13 shows a wall-normal plane from the corresponding DMD mode, taken at the grid point closest to the wall. We see that generally, the streamwise slices are characterized by same-sign velocity perturbations on each side of the corner ($x/\delta = 0$). This is particularly clear in the imaginary part of the mode. For instance, at $y/\delta = 0.25$, there is a strong positive perturbation from $x/\delta = -3$ to $x/\delta = -2$, as well as from $x/\delta = 3$ to the end of the domain. This is similar to the behavior observed in the DMD mode computed from spanwise-averaged data (Figures 4.10 and 4.11).

However, in the three-dimensional DMD mode spanwise variations are also prevalent. In Figure 4.13, we see strong evidence of streamwise streaks downstream of the corner, which matches the findings of Priebe and Martín [132]. These streaks can be seen clearly by plotting iso-contours of the conservative variable ρu , as shown in Figure 4.14. The spanwise variations near the shock foot ($x/\delta = -2$ to $x/\delta = 0$) are weaker, but non-trivial. From Priebe and Martín [132], we expect the flow field in this region to be strongly two-dimensional. This suggests that other DMD modes, perhaps ones with frequencies close to $f^+ = 0.060$, should also be considered, as two-dimensionality may be a cumulative result of many simultaneous mode oscillations.

4.2.4 Summary

In this section, we used DMD to investigate the interaction of shock waves with turbulent boundary layers. Such flows are characterized by low-frequency oscillations, the cause of which is not known. Using DNS of a Mach 2.9 flow over a 24° compression ramp, we found a DMD mode whose frequency matches that of the shock motion. The spatial support of this mode matches flow features observed previously by Priebe and Martín [132] and Touber and Sandham [165]. The same

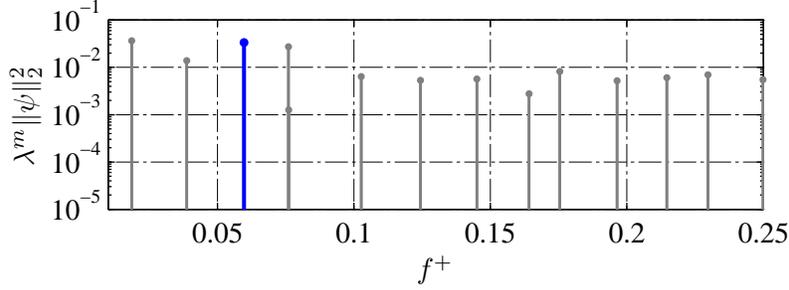


Figure 4.12: DMD spectrum computed from three-dimensional STBLI data. The spectral peak corresponding to the mode depicted in Figure 4.14 is shown in blue. As with the spanwise-averaged data (Figure 4.9) the spectrum is fairly broadband, with a peak appearing at the expected frequency.

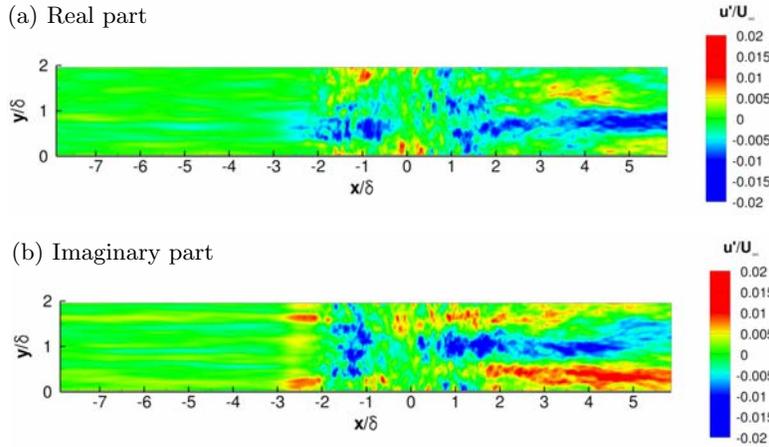


Figure 4.13: Wall-normal plane from a three-dimensional DMD mode, taken at grid point closest to the wall and illustrated using contours of u . Each streamwise slice is characterized by regions of same-sign velocity perturbations upstream and downstream of the corner ($x/\delta = 0$). The variations in the spanwise direction indicate the presence of streamwise streaks. (Figure courtesy of Stephan Priebe.)

agreement is observed whether the DMD mode is computed using spanwise-averaged or three-dimensional flow fields. These initial results support the hypothesis that the low-frequency oscillations observed in STBLIs result from a saturation of a linear instability [132, 165].

4.3 Conclusions

This chapter focused on the use of DMD to analyze oscillatory fluid flows. Two examples were explored: a separated flow and the interaction between a shock wave and a turbulent boundary layer. In the former, DMD was applied to identify characteristic flow frequencies and their corresponding spatial structures. Trends in the modal decompositions were found to correlate with the effectiveness of ZNMF actuation in decreasing the mean separation bubble height. The latter example used DMD to identify coherent structures corresponding to a known frequency. Features

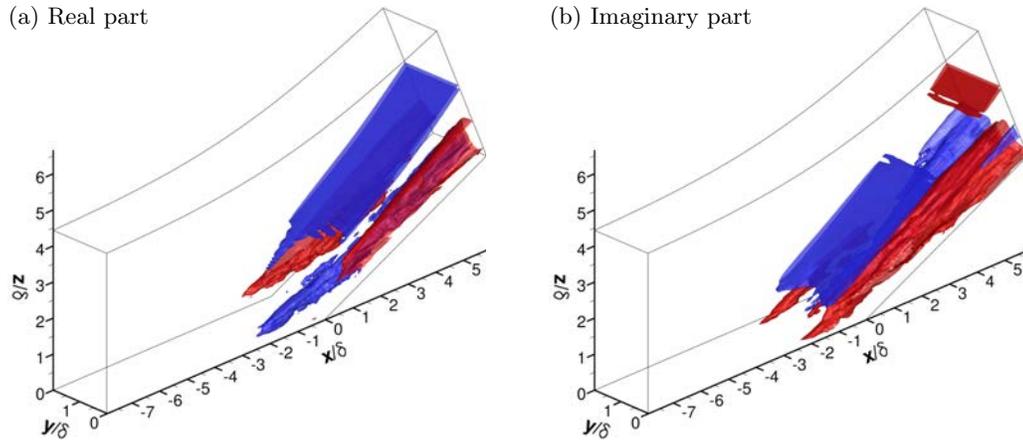


Figure 4.14: DMD mode computed from three-dimensional STBLI data, illustrated using iso-surfaces of ρu ($\rho u = 2$ in red, $\rho u = -2$ in blue). The shock itself is fairly two dimensional. Beneath the shock, streamwise streaks are evident. (Figure courtesy of Stephan Priebe.)

of this mode matched those observed in previous work, confirming behavior found previously using other methods. Both cases demonstrate the power of DMD as a tool for analyzing large datasets. While DMD analysis alone cannot explain the physics governing a complex system, it can provide insight into those physics and suggest avenues for further study.

Chapter 5

Overcoming time resolution issues via dynamic estimation

The previous two chapters have both demonstrated applications of dynamic mode decomposition (DMD) to simulation data. Of course, one of the advantages of DMD is that it can also be applied to experimental data. This was first shown by Schmid [141] in his 2010 paper. Over the next two chapters, we explore one of the key challenges in collecting experimental data for DMD: time resolution.

DMD, like most signal processing methods, requires that data be collected at least twice as fast as any frequency of interest; this is a result of the well-known Nyquist-Shannon sampling theorem [117, 152]. In fluid applications, DMD is typically applied to velocity fields, which are typically collected using particle image velocimetry (PIV). PIV hardware limitations make it impossible to satisfy the Nyquist-Shannon sampling criterion for many fluid flows. On the other hand, point measurements, for instance of velocity or pressure, can be collected on very fast time scales using standard equipment.

In this chapter, we describe a three-step method that takes advantage of this fact, leveraging both time-resolved point measurements and non-time-resolved PIV data to estimate time-resolved velocity fields. First, a variant of linear stochastic estimation (LSE) is used to obtain an initial set of time-resolved estimates of the flow field. These estimates are then used to identify a linear model of the flow dynamics. Finally, the model is incorporated into a Kalman smoother, which provides an improved set of estimates. These estimates can be used for DMD computations, which would otherwise be unable to capture the relevant flow physics.

We verify this method with an experimental study of the wake behind an elliptical-leading-edge flat plate at a thickness Reynolds number of 3,600. We find that, for this particular flow, the Kalman smoother estimates are more accurate and more robust to noise than the initial, stochastic estimates. Consequently, DMD more accurately identifies coherent structures in the flow when applied to the Kalman smoother estimates. Causal implementations of the estimators, which are necessary for flow control, are also investigated. Similar outcomes are observed, with model-based estimation outperforming stochastic estimation, though the advantages are less pronounced.

The remainder of this chapter is structured as follows: Section 5.1 motivates the need for improved estimation techniques in fluid mechanics. In Section 5.2, we

provide a brief introduction to the theory of stochastic and dynamic estimation. These estimation techniques are implemented using the numerical methods detailed in Section 5.3 and demonstrated using the experiment described in Section 5.4. The results of this experiment are discussed in Section 5.5, and conclusions drawn therefrom are presented in Section 5.6.

This material is based on the article by Tu et al. [173], and as such, we use the notation found in that work. All experimental data acquisition was done by John Griffin (University of Florida) and Adam Hart (University of Florida Research and Engineering Education Facility), under Professors Louis N. Cattafesta III (Florida State University) and Lawrence S. Ukeiley (University of Florida). John Griffin was also responsible for all computations related to stochastic estimation.

5.1 Motivation

Knowledge of the full velocity field can be of great use in identifying and visualizing spatial structures in a flow. For instance, proper orthogonal decomposition (POD) can be used to identify structures with high-energy content [96]. However, the data must be *time-resolved* in order to elucidate the *full* dynamics of the flow. Certainly, if the data do not resolve the time scales of interest, then the corresponding behaviors will not be captured. If time-resolved velocity fields are available, structures of dynamical importance can be identified using methods such as balanced POD (BPOD) and DMD [137, 138, 141]. Time-resolved, full-field information is also helpful in forming reduced-order models for closed-loop flow control, or for simply visualizing a flow. Unfortunately, time-resolved velocity fields are difficult to obtain.

PIV is the standard technique for measuring velocity fields (“snapshots” of a flow), but time-resolved PIV (TRPIV) systems are costly and thus uncommon. In addition, such systems are often restricted to low-speed flows due to the larger time interval needed between snapshots when using a high-speed laser. The required sampling rates can also limit the spatial extent of the data that can be captured [163]. As such, typical PIV systems are not time-resolved and as a result are often incapable of resolving the characteristic frequencies of a flow.

On the other hand, many instruments exist for capturing time-resolved “point” measurements, including hot-wire probes and unsteady pressure sensors. Arrays of such instruments can be used to take measurements that span large spatial regions, but these data may not resolve all the spatial scales of interest. The dense arrays necessary to capture small-scale structures are often too intrusive, and any measurement is limited by spatial averaging on the scale of the sensor’s size. Furthermore, point measurements can be sensitive to the placement of the instrument, which is generally predetermined.

In this work, we demonstrate a three-step method that integrates time-resolved point measurements of velocity, non-time-resolved PIV snapshots, and a dynamic model to estimate the time evolution of a velocity field. As we only wish to resolve the dominant coherent structures, we use POD to obtain a low-order description of the flow. First, a variant of LSE is used to compute an initial set of time-resolved estimates of the velocity field. We then form a model of the flow physics by combining an analytic characterization of the flow with a stochastic one identified from the

initial estimates. The resulting model is used as the basis for a dynamic estimator called a Kalman smoother, with which a second set of estimates is computed.

Whereas the initial LSE estimates are determined by the point measurements alone, the Kalman smoother also incorporates the non-time-resolved PIV snapshots. These two sets of measurements are used to correct an internal, model-based prediction of the estimate. The dynamics of the model prevent the Kalman smoother estimates from evolving on time scales that are fast with respect to the flow physics, filtering out measurement noise. Thus, we can leverage knowledge of the flow physics and a non-time-resolved description of the velocity field to obtain a more accurate and robust set of estimates.

In many ways, this work builds on that of Murray and Ukeiley [111], Taylor and Glauser [160], and Tinney et al. [164] (among others), who all used LSE-based methods to estimate the time evolution of a flow field. The key difference between our approach and those based solely on LSE is our use of a dynamic model. LSE is a conditional technique for capturing those features of the flow that are correlated with a measurement signal, and does not rely on, or provide, a model of the flow physics. Our approach also differs from the recent work by Legrand et al. [91, 92], in which a phase-averaged description of a velocity field is obtained directly from a large ensemble of PIV data. Theirs is a post-processing technique that does not make use of any other measurement signal.

As a proof of concept, we apply this method in a bluff-body wake experiment. A finite-thickness flat plate with an elliptical leading edge and blunt trailing edge is placed in a uniform flow, resulting in oscillatory wake dynamics. (This is the same flow configuration analyzed in Section 2.4.4.) We collect TRPIV snapshots, from which we extract the velocity at a single point in the wake, simulating a probe signal. POD modes are computed from the TRPIV data and a set of basis modes is chosen for approximating the flow field. The TRPIV snapshots are then downsampled (in time), and these non-time-resolved data are fed to the dynamic estimator along with the time-resolved probe signal. This generates an estimated, time-resolved trajectory for each POD mode coefficient.

The estimation error is quantified using the original TRPIV data, with the following analysis applied to both the initial LSE estimates and the Kalman smoother estimates. For each TRPIV snapshot, we compute the difference between the estimated POD representation of the velocity field and its projection onto the POD basis. The kinetic energy contained in this difference is then calculated. We collect the values and find the mean value of the error energy and its distribution. This procedure is then repeated with various levels of artificial noise injected into the probe signal, in order to test each method’s sensitivity to noise. Finally, the estimated flow fields are used to compute DMD modes, testing whether or not the estimates are accurate enough to identify the oscillatory structures in the flow.

This last computation demonstrates the value of our method in post-processing analysis, as DMD would not be possible without time-resolved estimates of the velocity field. Previous work has shown that dynamic estimators and reduced-order models can also be useful in flow control applications. Gerhard et al. [61] reduced wake oscillations in simulations of the flow over a circular cylinder using a dynamic estimator and a low-dimensional Galerkin model. Li and Aubry [93] and Protas [133] achieved similar results using low-order vortex models. Pastoor et al. [122] used a vortex model to describe the wake behind a D-shaped body (similar to the

one analyzed in this work), successfully stabilizing it in experiment using both open- and closed-loop control. In that work, a Kalman filter was applied to dynamically estimate the base pressure fluctuations for vortex shedding synchronization. While the focus in this chapter is reduced-order estimation and not feedback control, we note that our method can easily be modified for flow control purposes.

5.2 Background

5.2.1 Stochastic estimation

In many instances, we may wish to estimate the value of an event based on the value of another one. Suppose we would like to use the velocity measurement at one point in a flow, $\mathbf{u}(\mathbf{x})$, to estimate the velocity at another point, $\mathbf{u}(\mathbf{x}')$. The conditional average

$$\hat{\mathbf{u}}(\mathbf{x}') = \langle \mathbf{u}(\mathbf{x}') | \mathbf{u}(\mathbf{x}) \rangle \quad (5.1)$$

provides the expected value of $\mathbf{u}(\mathbf{x}')$ given the measurement $\mathbf{u}(\mathbf{x})$ [120, Chapter 2].

We can estimate the conditional average by measuring $\mathbf{u}(\mathbf{x}')$ repeatedly and averaging over those values that occur whenever $\mathbf{u}(\mathbf{x})$ is near a nominal value $\mathbf{u}^*(\mathbf{x})$ [74], but this procedure would have to be repeated for all potential values of $\mathbf{u}(\mathbf{x})$. (I.e., it would only give us $\langle \mathbf{u}(\mathbf{x}') | \mathbf{u}(\mathbf{x}) = \mathbf{u}^*(\mathbf{x}) \rangle$.) Adrian [1] introduced stochastic estimation to the turbulence community, approximating the conditional average (as a function) with the power series

$$\langle \mathbf{u}_i(\mathbf{x}') | \mathbf{u}_i(\mathbf{x}) \rangle \approx \mathbf{A}_{ij}(\mathbf{x}') \mathbf{u}_j(\mathbf{x}) + \mathbf{B}_{ijk}(\mathbf{x}') \mathbf{u}_j(\mathbf{x}) \mathbf{u}_k(\mathbf{x}) + \dots, \quad (5.2)$$

where summation over repeated indices is implied. In the case of LSE, only the linear coefficients \mathbf{A}_{ij} are retained. These can be computed from the two-point, second-order correlation tensor $\mathbf{R}_{ij}(\mathbf{x}')$ [2].

Similar procedures exist for higher-order estimates, making use of higher-order two-point correlations. While Tung and Adrian [175] found that higher-order estimation procedures did not provide much additional accuracy, later studies showed that this is not always the case. For instance, quadratic estimation can be more effective when the estimation of a given quantity (e.g., velocity) is based on the measurement of another (e.g., pressure) [110, 113].

Other studies achieved improved performance by accounting for time delays between the conditional and unconditional variables [74]. Ewing and Citriniti [55] developed a multi-time LSE technique in the frequency domain that was a significant improvement over single-time LSE. This multi-time formulation also incorporated global analysis tools, namely POD, that yielded low-dimensional representations of the turbulent jets being studied. The multi-time approach was later translated into the time domain and used to predict future pressure values from past measurements [176]. Durgesh and Naughton [51] demonstrated the existence of an optimal range of delays when they estimated the POD mode coefficients of a bluff-body wake in a non-causal, post-processing fashion.

We note that the stochastic estimation of POD coefficients from measurements is typically referred to as *modified* LSE (mLSE), or more recently, modified linear stochastic measurement. The latter name is used to distinguish this as a measurement (state) estimation as opposed to a plant estimation (system identification),

which would also be typical from a controls perspective [63]. In this work, we use the term “LSE” as it is more prevalent in the literature.

It is important to note that stochastic estimation does not involve any modeling of a system’s dynamics. Rather, it simply provides a statistical estimate of a random variable given the knowledge of other random variables [2]. We can think of stochastic estimation as a static mapping, computed using a pre-existing dataset, that yields the most statistically likely value of some unknown (conditional) variable, given some other measured (unconditional) data. For a fluid flow, such a method can produce visual representations of the flow field, but cannot suggest, without further analysis, what events should be observed or how they might be related to the underlying flow physics [35]. Furthermore, in LSE, the estimated values will lie in a subspace whose dimension is limited to the number of measurements. This is especially important when using a small number of measurements to predict a high-dimensional variable, such as a velocity field. Depending on the application, it can be either a limitation or an advantage, unnecessarily restricting the estimates or capturing only the features of interest. The use of a static map can also lead to uniqueness issues, as a particular measurement value will always yield the same estimate. For instance, a pressure sensor may measure the same value at two points in time, leading to identical estimates, even though the corresponding velocity fields are different. Increasing the number of sensors can decrease the likelihood of this happening but is not always feasible.

We note that there exist “multi-time” variants of LSE that incorporate not only current measurements, but also past ones. (See Section 5.3.1 for details on multi-time LSE.) Like standard LSE, such methods do not explicitly rely on dynamic models. However, by making use of past measurements, they in effect include dynamics. The use of multiple temporal measurements also enlarges the estimation subspace and helps in dealing with uniqueness issues (in the same way that using multiple probes does).

5.2.2 Dynamic estimation

Dynamic estimators are a fundamental topic in control theory. They estimate a system’s state using a model of its dynamics along with real-time measurement updates. The measurement updates are used to correct the trajectory of the model, which will drift from the true trajectory due to parameter uncertainty, unmodeled dynamics, and external disturbances (process noise). This is in contrast to static estimation techniques, including stochastic estimation, which use a fixed relationship to estimate the system state from a set of measurements. The static estimation approach does not take advantage of the fact that the equations governing a system’s evolution are often known.

In this work, we focus on the Kalman filter and smoother, both standard subjects in the study of estimation. (For a more in-depth discussion, see any standard text on estimation, for instance the book by Simon [154].) Suppose we are interested in the evolution of a system described by a linear model

$$\begin{aligned} \boldsymbol{\xi}_k &= \mathbf{F}\boldsymbol{\xi}_{k-1} + \mathbf{d}_k & \boldsymbol{\xi} &\in \mathbb{R}^{N_s} \\ \boldsymbol{\eta}_k &= \mathbf{H}_k\boldsymbol{\xi}_k + \mathbf{n}_k & \boldsymbol{\eta} &\in \mathbb{R}^{N_o}, \end{aligned} \tag{5.3}$$

where $\boldsymbol{\xi}$ is a vector of N_s state variables, $\boldsymbol{\eta}$ is a vector of N_o measurements of the state, \mathbf{d} represents process noise, and \mathbf{n} represents sensor noise. At any given iteration k , we assume that we can observe the measurement $\boldsymbol{\eta}_k$. From this, we would like to estimate the value of $\boldsymbol{\xi}_k$, which is otherwise unknown.

The dimension of $\boldsymbol{\eta}$ may be smaller than that of $\boldsymbol{\xi}$, meaning that even without sensor noise, the matrix \mathbf{H} relating the two may not be invertible. However, if the system is *observable*, we can use knowledge of the system dynamics \mathbf{F} and the time history of $\boldsymbol{\eta}$ to produce an estimate $\hat{\boldsymbol{\xi}}$ that converges, in the case of no noise, to the true value $\boldsymbol{\xi}$. In the presence of noise, the Kalman filter will minimize the expected value of the error

$$\left(\boldsymbol{\xi}_k - \hat{\boldsymbol{\xi}}_k\right)^T \left(\boldsymbol{\xi}_k - \hat{\boldsymbol{\xi}}_k\right).$$

The Kalman filter is a *causal* filter, meaning that only measurements made up to and including iteration k are available in forming the estimate $\hat{\boldsymbol{\xi}}_k$. In some situations, we may also have access to measurements occurring after iteration k , for instance in a post-processing application. We can use that information to improve our estimate of $\boldsymbol{\xi}_k$. This yields a *non-causal* filter, generally referred to as a *smoother*. In this work, we use a variant of the Kalman smoother developed by Rauch, Tung, and Striebel, known as the RTS smoother [cf., 154, Section 9.4.2]. The RTS smoother is a fixed-interval smoother, meaning that all measurements taken over a fixed time interval are used to estimate the state evolution within that interval. Algorithmically, it consists of a forward pass with a Kalman filter followed by a backward, smoothing pass. The specifics of the Kalman filter and RTS smoother algorithms are described in Section 5.3.2.

5.3 Numerical methods

In this section, we detail the various numerical methods used in our estimation procedure. These methods include stochastic estimation techniques and dynamic estimation techniques. (For details on modal decomposition techniques, used here to approximate the flow field and investigate flow physics, see Chapter 2 (DMD) and Appendix A (POD).) We also provide a summary of our three-step dynamic estimation procedure, laying out how the numerical methods listed above are used to form a dynamic estimator for experimental applications.

5.3.1 Modified stochastic estimation

Stochastic estimation is a means of approximating a conditional average using a knowledge of unconditional statistics. Adrian and Moin [3] proposed a stochastic estimate of the conditional average by means of a Taylor series expansion

$$\hat{\mathbf{a}}_i(t) = \langle \mathbf{a}_i(t) | \mathbf{p}_j(t) \rangle \approx \mathbf{A}_{ij} \mathbf{p}_j(t) + \mathbf{B}_{ijk} \mathbf{p}_j(t) \mathbf{p}_k(t) + \dots, \quad (5.4)$$

where \mathbf{a}_i is the i th POD coefficient, \mathbf{p}_j is the j th probe measurement, $\langle \cdot \rangle$ is the expected value, and $\hat{\mathbf{a}}_i$ is the estimate of the conditional average. The stochastic estimation coefficients \mathbf{A}_{ij} , \mathbf{B}_{ijk} , and so on are determined by minimizing the mean-square error of the estimate

$$\langle (\hat{\mathbf{a}}_i(t) - \mathbf{a}_i(t))^2 \rangle,$$

which requires solving a set of linear equations [74]. (We note that in some situations, for instance in the case of periodic flows, additional assumptions may be necessary to uniquely determine the estimation coefficients.)

The particular form of stochastic estimation given in (5.4), in which the time-varying POD coefficient is the conditional variable, is referred to as *modified* stochastic estimation. This approach can be more favorable than estimating a full PIV velocity field, typically consisting of thousands of data points, because the dominant behavior of many flows can be captured by a handful of POD modes. The estimated POD coefficients, paired with the corresponding modes, yield low-dimensional estimates of velocity fields.

Modified stochastic estimation has been applied by Bonnet et al. [24] and Taylor and Glauser [160] for linear estimates, Naguib et al. [113] and Murray and Ukeiley [112] for quadratic stochastic estimation, and Durgesh and Naughton [51] for linear estimates using time delays. Here, we use modified stochastic estimation in two ways. First, we use it to obtain initial estimates of time-resolved POD coefficients from non-time-resolved PIV measurements and time-resolved point measurements. Based on these initial estimates, it is used again to estimate model parameters that are later used as part of a dynamic estimator.

Single-time-delay modified linear stochastic estimation

In mLSE, only the linear term in (5.4) is retained. Then given the value of the probe measurements, the estimate is

$$\hat{\mathbf{a}}_i(t) = \mathbf{A}_{ij}\mathbf{p}_j(t - \tau), \quad (5.5)$$

where a constant time delay τ is introduced to account for a potential lead or lag between the conditional and unconditional variables. This increases the correlations between $\mathbf{a}(t)$ and $\mathbf{p}(t)$ for some systems [35, 74, 140]. To calculate the coefficients \mathbf{A}_{ij} , the mean-square error of the estimates must be minimized, which requires solving the equation

$$\mathbf{A}^T = [\mathbf{PP}]^{-1} [\mathbf{aP}], \quad (5.6)$$

where

$$\mathbf{A}^T = \begin{bmatrix} \mathbf{A}_{1,i} \\ \mathbf{A}_{2,i} \\ \vdots \\ \mathbf{A}_{N_p,i} \end{bmatrix}, \quad [\mathbf{PP}] = \begin{bmatrix} \overline{\mathbf{p}_1\mathbf{p}_1} & \overline{\mathbf{p}_1\mathbf{p}_2} & \cdots & \overline{\mathbf{p}_1\mathbf{p}_{N_p}} \\ \overline{\mathbf{p}_2\mathbf{p}_1} & \overline{\mathbf{p}_2\mathbf{p}_2} & \cdots & \overline{\mathbf{p}_2\mathbf{p}_{N_p}} \\ \vdots & \vdots & \ddots & \vdots \\ \overline{\mathbf{p}_{N_p}\mathbf{p}_1} & \overline{\mathbf{p}_{N_p}\mathbf{p}_2} & \cdots & \overline{\mathbf{p}_{N_p}\mathbf{p}_{N_p}} \end{bmatrix}, \quad \text{and}$$

$$[\mathbf{aP}] = \begin{bmatrix} \overline{\mathbf{a}_i\mathbf{p}_1} \\ \overline{\mathbf{a}_i\mathbf{p}_2} \\ \vdots \\ \overline{\mathbf{a}_i\mathbf{p}_{N_p}} \end{bmatrix}.$$

N_p is the number of probe measurements and time dependence in the above equations is neglected for brevity. Overbars denote time averages. (These equations are standard in the LSE literature; a derivation can be found in Durgesh and Naughton [51].)

Multi-time-delay modified linear stochastic estimation (MTD-mLSE)

The equation (5.5) is the “single time” form of mLSE. However, a single delay may increase the correlation between certain pairings of the unconditional and conditional variables but not others. In general, we can account for multiple time delays, summing the correlations over several values of τ [51, 176]:

$$\hat{\mathbf{a}}_i(t) = \mathbf{A}_{ijk} \mathbf{p}_j(t - \tau_k). \quad (5.7)$$

The use of multiple time delays, rather than a single one, is advantageous if the exact time delay is not optimal for all pairings, unknown, or not resolved well enough in time. It also effectively incorporates dynamics, as the estimates now depend on a time history of measurements. Multi-time-delay mLSE (MTD-mLSE) has been developed for purely negative time delays, requiring only past data [176], as well as for two-sided delays that also use future data [51].

The latter method is applied in this work to estimate the time-dependent POD coefficients $\mathbf{a}(t)$, and is hereafter referred to as MTD-mLSE, unless distinguished as the purely negative delay version. While using both past and future data may strengthen correlations, it comes at the cost of yielding a non-causal process. As such, two-sided MTD-mLSE cannot be used in real-time estimation or flow control applications. For a derivation of the MTD-mLSE algorithm, we refer the reader to Durgesh and Naughton [51].

We note that (5.5) provides a *static* map from the measurement $\mathbf{p}(t)$ to the estimate $\hat{\mathbf{a}}(t)$. When computing the coefficients \mathbf{A}_{ij} , we make sure to average over large datasets, mitigating the effects of sensor noise. However, in using those coefficients to compute an estimate, the static map will respond directly to the probe measurements (without averaging), making the estimates sensitive to noise. In contrast, (5.7) takes into account a history of the probe measurement. This increases the robustness of the method to sensor noise, as the inclusion of a measurement history will offset the effect of instantaneous aberrations in the probe measurement.

5.3.2 Dynamic estimation

Model identification

Our goal is to use a dynamic estimator to estimate the state of a bluff-body wake experiment. We assume that a time-resolved velocity probe signal is available, as well as PIV velocity fields captured at a slower, non-time-resolved sampling frequency. To implement a dynamic estimator, we need a model for the time evolution of the system. A high-fidelity numerical discretization of the Navier–Stokes equation is far too computationally intensive for this purpose, and would in any case be difficult to match to the experiment. As such, we develop an empirically derived, low-order model. We focus on POD-based models, as the first r POD modes optimally capture the kinetic energy contained in a set of snapshots, for any model order r .

From a large, statistically independent ensemble of PIV snapshots, we can compute a single set of well-converged POD modes. For the model identification procedure, we assume only non-time-resolved data are available. (See Section 5.4.3 for a detailed description of the particular dataset used for this computation.) We fix a desired model order r based on the energy content of the modes, which can be

determined from the POD eigenvalues. These r modes form a basis for our low-order model.

Due to noise and low spatial resolution, methods such as Galerkin projection can be difficult to apply when using experimentally acquired velocity fields. As such we take a stochastic approach in identifying a dynamic model. First, we collect a set of non-time-resolved PIV snapshots synchronously with a time-resolved probe signal. The PIV data are projected onto the POD basis to get a *non-time-resolved* set of POD coefficients $\{\mathbf{a}_{N_{ps}k}\}$, where N_{ps} is the ratio of the probe and PIV sampling rates. (We note that here, the notation \mathbf{a}_k denotes a vector of POD coefficients corresponding to a time t_k , not to be confused with the previous use of \mathbf{a}_i to denote the i th element of \mathbf{a} .) These coefficients are used along with synchronous probe measurements as “training data” to calculate the linear coefficients for MTD-mLSE, as described above in Section 5.3.1. The MTD-mLSE coefficients are then applied to the full, time-resolved probe signal, providing a set of *time-resolved estimates* of the POD coefficients, $\{\hat{\mathbf{a}}_k\}$.

We then apply LSE to these vectors, recalling that LSE estimates the expected value of a conditional variable as a linear function of an unconditional variable. If we take $\hat{\mathbf{a}}_k$ to be the conditional variable and $\hat{\mathbf{a}}_{k-1}$ to be the unconditional variable, then we can use LSE to identify a linear, discrete-time dynamical map:

$$\hat{\mathbf{a}}_k \approx \langle \hat{\mathbf{a}}_k | \hat{\mathbf{a}}_{k-1} \rangle \approx \mathbf{F}^{\text{LSE}} \hat{\mathbf{a}}_{k-1}. \quad (5.8)$$

So long as the MTD-mLSE estimates of the POD coefficients are accurate enough, then the resulting model will capture enough of the true dynamics to be used as the basis for a Kalman filter.

Finally, we note that it can be shown that the solution to the above LSE problem is the same as the least-squares/minimum-norm solution to the problem

$$\mathbf{B} = \mathbf{F}^{\text{LSE}} \mathbf{A},$$

where the columns of \mathbf{B} are the vectors $\{\hat{\mathbf{a}}_k\}_{k=1}^m$ and the columns of \mathbf{A} are the vectors $\{\hat{\mathbf{a}}_k\}_{k=0}^{m-1}$, collected over all runs. (The proof is simple and omitted here.) As such, \mathbf{F}^{LSE} can be computed by simply taking the Moore-Penrose pseudoinverse of \mathbf{A} .¹ However, the analogy to LSE provides an additional interpretation to the dynamics it defines, as it provides a linear estimate of the most statistically likely value of $\hat{\mathbf{a}}_k$ given a value of $\hat{\mathbf{a}}_{k-1}$, according to the ensemble defined by \mathbf{A} and \mathbf{B} . Based on this interpretation, this modeling procedure can naturally be extended using quadratic stochastic estimation (QSE), or even higher-order methods, for which there are no analogs to the pseudoinverse.

The bluff-body wake studied in this work is dominated by a Kármán vortex street. A computational study of a very similar flow shows that this behavior is captured well by the first two POD modes alone, which by virtue of their similarity to the dominant DMD modes, have purely oscillatory dynamics [172]. To take advantage of this knowledge in developing a model, we decouple the dynamics into two parts: an analytic, oscillatory component describing the Kármán vortex shedding, and a stochastic component describing the dynamics of all the other POD modes.

¹As mentioned at the end of Section 2.6.2, this is equivalent to linear inverse modeling.

This yields a system with dynamics

$$\hat{\mathbf{a}}_k = \begin{bmatrix} \mathbf{F}^{\text{osc}} & 0 \\ 0 & \mathbf{F}^{\text{LSE}} \end{bmatrix} \hat{\mathbf{a}}_{k-1}, \quad (5.9)$$

where \mathbf{F}^{osc} is a 2×2 matrix

$$\mathbf{F}^{\text{osc}} = \begin{bmatrix} \lambda^{\text{re}} & -\lambda^{\text{im}} \\ \lambda^{\text{im}} & \lambda^{\text{re}} \end{bmatrix}. \quad (5.10)$$

We choose $\lambda = \lambda^{\text{re}} + i\lambda^{\text{im}}$ such that $\arg(\lambda)$ is equal to the shedding frequency (identified using an autospectrum of the probe signal), and such that $\|\lambda\|$ is close to one, indicating nearly perfectly oscillatory dynamics. (In practice we choose $\|\lambda\| = 0.999$ to ensure stable dynamics.) The stochastic dynamics \mathbf{F}^{LSE} are identified using the method discussed above.

We note that in practice, the oscillatory dynamics can be captured directly by the stochastic modeling procedure. This negates the need for an a priori knowledge of the dynamics. For more complex systems, this approach may not suffice, though one could attempt to use more sophisticated system identification tools, for instance the eigensystem realization algorithm (ERA) [85], the auto-regressive Markov (AR-MARKOV) algorithm [5, 83], or observer Kalman identification (OKID) [86, 127]. However, we emphasize that the point of dynamic estimation is to leverage knowledge of a system's dynamics to estimate its state. As such, the need for a model should not be seen as a hindrance. If a model is not available, this simply indicates that dynamic estimation may not be an appropriate method for the task at hand.

Kalman filter

We use the procedure detailed in the preceding section to model the bluff-body wake as a dynamical system

$$\begin{aligned} \mathbf{a}_k &= \mathbf{F}\mathbf{a}_{k-1} + \mathbf{d}_k \\ \boldsymbol{\eta}_k &= \mathbf{H}_k\mathbf{a}_k + \mathbf{n}_k, \end{aligned} \quad (5.11)$$

where \mathbf{a} is a vector of POD coefficients, $\boldsymbol{\eta}$ is some measured quantity, \mathbf{d} is process noise, and \mathbf{n} is sensor noise. The matrix \mathbf{F} is the block diagonal matrix given in (5.9). The measurement matrix \mathbf{H}_k can be varied according to the timestep. At times when a non-time-resolved PIV snapshot is available, we choose $\mathbf{H}_k = \mathbf{I}$, allowing access to the POD coefficients of that snapshot. Otherwise, we let \mathbf{H}_k be a row vector containing the vertical velocity of each POD mode at the probe location. This makes $\boldsymbol{\eta}_k$ a POD approximation of the probe signal.

We assume that \mathbf{d} and \mathbf{n} are white, zero-mean, and uncorrelated, with covariances \mathcal{Q} and \mathcal{R} . This yields the equations

$$\begin{aligned} E[\mathbf{d}_i\mathbf{d}_j^T] &= \mathcal{Q}_i\delta_{ij} \\ E[\mathbf{n}_i\mathbf{n}_j^T] &= \mathcal{R}_i\delta_{ij} \\ E[\mathbf{d}_i\mathbf{n}_j^T] &= 0. \end{aligned}$$

Here \mathcal{Q} and \mathcal{R} are user-defined matrices, which we can consider to be design parameters. Their relative magnitudes weigh the relative accuracy of the model versus

the sensor and can be used to account for the effects of noise on the system. If we have a very noisy sensor, we want to rely more heavily on the model and we make \mathcal{R} large to penalize the sensor noise. On the other hand, if we have an inaccurate model, then we would do better by simply following the sensor, and we increase \mathcal{Q} to penalize process noise. For this particular experiment, we let the covariances \mathcal{Q} and \mathcal{R} vary in time according to which measurement is available. A higher penalty is given to the noisy probe signal, whereas the PIV data (when available) are assumed to be very accurate.

We initialize the Kalman filter with the values

$$\begin{aligned}\hat{\mathbf{a}}_{f,0}^+ &= E[\mathbf{a}_0] \\ \mathcal{P}_{f,0}^+ &= E[(\mathbf{a}_0 - \hat{\mathbf{a}}_0^+)(\mathbf{a}_0 - \hat{\mathbf{a}}_0^+)^T],\end{aligned}$$

where \mathcal{P} is the covariance of the estimation error. The filter is then updated using the following (standard) equations, for $k = 1, 2, \dots$ [154, Section 5.1]:

$$\mathcal{P}_{f,k}^- = \mathbf{F}\mathcal{P}_{f,k-1}^+\mathbf{F}^T + \mathcal{Q}_{k-1} \quad (5.12)$$

$$\mathcal{K}_{f,k} = \mathcal{P}_{f,k}^+\mathbf{H}_k^T\mathcal{R}_k^{-1} \quad (5.13)$$

$$\hat{\mathbf{a}}_{f,k}^- = \mathbf{F}\hat{\mathbf{a}}_{f,k-1}^+ \quad (5.14)$$

$$\hat{\mathbf{a}}_{f,k}^+ = \hat{\mathbf{a}}_{f,k}^- + \mathcal{K}_{f,k}(\eta_k - \mathbf{H}_k\hat{\mathbf{a}}_{f,k}^-) \quad (5.15)$$

$$\mathcal{P}_{f,k}^+ = (\mathbf{I} - \mathcal{K}_{f,k}\mathbf{H}_k)\mathcal{P}_{f,k}^- \quad (5.16)$$

Kalman smoother

The Kalman filter is a causal estimation technique, using only past and current data in forming a state estimate. In a pure post-processing application, we can make use of data at future timesteps to further improve the estimate. Such non-causal filters are referred to as *smoothers*. We focus here on fixed-interval smoothing, in which data are available over a fixed interval (here, the duration of the experiment). Specifically, we use a variant of the Kalman smoother called the Rauch–Tung–Striebel (RTS) smoother. RTS smoothing consists of a forward pass over the data using a standard Kalman filter (as described above), followed by a backward pass with the RTS smoother.

We assume that the data are available from timesteps 0 to N_t . After performing a forward pass with a Kalman filter, the smoother is initialized with the values

$$\begin{aligned}\hat{\mathbf{a}}_{s,N_t} &= \hat{\mathbf{a}}_{f,N_t}^+ \\ \mathcal{P}_{s,N_t} &= \mathcal{P}_{f,N_t}^+\end{aligned}$$

We then iterate over $k = N_t - 1, \dots, 1, 0$ [154, Section 9.4.2]:

$$\mathcal{I}_{f,k+1}^- = \left(\mathcal{P}_{f,k+1}^- \right)^{-1} \quad (5.17)$$

$$\mathcal{K}_{s,k} = \mathcal{P}_{f,k}^+ \mathbf{F}^T \mathcal{I}_{f,k+1}^- \quad (5.18)$$

$$\mathcal{P}_{s,k} = \mathcal{P}_{f,k}^+ - \mathcal{K}_{s,k} \left(\mathcal{P}_{f,k+1}^- - \mathcal{P}_{s,k+1} \right) \mathcal{K}_{s,k}^T \quad (5.19)$$

$$\hat{\mathbf{a}}_{s,k} = \hat{\mathbf{a}}_{f,k}^+ + \mathcal{K}_{s,k} \left(\hat{\mathbf{a}}_{s,k+1} - \hat{\mathbf{a}}_{f,k+1}^- \right). \quad (5.20)$$

5.3.3 Dynamic estimator implementation

As depicted by the flow chart in Figure 5.1, the goal in this work is to leverage the spatial coverage of PIV data with the temporal resolution of point measurements to improve the accuracy of reduced-order flow field estimates. The estimation procedure can be broken into three general tasks:

1. *Compute an initial set of stochastic estimates.*
 - (a) Collect PIV data synchronously with time-resolved probe measurements. The PIV data need not be time-resolved. From the PIV data, compute the dominant POD modes, for instance using (A.8) to select modes based on their energy content. This yields a set of r basis vectors $\{\phi_j\}_{j=0}^{r-1}$, to be used in approximating the flow field.
 - (b) Project the PIV vector fields onto the selected POD modes, yielding a non-time-resolved history of the POD coefficients $\mathbf{a}(t_{N_{ps}k})$, where N_{ps} is the ratio of probe and PIV sampling rates. Pair each set of POD coefficients with its corresponding set of probe measurements. Using the POD coefficients as the conditional data and a downsampled subset of the probe measurements as the unconditional data, compute the coefficients in the matrix \mathbf{A} for MTD-mLSE.
 - (c) Apply the MTD-mLSE coefficients to the full set of time-resolved probe data. This yields a time-resolved estimate of the time history of the POD coefficients, $\hat{\mathbf{a}}(t_k)$.
2. *Identify a model using the initial estimates.*

Combine physical intuition with the initial, stochastic estimates to form a dynamic model for the evolution of the POD coefficients. For instance, the procedure described in Section 5.3.2 can be used for suitable, well-behaved, oscillatory wakes.
3. *Compute an improved set of estimates using a model-based dynamic estimator.*

Use the dynamic model constructed above to implement a Kalman smoother. Apply the Kalman smoother, using the time-resolved probe data and non-time-resolved PIV velocity fields (when available) as measurement updates. The time history of the Kalman smoother's state provides a time-resolved, low-order estimate of the velocity field.

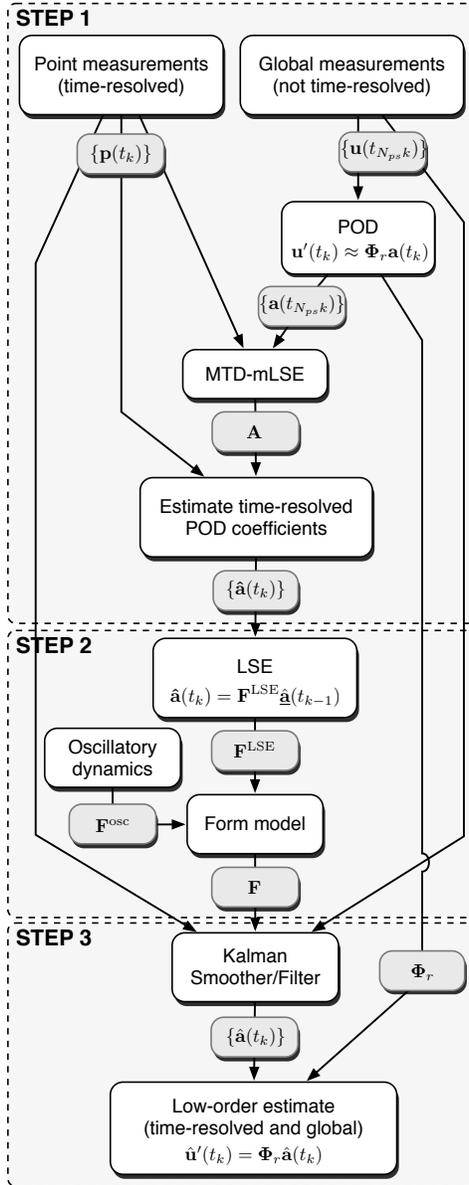


Figure 5.1: Flow chart of dynamic estimator implementation. (Figure courtesy of John Griffin.)

5.4 Experimental setup

5.4.1 Facility and instrumentation

We use TRPIV to measure the velocity in the near wake behind a flat plate model with an elliptical leading edge and blunt trailing edge. (This is the same flow configuration analyzed in Section 2.4.4.) The experiments are conducted in an Aerolab wind tunnel at the University of Florida Research and Engineering Education Fa-

cility.² This open-return, low-speed wind tunnel has a test section that measures 15.3 cm \times 15.3 cm \times 30.5 cm in height, width, and length, respectively. The test section is preceded by an aluminum honeycomb, an anti-turbulence mesh screen, and a 9:1 area-contraction section. An upstream centrifugal fan, driven by a variable frequency motor, controls the airspeed. The test section velocity is set by referencing the static and stagnation pressures from a Pitot-static tube placed at the inlet of the test section. The pressure differential is read by a Heise ST-2H pressure indicator with a 0–2 in-H₂O differential pressure transducer. For the experimental results that follow, the leading edge of the model is placed a few millimeters downstream of the test section entrance, as seen in Figure 5.2.

The two-dimensional flat plate model has a 4:1 (major axis-to-minor axis) elliptical leading edge, transitioning to a flat portion at the minor axis of the ellipse and terminating abruptly with a flat trailing edge. Unlike other two-dimensional bluff bodies with similar wake dynamics (e.g., a cylinder), the lack of surface curvature at the trailing edge simplifies the measurement of the near wake region. This geometry allows the PIV laser sheet to illuminate the entire region behind the trailing edge without mirrors or complex laser alignment. The thickness-to-chord ratio is 7.1 %, with a chord of 17.9 cm and a span of 15.2 cm. For this analysis, the freestream velocity U_∞ is set to 4.2 m/s, which corresponds to a Reynolds number of 3,600 based on the plate thickness h .

A Lee Laser 800-PIV/40G Nd:YAG system capable of up to 40 W at 10 kHz is paired with appropriate optics to generate a laser sheet for PIV measurements. As shown in Figure 5.2, the light sheet enters the test section through a clear floor. The vertically oriented light sheet is aligned with the midspan of the model and angled such that the rays of light run parallel to the trailing edge without grazing the surface. This alignment prevents unwanted, high-intensity surface reflections and is

²Experimental data acquisition and processing was done by John Griffin (University of Florida) and Adam Hart (University of Florida Research and Engineering Education Facility).

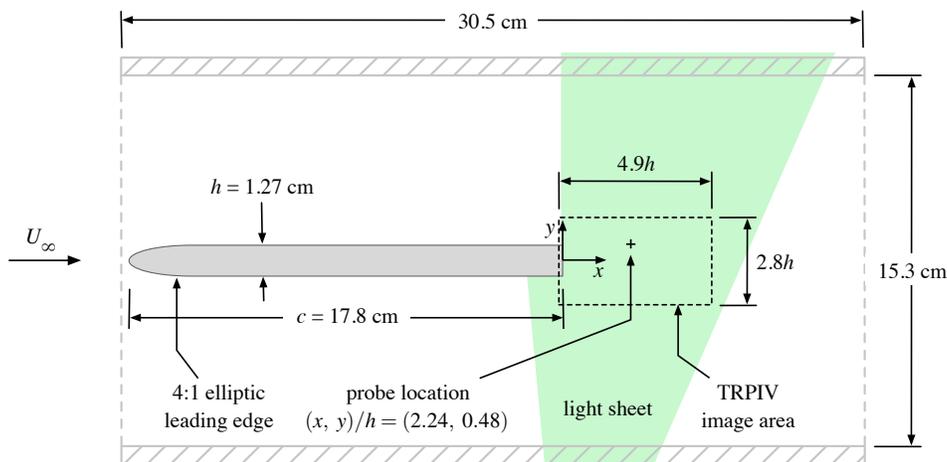


Figure 5.2: Schematic of experimental setup. A laser light sheet for PIV measurements illuminates the region behind the blunt trailing edge of a flat plate model. A probe measurement of v' is extracted from the TRPIV wake measurements at $x/h = 2.24$ and $y/h = 0.48$. (Figure courtesy of John Griffin.)

necessary for well-illuminated flow near the trailing edge, where particle densities can be low.

We image the seeded flow with an IDT MotionPro X3 camera and a 60 mm Nikon lens. The camera has a maximum resolution of $1,280 \times 1,024$ and a sampling rate of 500 Hz for integration of all pixels. A sampling frequency of 800 Hz is achieved by reducing the number of pixels captured for each image, such that the effective image resolution is $600 \times 1,024$. The laser and cameras are synchronized by a Dantec Dynamics PIV system running Dantec Flow Manager software. The seeding for the freestream flow is produced by an ATI TDA-4B aerosol generator placed upstream of the tunnel inlet. The seed material is olive oil, and the typical particle size is $1 \mu\text{m}$.

LaVision DaVis 7.2 software is used to process the PIV data, using the following procedure: first, local minimum-intensity background frames are subtracted from the raw image pairs. This step increases the contrast between the bright particles and the illuminated background by reducing the influence of static background intensities and noise bands. Then, surface regions and areas with poor particle density are masked (ignored) before computing multigrid cross-correlations. The processing consists of three passes with 64×64 pixel² interrogation windows and 75 % overlap, followed by two refining passes with 32×32 pixel² interrogation windows and 50 % overlap. In between passes, outliers are reduced by applying a recursive spatial outlier detection test [179]. The final vectors are tested for outliers via the universal outlier spatial filter [180] and the multivariate outlier detection test [71], an ensemble-based technique. Holes or gaps left by vector post-processing, which comprise less than 6 % of the total vectors over all ensembles, are filled via gappy POD [54, 112]. The final spatial resolution of the PIV measurements is approximately 8 % of the trailing edge thickness.

5.4.2 Data acquisition

We acquire ten records of TRPIV images at a rate of 800 Hz. Each record comprises nearly 1,400 sequential image pairs. To obtain a coarsely sampled (in time) set of velocity fields, we simply take a downsampled subset of the original TRPIV data. This is intended to mimic the capture rate of a standard PIV system, which for many flows is not able to resolve all of the characteristic time scales. Typical sampling rates for such commercially available systems are on the order of 15 Hz. For the estimation results that follow, one out of every 25 sequential velocity fields is used for estimation, which corresponds to a sampling rate of 32 Hz. The Nyquist frequency based on this reduced sampling rate is 16 Hz and well below the shedding frequency of about 90 Hz.

We also acquire a time-resolved probe signal by extracting the vertical velocity v from a single point in the TRPIV velocity fields. Because this probe originates from within the velocity field, the probe data are acquired synchronously with the coarsely sampled velocity fields and span the time intervals between them (Figure 5.3). This simulates the type of signal that would be measured by an in-flow sensor like a hot-wire probe. However, in-flow probes are intrusive and may interfere with attempts to take simultaneous PIV measurements, in addition to potentially disturbing the natural flow. Furthermore, such probes are not feasible in real-world applications. To emulate a more realistic flow control setup, other experiments similar to this one

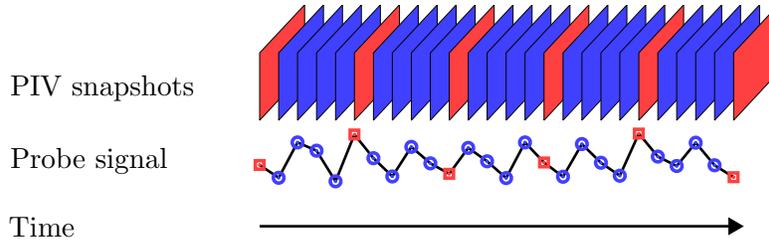


Figure 5.3: Cartoon of data acquisition method. Probe data is collected synchronously with TRPIV snapshots. The TRPIV are downsampled to get a non-time-resolved dataset (*red*). This subset of the TRPIV data is used to develop static and dynamic estimators. (The cartoon does not depict actual sampling rates.)

have used non-intrusive, surface-mounted pressure sensors to perform stochastic estimation [51, 111], as well as Kalman filter-based dynamic estimation [121, 122]. Based on their success, the methods developed here can likely be extended to work with surface-mounted sensors as well.

The dynamic estimators in this work rely, at least partially, on the correlation between point measurements and the time-varying POD coefficients. As such, the time-resolved probe measurements must correlate to structures within the flow field in order for the estimation to work properly. Consequently, the outcome of the estimation can be sensitive to the placement of the sensors [35]. Motivated by the work of Cohen et al. [34], we place our sensor at the node of a POD mode (see Figure 5.2 for the sensor location). In particular, we choose the point of maximum v -velocity in the third POD mode (Figure 5.6 (d)), as a heuristic analysis determined that the dynamics of this mode were the most difficult to estimate.

5.4.3 Data partitioning

Here, we describe the division of the TRPIV data into two partitions: one for *implementation* and the other for *validation*. The first partition consists of five TRPIV records that we use to implement the estimation procedure described in Section 5.3.3. We refer to these records as “training sets.” The remaining five records are reserved for error analysis and validation of the resulting dynamic estimator.

There are only two instances in which we make use of the full TRPIV records. The first is in the POD computation, where the time-resolved aspect of the records is actually not utilized. The key assumption here is that the POD modes computed from the time-resolved velocity fields are the same as those generated from randomly sampled velocity fields. This is valid in the limit of a large, statistically independent snapshot ensemble. For the remainder of the estimator implementation, we consider only the downsampled subset of the training sets. The second place that we use time-resolved velocity fields is in our error analysis. Here, we make use of the 800 Hz TRPIV data as a basis of comparison for our estimates of the time-resolved velocity field.

5.5 Results and discussion

The results of the experiment described in Section 5.4 are discussed below. This discussion is broken into two main parts. First, we analyze the dynamics of the wake flow, using POD, DMD, and standard spectral analysis methods. An effort is made to identify key characteristics of the wake, including the dominant frequencies and any coherent structures. In doing so, we allow ourselves access to the TRPIV velocity fields, taken at 800 Hz.

Then the PIV data are downsampled, leaving snapshots taken at only 32 Hz. These velocity fields are combined with time-resolved point measurements of velocity (again at 800 Hz) for use in estimating the time-resolved flow field. We compare the results of MTD-mLSE on the POD coefficients to those of dynamic estimation using a Kalman smoother. As a proof of concept, we also investigate causal implementations of the estimators, which are necessary for flow control applications.

5.5.1 Wake characteristics

Global/modal analysis

At a thickness Reynolds number of $Re_h = 3,600$, the wake behind the flat plate displays a clear Kármán vortex street, as seen in Figure 5.4. POD analysis (of the first four training set records, out of five total) shows that 79.6 % of the energy in the flow is captured by the first two modes (Figure 5.5). Each subsequent mode contributes only a small fraction more energy, with the first seven modes containing 85.0 % in total. As such, for the remainder of this analysis, we take these seven modes as our POD basis. (Though seven modes are required to accurately describe the state, wake stabilization may be possible using fewer [61, 88].)

The structure of the dominant modes, illustrated in Figure 5.6 (b), (c), demonstrates that they capture the dominant vortex shedding behavior. The lower-energy modes also contain coherent structures, though without further analysis, their physical significance is unclear. All but the third mode (Figure 5.6 (d)) resemble modes computed by Tu et al. [172] for a simulation of a similar flow. However, the modes computed here are not all perfectly symmetric or antisymmetric, as might be expected [41, 115]. While it is possible to enforce symmetry by expanding the snapshot

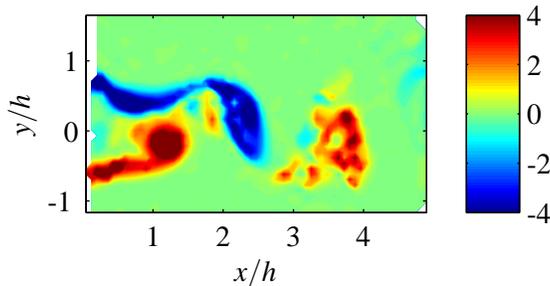


Figure 5.4: Instantaneous spanwise vorticity field computed from PIV data (scaled by the ratio of the freestream velocity U_∞ to the plate thickness h). A clear Kármán vortex street is observed.

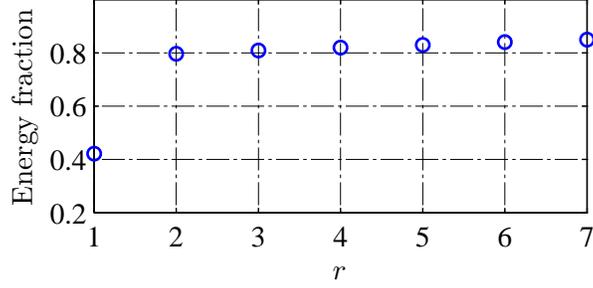


Figure 5.5: Energy content of the first r POD modes (i.e., of a POD basis $\{\phi_j\}_{j=0}^{r-1}$).

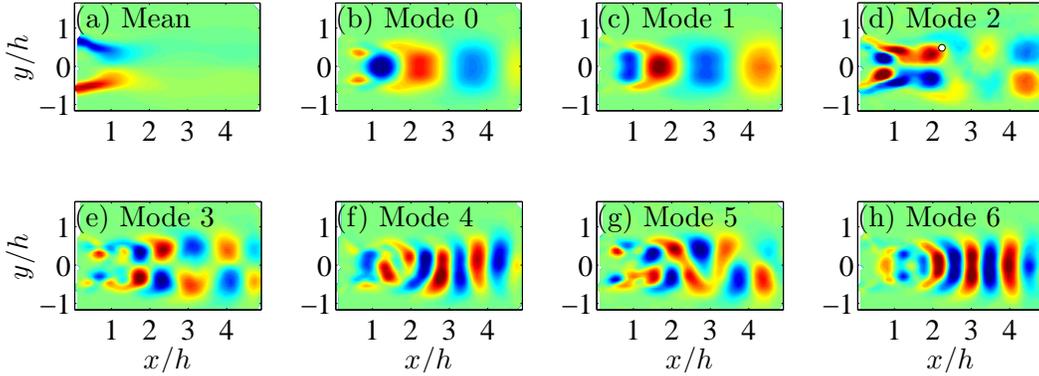


Figure 5.6: Spanwise vorticity of POD modes computed from TRPIV fields. The modes are arranged in order of decreasing energy content. Most resemble modes computed by Tu et al. [172] in a computational study of a similar flow. (a) Mean flow; (b), (c) dominant shedding modes; (d) unfamiliar structure, with v -velocity probe location marked by (o); (e), (g) antisymmetric modes; (f), (h) spatial harmonics of dominant shedding modes.

ensemble [155], we choose not to do so, taking the lack of symmetry in the modes to indicate a possible lack of symmetry in the experiment.

DMD analysis of the time-resolved velocity fields (from the first training set record) reveals that the flow is in fact dominated by a single frequency. The spectrum shown in Figure 5.7 has a clear peak at a Strouhal number $St_h = 0.27$ (based on U_∞ and h), with secondary peaks at the near-superharmonic frequencies of 0.52 and 0.79. The dominant frequency is in good agreement with that measured by Durgesh and Naughton [51]. The corresponding DMD modes (Figure 5.8) show structures that resemble the POD modes discussed above. Because DMD analysis provides a frequency for every spatial structure, we can clearly identify the harmonic nature of the modes, with the modes in Figure 5.8 (a) corresponding to the fundamental frequency, those in Figure 5.8 (b) corresponding to its first superharmonic, and those shown in Figure 5.8 (c) corresponding to its second superharmonic.

Furthermore, because DMD identifies structures based on their frequency content, rather than their energy content (as POD does), these modes come in clean pairs. Both DMD and POD identify similar structures for the dominant shedding modes (Figures 5.6 (b), (c), 5.8 (a)), but the superharmonic pairs identified by

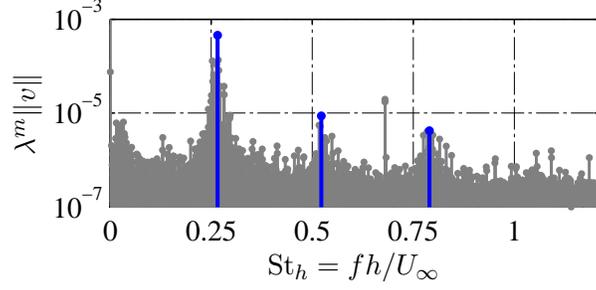


Figure 5.7: DMD spectrum computed from TRPIV velocity fields. Clear harmonic structure is observed, with a dominant peak at $St_h = 0.27$, followed by superharmonic peaks at 0.52 and 0.79. Peaks corresponding to the modes shown in Figure 5.8 are highlighted in blue.

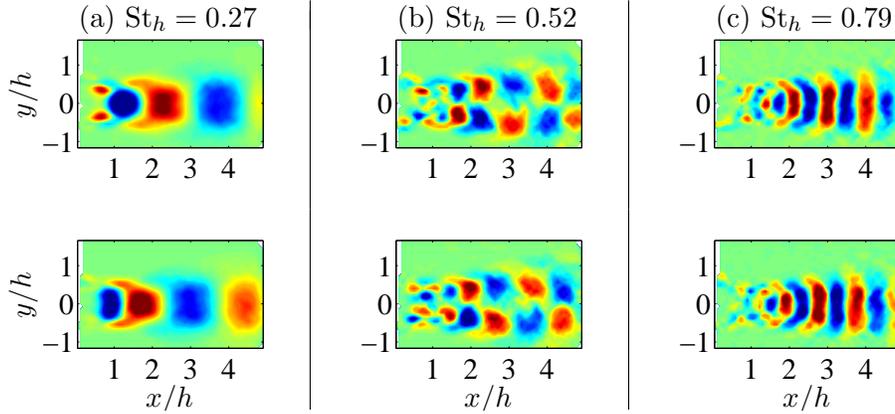


Figure 5.8: Spanwise vorticity of DMD modes computed from TRPIV velocity fields. The real and imaginary components are shown in the *top* and *bottom* rows, respectively. Note the similarity of these modes to the POD modes depicted in Figure 5.6. (a) Dominant shedding modes; (b) temporally superharmonic modes, spatially anti-symmetric; (c) further superharmonic modes, spatial harmonics of dominant shedding modes.

DMD do not match as well with their closest POD counterparts. For instance, the POD mode shown in Figure 5.6 (e) closely resembles the DMD modes shown in Figure 5.8 (b), whereas the mode shown in Figure 5.6 (g) does not. Similarly, Figure 5.6 (h) depicts a mode resembling those in Figure 5.8 (c), while Figure 5.6 (f) does not.

Interestingly, the third POD mode (Figure 5.6 (d)) is not observed as a dominant DMD mode. This suggests that the structures it contains are not purely oscillatory, or in other words, that it has mixed frequency content. As such, its dynamics are unknown a priori. This is in contrast to the other modes, whose dynamics should be dominated by oscillations at harmonics of the wake frequency, based on their similarity to the DMD modes. This motivates our placement of a velocity probe at the point of maximum v -velocity in the third POD mode [34], in an effort to better capture its dynamics. This location is shown in Figures 5.2 and 5.6 (d).

Point measurements

Figure 5.9 shows a time trace of the probe signal collected in the flat plate wake. We recall that there is no physical velocity probe in the wake. Rather, we simulate a probe of v -velocity by extracting its value from the TRPIV snapshots (see Figure 5.2 or 5.6 (d) for the probe location). Because PIV image correlations are both a spatial average across the cross-correlation windows and a temporal average over the time interval between image laser shots, PIV probe measurements typically do not resolve the fine scale structures of turbulence. To simulate a more realistic probe, Gaussian white noise is artificially injected into this signal, at various levels. We define the noise level γ as the squared ratio of the root-mean-square (RMS) value of the noise to the RMS value of the fluctuating probe signal:

$$\gamma = \left(\frac{n'_{\text{RMS}}}{v'_{\text{RMS}}} \right)^2, \quad (5.21)$$

where the prime notation indicates a mean-subtracted value. This noise level is the reciprocal of the traditional signal-to-noise ratio. Note that the noise level only reflects the amount of artificially added noise and does not take into account any noise inherent in the velocity probe signal. We consider six noise levels, ranging from 0.01 to 0.36, in addition to the original signal ($\gamma = 0$), focusing on the extreme noise levels in the following discussion. Figure 5.9 shows a comparison of the original signal to artificially noisy signals. We see that though the addition of noise produces random fluctuations, the dominant oscillatory behavior is always preserved.

A spectral analysis of the probe data, seen in Figure 5.10, confirms that the shedding frequency is still detected in the presence of the artificially added noise. This is to be expected, as the addition of white noise only adds to the broadband spectrum. The dominant peaks in the autospectra lie at $St_h = 0.27$, in agreement with the dominant DMD frequency. This confirms our previous characterization of the dominant DMD (and POD) modes as structures capturing the dominant vortex shedding in the wake.

The autospectra also show clear harmonic structure, again confirming the behavior seen in the DMD spectrum. However, as the broadband noise levels increase,

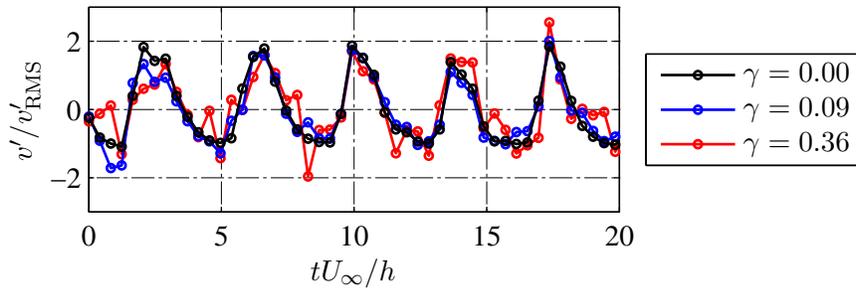


Figure 5.9: Measurement of v' taken in the wake at $x/h = 2.24$ and $y/h = 0.48$, the point of maximum v in the third POD mode (Figure 5.6d). The values are extracted from the TRPIV snapshots. Noise is artificially injected to simulate a physical velocity sensor, with the noise level γ defined in Equation (5.21). In all cases, clear oscillatory behavior is observed.

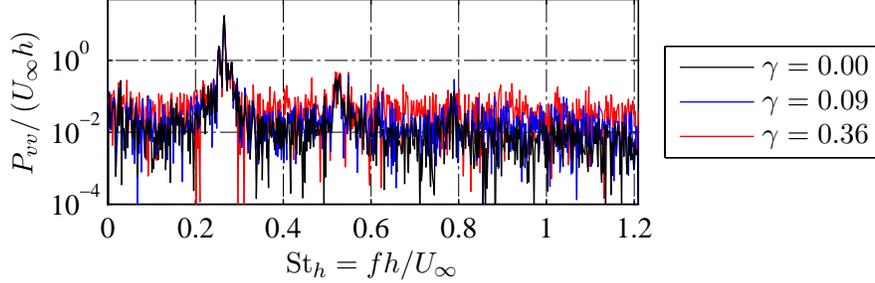


Figure 5.10: Autospectra of probe signals shown in Figure 5.9. Clear harmonic structure is observed, with a dominant peak at $St_h = 0.27$, which agrees well with the dominant DMD frequency. Superharmonic structure is also seen, again confirming behavior observed using DMD analysis (compare with Figure 5.7).

the third and fourth harmonics of the wake frequency become less prominent relative to the noise floor. This loss of harmonic structure carries certain implications for estimation. Most notable is that the fluctuations associated with these higher harmonics do not correlate as strongly with the time-varying POD coefficients. Consequently, the flow field estimates based on the noisy probe signals may not capture the corresponding harmonic structures as well. The inclusion of noise is designed to be a test of estimator robustness. Future work will apply the same general static and dynamic estimators presented here, but with pressure and shear stress sensors, which are inherently noisy (perhaps in a non-Gaussian way).

5.5.2 Low-dimensional flow-field estimation

Optimal delay interval for MTD-mLSE

We find that with $\tau^* = 0$, MTD-mLSE estimation of the first two POD coefficients is poor, unless multiple probes are used.³ Here, τ^* is the non-dimensional time delay, defined as

$$\tau^* = \tau U_\infty / h. \quad (5.22)$$

This follows the results of Durgesh and Naughton [51], who conducted a very similar bluff-body wake experiment. The cause lies in the fact that there is often a phase difference between the probe signal and the time history of one or more of the POD coefficients, decreasing the magnitude of the LSE cross-correlations.

Durgesh and Naughton [51] were able to significantly improve their estimates by using MTD-mLSE, which accounts for possible phase mismatches. For that reason, we use the same method in this work. In this variant of MTD-mLSE, to estimate the flow field at a given moment in time, we make use of probe data collected prior to that moment, as well as after. That is, to estimate the flow field at time t , we use probe data collected at times $t \pm \tau_j$, for a range of delays τ_j satisfying

$$\tau_j^* \leq \tau_{\max} U_\infty / h. \quad (5.23)$$

Durgesh and Naughton [51] also demonstrated that an optimum bound τ_{\max}^* exists for estimating the unknown POD coefficients. In order to determine the optimal

³Stochastic estimation computations were performed by John Griffin (University of Florida).

value, an estimation error must be computed and evaluated. For the present study, the flow field is approximated using the first seven POD modes. The corresponding vector $\mathbf{a}(t_k)$ of POD coefficients encodes a low-dimensional representation of the velocity field at time t_k , with a corresponding kinetic energy $\|\mathbf{a}(t_k)\|_2^2$ (see (A.7)). We wish to quantify the error between the true coefficients \mathbf{a}_k and the estimated POD coefficients $\hat{\mathbf{a}}(t_k)$. One way to do so is to simply compute the kinetic energy contained in the difference of the two corresponding velocity fields. If we normalize by the mean kinetic energy of the snapshot set, this gives us an error metric

$$\begin{aligned} e(t_k) &= \frac{\|\hat{\mathbf{a}}(t_k) - \mathbf{a}(t_k)\|_2^2}{\langle \|\mathbf{a}(t_k)\|_2^2 \rangle} \\ &= \frac{\sum_{i=1}^r [\hat{\mathbf{a}}_i(t_k) - \mathbf{a}_i(t_k)]^2}{\sum_{i=1}^r \langle \mathbf{a}_i(t_k)^2 \rangle}. \end{aligned} \quad (5.24)$$

This can be interpreted as the fraction of the mean kinetic energy contained in the estimation error.

In finding the optimal delay interval for MTD-mLSE, we use only downsampled PIV data (from the first four training set records) to compute the MTD-mLSE coefficients. The estimation error is then evaluated by taking another PIV record (the fifth training set record) and estimating its POD coefficients $\{\hat{\mathbf{a}}_k\}$. These other PIV velocity fields are also projected onto the POD modes to get their true coefficients $\{\mathbf{a}_k\}$, which we then compare to the estimated coefficients. The mean energy in the error \bar{e} is calculated from these coefficients for values of τ_{\max}^* ranging from 0 to 12. The results are plotted in Figure 5.11.

The minimum value of \bar{e} occurs for a delay interval with $\tau_{\max}^* = 2.48$. However, we note that due to the shallow valley around the minimum seen in Figure 5.11, similar estimator performance is expected for delays ranging between 1.7 and 3.0 (roughly). Note that the case of $\tau_{\max}^* = 0$ empirically demonstrates that mLSE without any time delay yields poor estimates in this experiment (as suggested by theory).

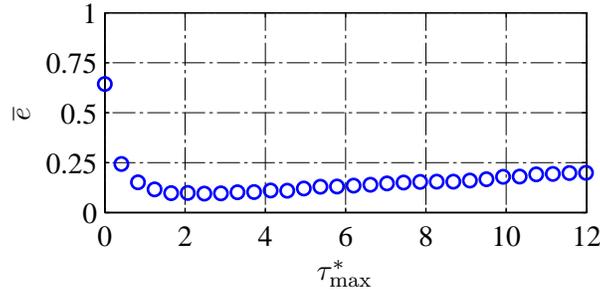


Figure 5.11: Mean energy in the MTD-mLSE estimation error, for various symmetric time delay intervals. An optimal value of τ_{\max}^* is observed.

Kalman smoother design

The model for the Kalman smoother is identified using the method described in Section 5.3.2. We recall that this model consists of two decoupled submodels. The first describes the dynamics of the two dominant POD modes, which are assumed to be oscillatory. Figure 5.10 shows autospectra computed from the time-resolved probe data, which we use to determine the oscillation frequency. The second submodel describes the dynamics of the remaining five modes and is identified using the initial set of MTD-mLSE estimates.

Once the model has been obtained, the Kalman smoother is initialized with the values

$$\begin{aligned}\hat{\mathbf{a}}_{f,0}^+ &= \mathbf{a}_0 \\ \mathcal{P}_{f,0}^+ &= 5\mathbf{I},\end{aligned}$$

where \mathbf{I} is the identity matrix. We assume that the initial set of POD coefficients \mathbf{a}_0 is known, as this is a post-processing application where the PIV data are available at certain (non-time-resolved) instances. We use the noise covariances as tuning parameters and take their values to be

$$\mathcal{Q}_k = \begin{bmatrix} 1 & & & & 0 \\ & 1 & & & \\ & & 0.004 & & \\ & & & 0.5 & \\ 0 & & & & \ddots \end{bmatrix}$$

and

$$\mathcal{R}_k = \begin{cases} 2 \times 10^4 & \text{when only probe data are available} \\ 1 \times 10^{-10}\mathbf{I} & \text{when PIV data are available.} \end{cases}$$

(These values are chosen based on tests of the Kalman smoother using the training set data.) We heavily penalize the time-resolved velocity signal to mitigate the effects of noise (large \mathcal{R}_k), while the PIV data are assumed to be very accurate relative to the model (small \mathcal{R}_k). In addition, the diagonal matrix \mathcal{Q}_k is designed to account for the observation that the lower-energy POD modes are more sensitive to noise in the probe signal, with the third mode more sensitive than the rest.

Error analysis

We now compare the performance of two estimators: a static MTD-mLSE estimator with an optimal time delay interval and a dynamic Kalman smoother, both described above. We apply each estimator to five PIV records designated for estimator validation. (These records were not used in any way in the development of the estimators.) The estimates of the POD coefficients are evaluated using the error metric e defined in (5.24).

The aggregate results are shown in Figure 5.12. By definition, e is non-negative, giving it a positively skewed distribution. As such, the spread of these values is not correctly described by a standard deviation, which applies best to symmetric

distributions. To account for this, the error bars in Figure 5.12 are adjusted for the skewness in the distribution of e [81]. We observe that for all noise levels, the mean error achieved with a Kalman smoother is smaller than that of the MTD-mLSE estimate. Furthermore, the rate of increase in the mean error is slower for the Kalman smoother than for the MTD-mLSE estimate, and the spread is smaller too. As such, we conclude that not only does the Kalman smoother produce more accurate estimates (in the mean), but it is also more robust to noise.

This robustness comes in two forms. The first is that for a given amount of noise in the signal, the expected value of the estimation error has a wider distribution for MTD-mLSE than for a Kalman smoother. Secondly, as the noise level is increased, the MTD-mLSE estimation error increases more rapidly, indicating a higher sensitivity to the noise level. This is expected, as MTD-mLSE does not have access to the PIV updates; it is making estimates using less data than the Kalman smoother.

The time evolution of the estimates further emphasizes this advantage. Figure 5.13 (a) compares the true history of the second POD coefficient to the corresponding MTD-mLSE and Kalman smoother estimates, for the worst-case noise level $\gamma = 0.36$. We see that for this particular mode, the Kalman smoother estimates are generally more accurate, deviating less from the true coefficient values. In particular, there is very little error during the instances surrounding a PIV update. This is even more obvious when we consider the evolution of e , which incorporates the errors in all seven mode coefficients (Figure 5.13 (b)). Here, we see that for the Kalman smoother alone, local minima in the error line up with the availability of PIV data, indicated by the dashed, vertical lines. (While decreases in the stochastic estimation error sometimes line up with PIV updates, this trend is not observed in general.)

Again, this is not unexpected, as the stochastic estimates are computed using only the probe signal. In contrast, the Kalman smoother also assimilates PIV velocity fields when they are available, driving the error to nearly zero at each assimilation step. The effects of this improvement are felt for many timesteps following and prior to the PIV update. As such, it is clear that a driving factor in the improved performance of the Kalman smoother is its ability to take advantage of information that MTD-mLSE cannot, in the form of infrequently available PIV velocity fields.

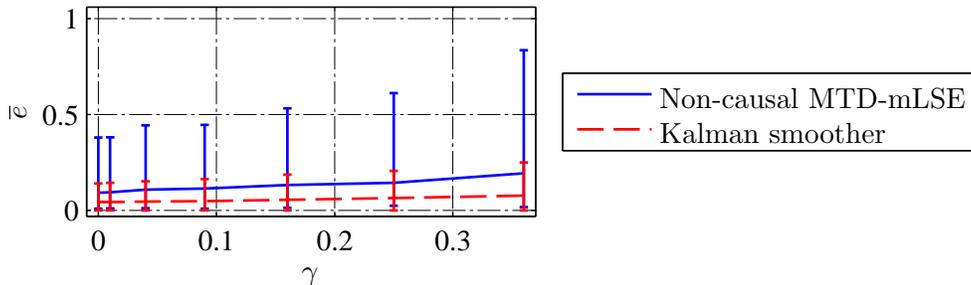


Figure 5.12: Mean and distribution of the energy in the estimation error, for various levels of sensor noise (as defined in Equation (5.24)). The error energy is normalized with respect to the average energy contained in the (mean-subtracted) velocity field. The Kalman smoother estimates are both more accurate and less sensitive to noise.

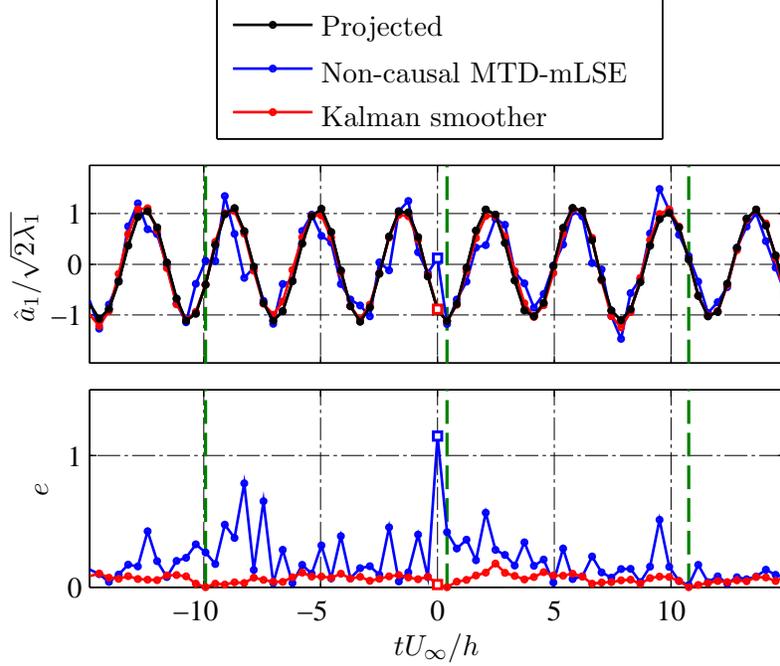


Figure 5.13: (a) Time history of the second POD coefficient ($j = 1$), along with the corresponding Kalman smoother and MTD-mLSE estimates; (b) time history of the energy in the estimation error. The *vertical, dashed* lines mark times when PIV data are available. The *square symbols* mark the time instance depicted in Figures 5.14 and 5.15. We observe that the Kalman smoother estimates are more accurate overall. Specifically, local minima in the error occur at instances of PIV data assimilation. The MTD-mLSE estimates do not make use of these data, and thus do not show the same general behavior.

These results are confirmed by comparing the estimated vorticity fields, for both $\gamma = 0$ and $\gamma = 0.36$. Figure 5.14 shows an instantaneous vorticity field and its projection onto the first seven POD modes. (This particular instance in time is denoted by square markers in Figure 5.13.) This projection is the optimal representation of the original vorticity field using these POD modes. We observe that the high-energy structures near the trailing edge are captured well, while the far wake structures tend to be more smoothed out. With no noise, the MTD-mLSE estimate of the vorticity field (Figure 5.15 (a)) matches the projected snapshot quite well. The spacing and shape of the high-energy convecting structures in the Kármán vortex street are correctly identified. However, when the probe signal is contaminated by noise with $\gamma = 0.36$, the estimated vorticity field shown in Figure 5.15 (b) bears little resemblance to the projection. In fact, the only structures that match are features of the mean flow (Figure 5.6 (a)). Not only are the downstream structures captured poorly, but spurious structures are also introduced. On the other hand, the Kalman smoother estimates match the projected snapshot for both clean and noisy probe data (Figure 5.15 (c), (d)).

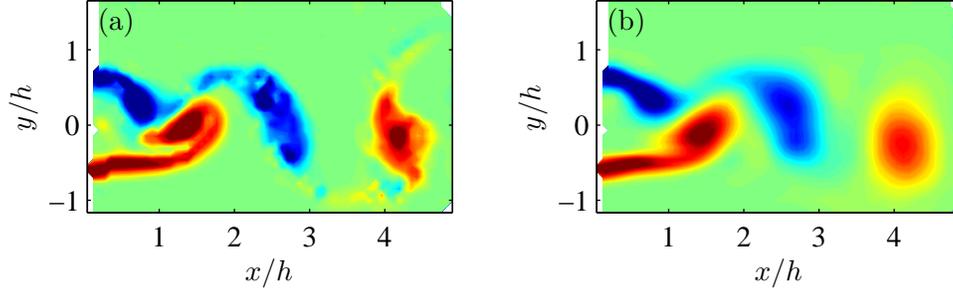


Figure 5.14: Comparison of spanwise vorticity fields. (a) True PIV snapshot; (b) projection onto a seven-mode POD basis. The first seven POD modes capture the location and general extent of the vortices in the wake, but cannot resolve small-scale features.

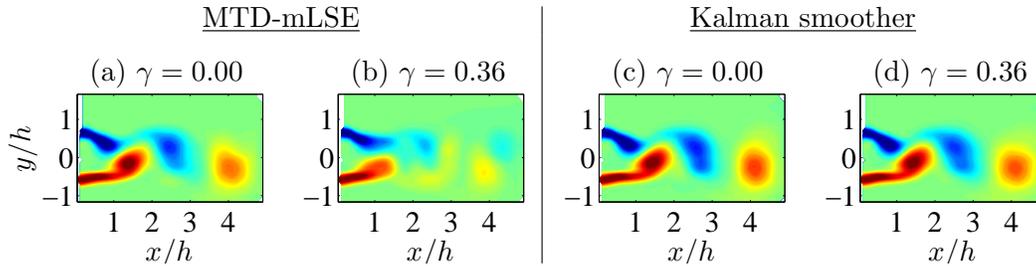


Figure 5.15: Comparison of estimated spanwise vorticity fields. Without noise, both the MTD-mLSE and Kalman smoother estimates match the POD projection shown in Figure 5.14b. The addition of noise to the probe signal causes the MTD-mLSE estimate to change dramatically, resulting in a large estimation error. In contrast, the Kalman smoother estimate remains relatively unchanged. (a) MTD-mLSE with $\gamma = 0$; (b) MTD-mLSE with $\gamma = 0.36$; (c) Kalman smoother with $\gamma = 0$; (d) Kalman smoother with $\gamma = 0.36$.

Estimation-based global/modal analysis

As a further investigation into the relative merits of MTD-mLSE and Kalman smoother estimation, we use the estimated velocity fields to perform DMD analysis. We recall that DMD analysis requires that the Nyquist-Shannon sampling criterion is met, for which the sampling rate must be at least double the highest frequency of interest. The DMD modes from the true TRPIV data are shown in Figures 5.7 and 5.8. The key results from the DMD analysis of the estimated flow fields (for both the MTD-mLSE and Kalman smoother estimates) are shown in Figure 5.16. Only the minimum and maximum noise levels are considered in this modal analysis.

The fundamental frequency $St_h = 0.27$ is captured well by estimation-based DMD for both estimators, for both noise levels. The corresponding modes match as well, and illustrations are therefore omitted. (Refer to Figure 5.8 (a) for typical mode structures associated with $St_h = 0.27$.) For the superharmonic frequencies, however, the estimation-based DMD modes differ in structure, both among the

various estimation cases (across methods, for varying noise levels) and in relation to the DMD modes computed directly from TRPIV data (Figure 5.8).

As seen in Figure 5.16, both the MTD-mLSE and Kalman smoother estimates capture the first superharmonic ($St_h \approx 0.53$) well when no noise is added to the probe signal. However, the Kalman smoother-based mode more accurately captures the expected antisymmetric distribution seen in Figure 5.8. When the noise level is increased to 0.36, both of the estimate-based modes deviate from the corresponding TRPIV-based mode, but less so for the Kalman smoother. This is not unexpected, as the Kalman smoother estimates are less sensitive to the addition of noise (Figure 5.12).

In contrast, both estimators perform poorly in capturing the DMD mode corresponding to the second superharmonic ($St_h \approx 0.79$). Without any artificial noise, the MTD-mLSE and Kalman smoother-based modes are similar to each other and bear some resemblance to the expected mode shape. However, for $\gamma = 0.36$ neither displays the correct vorticity distribution nor captures the right frequency. This decreased accuracy for higher harmonics is not unexpected, as the corresponding fluctuations in the probe signals correlate less and less with the POD coefficients as the noise floor increases.

We note that because our estimates are limited to a subspace spanned by only seven POD modes, so are any estimate-based DMD computations. That is, any behavior not captured by the first seven POD modes will not be captured by estimate-based DMD analysis either. Because the dominant POD modes correspond to the

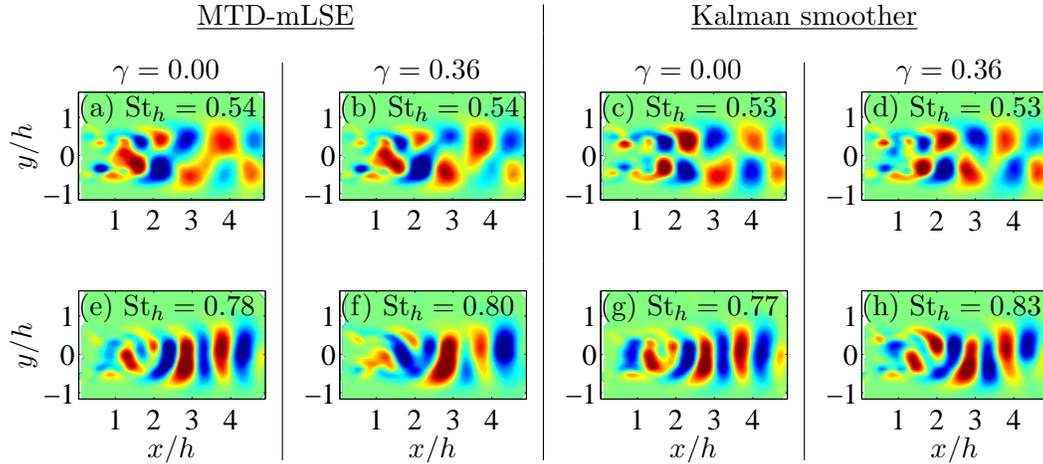


Figure 5.16: Estimation-based DMD modes. Computations of the first and second superharmonic wake modes are shown on the *top* and *bottom* rows, respectively. In general, the Kalman smoother results more closely resemble those shown in Figure 5.8. For both estimation methods, using a noisier probe signal leads to poorer results. This is especially pronounced for the second superharmonic mode. (a) MTD-mLSE with $\gamma = 0$; (b) MTD-mLSE with $\gamma = 0.36$; (c) Kalman smoother with $\gamma = 0$; (d) Kalman smoother with $\gamma = 0.36$; (e) MTD-mLSE with $\gamma = 0$; (f) MTD-mLSE with $\gamma = 0.36$; (g) Kalman smoother with $\gamma = 0$; (h) Kalman smoother with $\gamma = 0.36$.

highest-energy structures, the estimate-based DMD analysis will also be biased toward high-energy, and typically low-frequency, fluctuations. In this work, we observe that the dominant POD modes are quite similar to the dominant DMD modes. As such, it is no surprise that estimate-based DMD computations are successful in identifying the fundamental shedding mode and its harmonics.

5.5.3 Causal implementation

The Kalman smoother and MTD-mLSE methods discussed previously are non-causal, requiring future data to estimate the state. This makes them unsuitable for applications that require real-time estimates, such as estimation-based flow control. However, both methods have clear causal counterparts. We recall that the RTS Kalman smoother algorithm consists of a forward, Kalman filter estimation followed by a backward, smoothing correction. By simply not performing the smoothing operation and limiting ourselves to a Kalman filter, we can perform a causal, dynamic estimation that integrates time-resolved point measurements with non-time-resolved PIV snapshots. Similarly, the MTD-mLSE coefficients can easily be computed using one-sided delays only, eliminating the use of future data. We note that in most applications, online processing of PIV velocity fields is currently not feasible, due to computational limitations. However, such systems do exist, though they are generally limited to acquisition rates on the order of 10 Hz [11, 187]. This makes an accurate estimation procedure, which estimates the state of the system between the slow PIV updates, even more critical.

Figure 5.17 shows that overall, the causal implementations are more error-prone than the non-causal ones (compare to Figure 5.12). However, the same trends are observed in comparing the dynamic and stochastic estimates: dynamic estimation yields a lower mean error, a narrower error distribution, and a slower increase in the error with respect to γ . As before, this is not surprising, as the dynamic estimator make uses of not only the point measurements, but also full-field PIV data (when available).

When no online PIV system is available, then the best we can do is to estimate the state using the time-resolved point measurements alone. Figure 5.18 shows that when a Kalman filter is implemented without access to any PIV information, the mean estimation error is nearly the same as that of the MTD-mLSE estimator. For

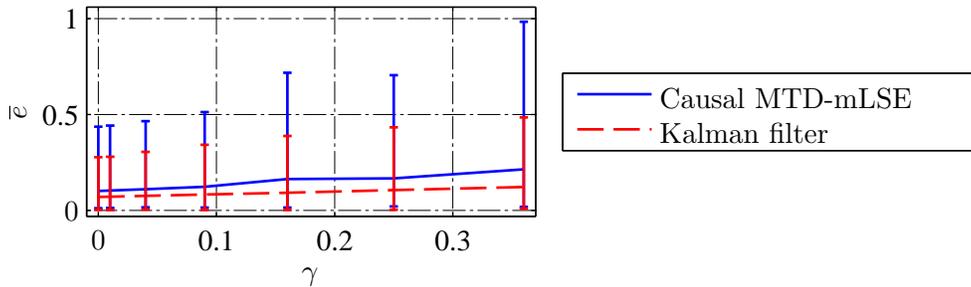


Figure 5.17: Mean and distribution of the energy in the estimation error for causal estimation. The same trends are observed as in the non-causal estimation (see Figure 5.12), with the Kalman filter estimates more accurate and less sensitive to noise than the MTD-mLSE estimates.

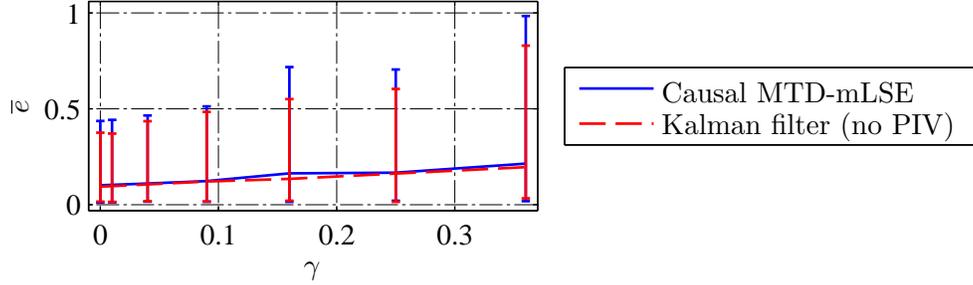


Figure 5.18: Mean and distribution of the energy in the estimation error for causal estimation using only point measurements. Without access to any PIV data, the Kalman filter is only marginally more effective than MTD-mLSE. The only benefit is a slight decrease in the sensitivity to noise.

any particular noise level, the distribution is smaller, but only marginally so. The increase in error with respect to noise level is comparable for both methods. As such, we can see that the assimilation of PIV measurements provides a significant benefit, even though it only occurs on relatively slow time scales.

5.6 Conclusions

The difficulty of acquiring time-resolved velocity fields is one of the main challenges in applying DMD to experimental data. In this chapter, we showed that dynamic estimation is a viable approach in dealing with this problem, as the three-step estimation procedure presented here proves to be effective in estimating the time-resolved velocity field of a bluff-body wake. Rather than estimate the flow field directly using MTD-mLSE, we use MTD-mLSE to aid in identifying a stochastic model for the lower-energy structures in the flow. This stochastic model is then combined with an analytic model of the dominant vortex shedding in the wake. The result is used to implement a Kalman smoother, whose estimates of the flow field are shown to be more accurate and robust to noise than the stochastic estimates used in the modeling process. A DMD analysis of the Kalman smoother estimates identifies the same coherent structures observed in an analysis of TRPIV data, showing that the estimates correctly capture the oscillatory dynamics of the flow. Similar trends are observed for a Kalman filter implementation, which would be suitable for flow control, whereas the Kalman smoother is limited to post-processing applications.

A potential limitation of this approach is that it requires the availability of a low-order model of the flow physics. For sufficiently complex flows, obtaining such a model may be nontrivial, or even prohibitively difficult. In the next chapter, we present an alternative approach to solving the time resolution issue, making use of ideas from compressed sensing. By turning to a statistical approach, we eliminate the need for a dynamic model. However, the compressed sensing framework presents its own unique challenges.

Chapter 6

Towards a compressed DMD algorithm

As discussed in the previous chapter, in order for dynamic mode decomposition (DMD) to provide meaningful results, it must be applied to data collected twice as fast as any frequency of interest [117, 152]. This is often not possible in practice, particularly for fluid flow experiments. (See Chapter 5 for details.) In this chapter, we deal with this limitation by combining ideas from DMD and compressed sensing. Given a vector-valued signal, we take measurements randomly in time (at a sub-Nyquist rate) and project the data onto a low-dimensional subspace. We then use compressed sensing/sparse approximation algorithms to identify the dominant frequencies in the signal and their corresponding modes. The result is an algorithm that, like DMD, produces spatial modes that each correspond to a single frequency, doing so using sub-Nyquist-rate data.

We demonstrate this method using two examples. First, we test the method using a canonical dataset. For these simple data, the dominant frequencies and modes are identified using ℓ_1 minimization techniques. Second, we apply the method to particle image velocimetry (PIV) data collected from the flow past a two-dimensional cylinder. For these more complex data, we identify the dominant frequencies and modes using a greedy algorithm called orthogonal matching pursuit (OMP) [167, 169]. In each case, our method correctly identifies the characteristic frequencies and oscillatory modes dominating the signal.

The remainder of this chapter is structured as follows: Section 6.1 provides an introduction to compressed sensing concepts, motivating the amalgamation of DMD and compressed sensing concepts. In Section 6.2 we describe the implementation of our numerical method. Examples demonstrating the capabilities of the method are presented in Section 6.3. Finally, we summarize our results and discuss directions for future work in Section 6.4.

This work was inspired by conversations with Professor J. Nathan Kutz (University of Washington), who also provided guidance as the project proceeded. The two-dimensional cylinder experiment was run by Jessica Shang (Princeton University) and Professor Alexander J. Smits (Princeton University).

6.1 Motivation

Many dynamical systems exhibit oscillatory behavior; fluid mechanical systems are no exception. As demonstrated in Chapter 4, DMD is an effective tool for analyzing such systems. Not only can DMD identify characteristic flow frequencies, but the corresponding modes may elucidate features of the underlying fluid mechanics. Unfortunately, DMD requires data that satisfy the Nyquist-Shannon sampling criterion [117, 152], which may not always be available in practice. In Chapter 5, we dealt with this limitation by using a dynamic estimator to estimate a time-resolved history of the fluid velocity field, performing DMD not on the raw data, but on the velocity field estimates. This procedure required identifying a model for the flow field dynamics. For flows more complex than the bluff body wake discussed in Chapter 5, it may be unclear how to identify such a model.

In the signal processing community, there has been a growing emphasis on dealing with time resolution issues using a method called “compressed sensing” [25, 27, 45].¹ Compressed sensing relies on the fact that many signals of interest are actually sparse in frequency space. If we sample such signals randomly in time, then we can reconstruct them accurately using ℓ_1 minimization techniques or greedy algorithms, even if the samples are taken at a sub-Nyquist-rate. (For a review of compressed sensing theory, we refer the reader to [18] and [26].) This approach has proven successful in a number of applications, including dynamic MRI [57, 98], facial recognition [182], imaging [46, 136], and radar [77, 129].

While typically applied to scalar-valued signals, compressed sensing algorithms readily extend to vector-valued signals. As such, in theory these methods could be applied directly to PIV data collected from fluids experiments. However, in reality this is not feasible; the fine resolution needed to accurately resolve pertinent flow features makes PIV data too large for standard compressed sensing algorithms. But despite their frequent representation as high-dimensional vectors, many fluid flows actually evolve in a low-dimensional subspace.

We propose a method for computing oscillatory modes from sub-Nyquist rate PIV data that combines concepts from DMD and compressed sensing. Recall from Section 2.2 that DMD is closely related to proper orthogonal decomposition (POD): the DMD modes are linear combinations of POD modes and the DMD eigenvalues come from the POD projection of a linear operator. Here, we use POD projections to represent high-dimensional PIV data using low-dimensional vectors. We then perform compressed sensing on these vectors of POD coefficients, lifting the resulting modes to the original space by taking linear combinations of POD modes, just as in DMD. Not only does this method rely on POD in the same way that DMD does, but POD bases are also optimal for reconstructing datasets (see Appendix A for more details on POD).

We demonstrate this method through two extended examples. In the first, we construct a canonical dataset in which we superpose two Gaussian spatial fields oscillating at different frequencies. We add noise to the signal to test the robustness of our method. By construction, the signal is almost exactly sparse (it is heavily dominated by two frequencies), so a compressed sensing approach is reasonable. However, we choose the amplitudes of these Gaussian fields such that the resulting

¹Compressed sensing is also known as “compressive sampling” or “compressive sensing” and is closely related to “sparse approximation”/“sparse reconstruction”/“sparse recovery” methods.

POD modes mix together the oscillatory structures. In the second example, we apply our method to experimental PIV data collected from the flow past a two-dimensional cylinder. This flow is dominated by a single frequency (the cylinder shedding frequency), but the data are not precisely sparse, only approximately so. Each of these examples poses different challenges for our method, but in both cases, we successfully identify the correct frequencies and modes.

6.2 Numerical method

In this section we introduce the basic concepts of compressed sensing and describe how we apply those concepts to compute oscillatory spatial modes from sub-Nyquist-rate data. First, we describe how compressed sensing can be used to reconstruct scalar signals. Then we show how it can be extended to vector-valued signals. For high-dimensional vectors, standard compressed sensing methods may be inefficient; we propose the use of POD to reduce the dimension of the problem and avoid this bottleneck. Finally, we describe strategies for collecting data samples suitable for compressed sensing and then summarize our numerical method.

6.2.1 Scalar signals

The field of compressed sensing has undergone astounding growth since the foundational works by Donoho [45], Candes et al. [27], and Candes and Tao [25] were published in 2006. We review the key concepts here. (For more a more in-depth introduction to these topics, we refer the reader to [18] and [26].) Consider a signal $\mathbf{f} \in \mathbb{R}^n$. For instance, \mathbf{f} could consist of n sequential measurements taken from a hot-wire velocity probe. We assume that these measurements are taken at a rate such that \mathbf{f} captures all dynamics of interest.

Typically, \mathbf{f} will not be sparse in the standard basis for \mathbb{R}^n (consisting of $(1, 0, 0, \dots)$, $(0, 1, 0, \dots)$, and so on). That is, a large number of these basis vectors are required to accurately describe \mathbf{f} . We say that \mathbf{f} is *compressible* if there exists a basis Ψ such that the representation of \mathbf{f} in Ψ is approximately sparse. Specifically, we say that \mathbf{f} is k -sparse in the basis Ψ if

$$\mathbf{f} = \Psi \hat{\mathbf{f}}, \tag{6.1}$$

where $\Psi \in \mathbb{R}^{n \times n}$ and $\hat{\mathbf{f}} \in \mathbb{R}^n$, with $\hat{\mathbf{f}}$ having only k nonzero values. (The less precise descriptor “compressible” requires only that $\hat{\mathbf{f}}$ have few large coefficients relative to the number of small ones.) The potential for savings is clear: rather than storing n values to describe the signal \mathbf{f} , we can get away with storing only the k nonzero elements of $\hat{\mathbf{f}}$. This is the principle on which JPEG-2000 compression is built [26, 159].

Now suppose that we do not have access to the full signal \mathbf{f} . Instead, all we know is an m -dimensional linear measurement

$$\mathbf{g} = \Phi^T \mathbf{f}, \tag{6.2}$$

where Φ is an $n \times m$ matrix. We can think of the columns of Φ as waveforms that we use to measure \mathbf{f} . For instance, if Φ contains sinusoids, then \mathbf{g} contains Fourier coefficients.

We are interested in the case where $m \ll n$, i.e., the undersampled case, for which (6.2) is underdetermined. As such, we cannot solve for \mathbf{f} from a knowledge of \mathbf{g} ; the solution, if it exists, is not unique. But suppose we substitute for \mathbf{f} using (6.1), giving us

$$\mathbf{g} = \Phi^T \Psi \hat{\mathbf{f}}. \quad (6.3)$$

Though $\hat{\mathbf{f}}$ is also an n -dimensional vector, it only has k nonzero elements. A standard approach in compressed sensing is to determine $\hat{\mathbf{f}}$ by solving the following optimization problem:

$$\min_{\hat{\mathbf{f}} \in \mathbb{R}^n} \|\hat{\mathbf{f}}\|_1 \quad \text{subject to} \quad \mathbf{g} = \Phi^T \Psi \hat{\mathbf{f}}. \quad (6.4)$$

Thus out of all vectors $\hat{\mathbf{f}}$ that are consistent with our measurement \mathbf{g} , we are interested in finding the one with the smallest ℓ_1 norm.

We choose the ℓ_1 norm because it promotes sparsity and can be solved using a linear program. Compared to the more common ℓ_2 norm, used for instance in solving least-squares problems, the ℓ_1 norm more harshly penalizes nonzero elements in $\hat{\mathbf{f}}$, which we know to be a sparse vector. In theory we would like to minimize the ℓ_0 norm of $\hat{\mathbf{f}}$, but that minimization problem is NP-incomplete and numerically unstable [18]. As such, we use the ℓ_1 norm as a computationally tractable proxy.

It was shown in [45] and [27] that in some cases, solving (6.4) can recover $\hat{\mathbf{f}}$ exactly (if it is k -sparse) or very accurately (if it is compressible). Much of the compressed sensing literature deals with finding conditions on Φ and Ψ for which these results hold. For instance, the columns of Φ and Ψ should be chosen to be maximally *incoherent*. Many proofs also rely on $\Phi^T \Psi$ obeying the *restricted isometry property*. These topics are outside the scope of this discussion and furthermore are most applicable to situations in which we have freedom to choose the measurement matrix. In this work, we restrict ourselves to the case that Ψ describes a Fourier basis and that Φ is a subset of the standard basis (see Section 6.2.2 for more details). A more relevant theoretical result is that solving (6.4) yields the best k -sparse approximation to $\hat{\mathbf{f}}$ even if $\hat{\mathbf{f}}$ is not exactly k -sparse (e.g., if it is only compressible) [26]. Furthermore, this procedure is robust to measurement noise [26].

Closely related to compressed sensing is the field of *sparse approximation*. Just as in compressed sensing, the goal of sparse approximation methods is to find the best sparse representation of a k -sparse or compressible signal. However, rather than solve an ℓ_1 minimization problem, sparse approximation methods make use of greedy algorithms. These algorithms are iterative: upon each iteration they add another basis vector (column of Ψ) to the support of $\hat{\mathbf{f}}$. By construction, the resulting estimate of $\hat{\mathbf{f}}$ will be sparse, as j iterations will yield j nonzero basis coefficients; the rest are assumed to be zero. There are many different greedy algorithms used for sparse approximation. In this chapter, we deal only with orthogonal matching pursuit (OMP), due to its simplicity [167, 169]. (CoSaMP is a similar algorithm that is also popular [114].) The theoretical guarantees on OMP are similar to those for compressed sensing: under certain technical conditions (again outside the scope

of this work), OMP exactly reproduces k -sparse vectors and closely approximates compressible ones [167].

6.2.2 Choice of basis, measurement

Much of the theoretical research on compressed sensing deals with characterizing matrices Φ and Ψ for which the method will succeed. In this work, we are motivated by practical concerns, and as such are restricted in our choices of Φ and Ψ . Because we are concerned with oscillatory temporal behavior, we choose Ψ such that \mathbf{f} contains Fourier coefficients. From (6.1), we see that this means Ψ is the matrix representation of the inverse discrete Fourier transform (DFT).

For ease of implementation, we assume that our measurement \mathbf{g} simply corresponds to values of \mathbf{f} sampled at particular instances in time. Suppose that \mathbf{f} corresponds to a fast hot-wire probe signal. The first element of \mathbf{f} is the value of the probe signal at time $t = 0$. The second element is the value at $t = \Delta t$, the third element corresponds to $t = 2\Delta t$, and so on. Now suppose that for our measurement \mathbf{g} , we sample our probe signal at $t = 0$. Then the first column of Φ is $(1, 0, 0, 0, \dots)^T$. If we wait until $t = 2\Delta t$ to get our next sample, then the second column of Φ is $(0, 0, 1, 0, \dots)^T$. Thus we see that Φ is a subset of the standard basis: each column contains only zeroes, except for one entry with value 1. We can think of our measurement waveforms as Dirac delta functions. As it turns out, delta functions and sinusoids are maximally incoherent, an important property for compressed sensing to work [26].

6.2.3 Vector-valued signals

In this work we are concerned with vector-valued signals $\mathbf{F} \in \mathbb{R}^{n \times p}$. (In the compressed sensing literature, such signals are referred to as “multiple-measurement vectors” [29, 38, 52, 100, 168, 170].) As before, n corresponds to the number of temporal measurements; p is the number of values measured at a given instant in time. If \mathbf{F} corresponds to a rake of hot-wire sensors, then p is the number of hot-wires. If \mathbf{F} corresponds to PIV velocity fields, then each field is reshaped into a row vector and p is the number of grid points in the velocity field multiplied by the number of velocity components measured (typically two). In this case, we observe that rows (of \mathbf{F}) correspond to points in time and columns to points in space.

We assume that there exists a basis Ψ in which the representation of \mathbf{F} is sparse. Since \mathbf{F} is a matrix, we must be careful in defining what we mean by “sparse.” For a vector-valued signal, we rewrite (6.1) as

$$\mathbf{F} = \Psi \hat{\mathbf{F}}, \quad (6.5)$$

where $\hat{\mathbf{F}} \in \mathbb{R}^{n \times p}$. For the simple case $p = 1$, for which (6.5) reduces to (6.1), sparsity requires that there \mathbf{f} have few large elements. When $p > 1$, the elements of \mathbf{f} correspond to rows of \mathbf{F} , so we require that there be few rows of \mathbf{F} with large norm. Letting \mathbf{G} be a vector-valued measurement analogous to \mathbf{g} , we can rewrite (6.4) as

$$\min_{\hat{\mathbf{F}} \in \mathbb{R}^{n \times p}} \|\hat{\mathbf{F}}\|_{1,q} \quad \text{subject to} \quad \mathbf{G} = \Phi^T \Psi \hat{\mathbf{F}}, \quad (6.6)$$

where $\mathbf{G} \in \mathbb{R}^{m \times p}$ and the mixed norm $\|\cdot\|_{1,q}$ of a matrix \mathbf{M} is defined as

$$\|\mathbf{M}\|_{1,q} \triangleq \sum_{i=0}^{n-1} \left| \left(\sum_{j=0}^{p-1} |\mathbf{M}_{i,j}|^q \right)^{1/q} \right|. \quad (6.7)$$

This norm can be interpreted as taking the ℓ_q norm of each row, stacking these values in a vector, and then taking the ℓ_1 norm of the vector of ℓ_q norms. The choice of q weights the relative importance of nonzero entries that occur in the same row versus those that occur in different ones. For instance, if $q = 1$, then we have an ℓ_1 equivalent of the Frobenius norm for matrices and all nonzero elements are penalized equally. However, in some applications we may expect that only a few rows of $\hat{\mathbf{F}}$ will contain nontrivial entries, but within those rows we may have no expectation of sparsity. In this case we would choose $q > 1$ to decrease the penalty on nonzero elements within rows. (For other examples demonstrating the use of compressed sensing with mixed norms, see [29, 38, 52, 100, 168, 170].) Recall from 6.2.2 that in this work we choose Ψ to be the DFT basis. Then each row of $\hat{\mathbf{F}}$ corresponds to a particular frequency. Our notion of row sparsity is then natural, as it corresponds to a signal dominated by a small number of frequencies.

We note that one could theoretically perform compressed sensing on the columns of \mathbf{F} individually, treating each as a scalar signal. Each computation would yield a sparse coefficient vector $\hat{\mathbf{f}}$. However, there would be no guarantee that the sparse elements would occur in the same entries across computations. For a Fourier basis, that means that while each computation would identify a small number of dominant frequencies, these frequencies might vary from computation to computation. This is an advantage of the vector-valued approach: sparsity is enforced using all of the data simultaneously.

6.2.4 Efficiency through POD projection

In practice, solving the optimization problem (6.6) can be computationally prohibitive when the matrix \mathbf{F} is large. For PIV velocity fields, the dimension p corresponds to the number of grid points multiplied by the number of velocity components. This can easily exceed 10^5 , which is the case for the data analyzed in Chapter 5. Fortunately, many fluid flows evolve in relatively low-dimensional subspaces. We can take advantage of this to make compressed sensing feasible for PIV data.

Consider a vector-valued signal \mathbf{F} where each row corresponds to a PIV velocity field. (We reshape each velocity field into a row vector and concatenate each velocity component to get a single vector describing the entire flow field.) The transposed matrix \mathbf{F}^T is often referred to as a “snapshot” matrix in the DMD and POD literature, as each of its columns describes a snapshot of the flow field at an instant in time.² We can compute the POD modes of \mathbf{F}^T using the method of snapshots, as described in Section 2.3.1. The projection of \mathbf{F}^T onto the first r POD modes is

²We note the convention in fluid mechanics is that each *column* of the snapshot matrix corresponds to an instant in time. For compressed sensing, the convention is reversed: each *row* of the signal matrix corresponds to a particular instant.

given by

$$\mathbb{P}_r \mathbf{F}^T = \mathbf{U}_r \mathbf{U}_r^T \mathbf{F}^T, \quad (6.8)$$

where $\mathbf{U}_r \in \mathbb{R}^{p \times r}$ is a matrix whose columns are POD modes. We refer to the $r \times n$ matrix

$$\mathbf{A}_r^T \triangleq \mathbf{U}_r^T \mathbf{F}^T \quad (6.9)$$

as the matrix of POD coefficients.³ We can project a vector-valued measurement \mathbf{G} in the same way, yielding the POD coefficient matrix

$$\mathbf{B}_r^T = \mathbf{U}_r^T \mathbf{G}^T, \quad (6.10)$$

where $\mathbf{B}_r \in \mathbb{R}^{m \times r}$. (Recall, n is the number of time points, m is the number of samples in time, p is the size of the data vector, e.g., the number of grid points, and r is the number of POD modes.)

Since \mathbf{F} has a sparse representation in Ψ , so should \mathbf{A}_r ; \mathbf{A}_r describes the same behavior in a different coordinate system. Then we can write

$$\mathbf{A}_r = \Psi \hat{\mathbf{A}}_r, \quad (6.11)$$

where $\hat{\mathbf{A}}_r \in \mathbb{R}^{n \times r}$, and we can apply compressed sensing to \mathbf{A}_r by solving

$$\min_{\hat{\mathbf{A}}_r \in \mathbb{R}^{n \times r}} \|\hat{\mathbf{A}}_r\|_{1,q} \quad \text{subject to} \quad \mathbf{B}_r = \Phi^T \Psi \hat{\mathbf{A}}_r. \quad (6.12)$$

By using the mixed norm $\|\cdot\|_{1,q}$, we enforce row sparsity, meaning that only a few rows of $\hat{\mathbf{A}}_r$ should contain nontrivial values.

Recall that we are interested in computing spatial modes that correspond to oscillatory frequencies. We can find such modes by using $\hat{\mathbf{A}}_r$ to linearly combine the POD modes. The matrix $\hat{\mathbf{A}}_r$ has rows that each correspond to a frequency and columns that each correspond to a POD mode. Then each column of the product $\mathbf{U}_r \hat{\mathbf{A}}_r^T$ is a spatial field corresponding to a particular frequency. These are the equivalent of DFT modes, computed using sub-Nyquist-rate data.

In an abstract way, this method is quite similar to DMD. Recall that DMD is closely related to POD, with the DMD modes computed as a linear combination of POD modes (see Remark 1 in Section 2.2). The rest of the DMD procedure can be considered a computation to determine the proper coefficients for this linear combination. The result is a set of modes that each correspond to a particular frequency (and growth rate). Similarly, by construction, the columns of $\mathbf{U}_r \hat{\mathbf{A}}_r^T$ are linear combinations of POD modes. We can consider the compressed sensing procedure as a computation to determine the right coefficients for this linear combination. However, unlike DMD, because we assume that Ψ is a DFT basis, the compressed sensing modes are purely oscillatory; there are no growth rates.

6.2.5 Sampling strategy

The allure of compressed sensing is that it can somehow circumvent the Nyquist-Shannon sampling criterion. A key requirement is that the signal of interest must be compressible, but this is not uncommon; many signals are dominated by a few

³Again, we use the compressed sensing convention: *rows* of \mathbf{A}_r correspond to instants in time.

characteristic frequencies. The other unique aspect of compressed sensing is that it relies on random sampling. Results on whether or not a compressible signal can be recovered generally focus on the number of measurements required, with no constraints on the sampling other than that it is random. However, not all random sampling strategies will work. For instance, if we happen to sample a scalar signal at only zero crossings, then we have $\mathbf{g} = \mathbf{0}$, and there is obviously not enough information to reconstruct a nontrivial signal. Furthermore, in practice a truly random sampling may not be possible due to physical constraints; the minimum time between samples is a common limitation.

In this work we develop sampling strategies based on physical intuition. We assume the elements of the nominal signal \mathbf{f} correspond to times $t = 0$, $t = \Delta t$, $t = 2\Delta t$, and so on, with Δt small enough that \mathbf{f} captures all dynamics of interest. (For simplicity we refer to scalar signals in this discussion, though everything extends to vector-valued signals.) Motivated by applications to PIV data, we assume that the closest any two samples can be in time is $s_{\min}\Delta t$. We know from the Nyquist-Shannon sampling criterion that if we sample \mathbf{f} at a fixed rate corresponding to $s_{\min}\Delta t$, we may alias the signal and be unable to recover any oscillations with frequencies faster than $1/(2s_{\min}\Delta t)$. Thus, we do not expect that any (even if random) subset of those samples will suffice for compressed sensing.

Instead, we assume that though we have a minimum separation between samples, we have enough accuracy to sample any element of \mathbf{f} , so long as it is not within s_{\min} samples of the previous one. That is, we are not interested in the fastest possible uniform sampling, which is given by data collected at times $t = 0$, $t = s_{\min}\Delta t$, $t = 2s_{\min}\Delta t$, and so on. Rather, we make use of the fact that we can collect data at $t = t^*$ and $t = t^* + (s_{\min} + j)\Delta t$ for any j . Intuitively, this allows us to sample all phases of our signal, even though we cannot do so in a frequency-resolved manner. Applying this strategy in a random manner (letting j vary randomly), the sampled signal should contain as much information as a truly random sampling (as is usually considered in compressed sensing).

We propose two possible sampling strategies based on this intuition:

1. Minimum/maximum separation strategy
 - (a) Define a minimum separation between samples s_{\min} .
 - (b) Define a maximum separation between samples s_{\max} .
 - (c) Sample the signal \mathbf{f} such that the time between samples is given by $j\Delta t$, where j is random and uniformly distributed between s_{\min} and s_{\max} .
2. Perturbed clock strategy
 - (a) Define a nominal separation between samples s_{avg} .
 - (b) Define the maximum allowable perturbation in the sample separation s_{pert} .
 - (c) Sample the signal \mathbf{f} such that the time between samples is given by $(s_{\text{avg}} + j)\Delta t$, where j is random and uniformly distributed between $-s_{\text{pert}}$ and s_{pert} .

In both strategies, the parameters can be adjusted so that no samples are collected faster than allowed by the maximum sampling rate. The parameters should be

chosen such that the maximum spacing between samples is as large as $1/f_{\min}$, where f_{\min} is a characteristic slow frequency. This ensures that all phases of the signal can be sampled.

6.2.6 Summary of method

We summarize the steps of our method here.

1. Select a random sampling strategy (see Section 6.2.5).
2. Use this strategy to generate a “chirp signal” of 1’s and 0’s, where a 1 corresponds to a time when a sample should be collected. When the chirp signal has value 0, no data should be collected.
3. Set up a triggering system such that data are only collected when the value of the chirp signal is 1.
4. Collect data according to the chirp signal.
5. Compute POD modes from the data.
6. Choose a set of r POD modes to represent the data, for instance setting a threshold for the amount of energy captured by the modes (as in Chapter 5). This defines the matrix \mathbf{U}_r .
7. Project the data onto the POD modes, resulting in a matrix of sampled POD coefficients \mathbf{B}_r .
8. Solve the optimization (6.12), where n is determined by the time elapsed between the first and last data samples.
9. Compute the compressed sensing modes as the columns of $\mathbf{U}_r \hat{\mathbf{A}}_r^T$.

We note that for especially long signals (large n), the optimization (6.12) can be replaced with a greedy algorithm such as OMP. In that case, only the nontrivial rows of $\hat{\mathbf{A}}_r$ will be computed, but the computation of the compressed sensing modes as a linear combination of POD modes is unchanged.

6.3 Results

In this section we present two extended examples that demonstrate the capabilities of the method described above. The first deals with a numerical dataset that we construct, designed to test various features of our method. The second applies our method to data collected from a fluid flow experiment. In both cases, we are able to correctly identify the characteristic frequencies and oscillatory modes that dominate the signal of interest, using only sub-Nyquist-rate samples.

6.3.1 Canonical dataset

The vast literature on compressed sensing leaves very little doubt that ℓ_1 minimization and greedy algorithms can in fact reconstruct compressible signals. Thus the features of our method that require verification are the sampling strategy and the use of a POD projection. As a test, we consider a dataset of the form

$$\mathbf{f}(t) = \sin(\omega_1 t)\mathbf{v}_1 + \sin(\omega_2 t)\mathbf{v}_2 + 0.1\mathbf{n}(t). \quad (6.13)$$

We choose frequencies $\omega_1 = 1.3$ and $\omega_2 = 8.48$ and draw the elements of \mathbf{n} independently from a uniform distribution on the open interval $(0, 1)$. The vectors \mathbf{v}_1 and \mathbf{v}_2 are the oscillatory spatial modes that we want to recover using compressed sensing. For illustrative purposes, we choose the Gaussians

$$\mathbf{v}_1 = 2 \exp\left(-\frac{(x-0.5)^2}{2(0.6)^2} - \frac{(y-0.5)^2}{2(0.2)^2}\right)$$

$$\mathbf{v}_2 = \exp\left(-\frac{(x+0.25)^2}{2(0.6)^2} - \frac{(y-0.35)^2}{2(1.2)^2}\right),$$

where x and y are spatial coordinates. Figure 6.1 shows a visualization of these modes.

By construction, this dataset is compressible, consisting of two dominant oscillations and low-amplitude broadband noise. It is thus suitable for compressed sensing. Furthermore, we see from Figure 6.2 that the POD modes are not aligned with the oscillatory ones. Rather, each POD mode combines features of both oscil-

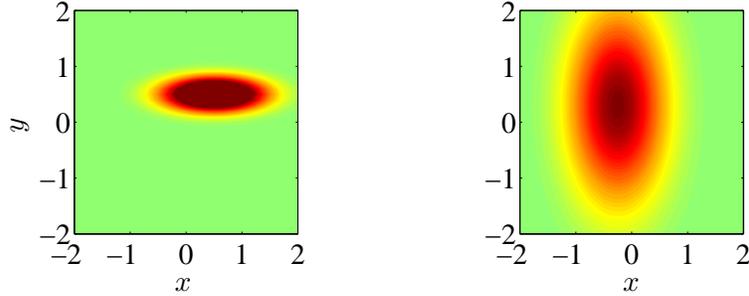


Figure 6.1: True oscillatory modes for canonical dataset. (Left) \mathbf{v}_1 ; (right) \mathbf{v}_2 .

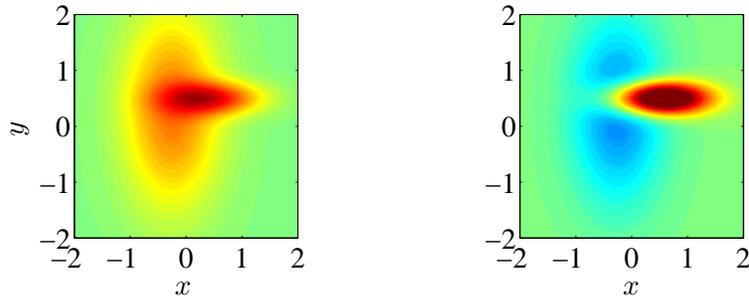


Figure 6.2: POD modes of canonical dataset.

latory modes. This is by design; for our method to work properly, it must correctly combine the POD modes such that these features are correctly isolated.

We generate a nominal signal \mathbf{F} whose columns are given by $\mathbf{f}(k\Delta t)$ for $k = 0, 1, \dots, n-1$, with $\Delta t = 0.05$ and $n = 8001$. The signal is sampled with a minimum spacing $s_{\min} = 60$ and a maximum spacing $s_{\max} = 75$, resulting in 117 total samples. Since the fastest frequency in the signal is $\omega_2 = 8.48$ and the underlying timestep is $\Delta t = 0.05$, the sample spacing that satisfies the Nyquist-Shannon sampling criterion is $s_{\text{Nyq}} = 7$. Thus we see that we are at best sampling at eight times slower than required by the Nyquist-Shannon sampling criterion. Figure 6.3 shows a plot of $\sin(8.48t)$ overlaid with points corresponding to the random sampling. It is clear that using traditional techniques, there is not enough data to reconstruct the original signal.

Figure 6.4 shows the result of solving (6.12), using the software package `cvx` [66, 67] to compute $\hat{\mathbf{A}}_r$. We see that using compressed sensing, we correctly identify the two dominant frequencies, with less than 2.5 % error in each case. As expected, the rest of the frequencies have negligible energy, a result of the ℓ_1 minimization. In addition, we see in Figure 6.5 that the correct oscillatory modes are identified. There are some aberrations, but for the most part the compressed sensing modes resemble the original Gaussians (Figure 6.1), rather than the POD modes shown in Figure 6.2. We note that because this method relies on random sampling, if we

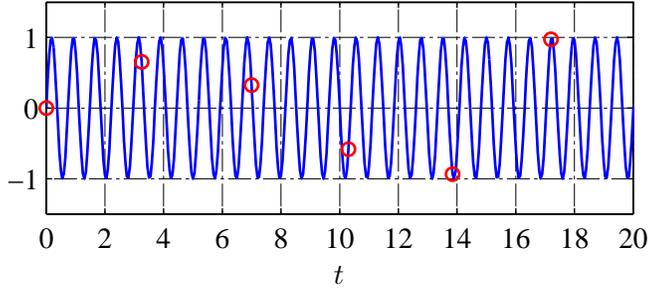


Figure 6.3: Random sampling for canonical dataset. The first six sample points are plotted over a sine wave with frequency $\omega_2 = 8.48$. Clearly, the sample points do not resolve the fastest oscillations in (6.13).

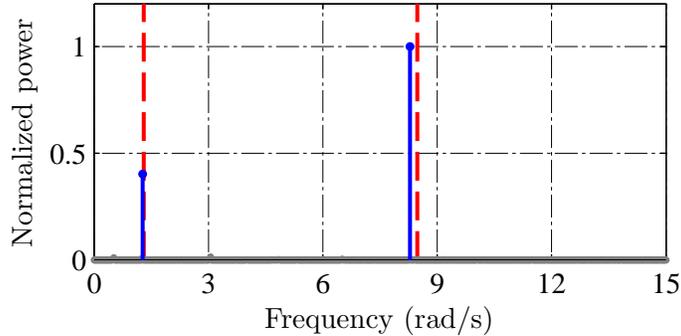


Figure 6.4: Compressed sensing spectrum computed from canonical dataset. The compressed sensing computation very accurately identifies the expected frequencies. (The true frequencies are denoted by red, dashed lines.)

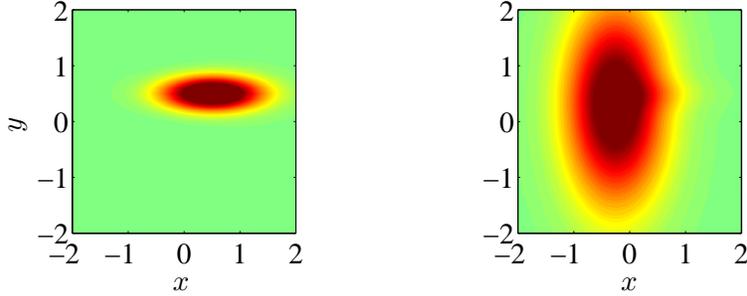


Figure 6.5: Compressed sensing modes for canonical dataset. Comparing to Figures 6.1 and 6.2, it is clear that compressed sensing has correctly combined the POD modes to recover the original oscillatory modes.

repeat the experiment, the aberrations are sometimes larger or smaller. However, we can decrease the likelihood of such errors by simply taking more samples (either by sampling faster or by taking a longer signal). Overall, Figures 6.4 and 6.5 show that our method is capable of identifying oscillatory structures in a spatial signal using sub-Nyquist-rate data, even in the presence of noise.

6.3.2 Flow past a two-dimensional cylinder

The low Reynolds number flow past a two-dimensional cylinder leads to sustained oscillations in the wake. The resulting wake structures are known collectively as a von Kármán vortex street. It is well known that a von Kármán vortex street is dominated by a single characteristic frequency. Thus while the flow may not be exactly sparse (in frequency space), it is an example of the type of flow that one might want to investigate experimentally using compressed sensing techniques. As such, it provides a valuable test of our method.

We conduct a cylinder flow experiment in a recirculating, free-surface water channel at a Reynolds number $Re = 413$.⁴ The cylinder used in the experiments is made of anodized aluminum and has diameter $D = 9.5$ mm and length $L = 260$ mm. To eliminate the effect of surface waves, we suspend the cylinder in the test section using an acrylic plate placed over the upper boundary of the water channel.

We generate a laser sheet using a Nd:YAG laser (Litron Nano L 50-50) and illuminate the cross-section at the mid-span of the cylinder. The sheet is imaged from below the water channel with a hybrid CCD/CMOS camera (LaVision, Imager sCMOS). The laser and the camera are synchronized with a programmable timing unit. We acquire 8000 image pairs with a delay of $8000 \mu s$ between exposures, at an overall sampling frequency of 20 Hz. For seeding, we use neutrally buoyant hollow ceramic spheres with an average diameter of $10 \mu m$.

PIV velocity fields are computed with a spatial cross-correlation algorithm using LaVision DaVis 8.1.2 software. The fields are processed using four passes with 50 % overlap: one pass with a 128×128 pixel interrogation window, one pass with a 64×64 pixel window, and two passes with a 32×32 pixel window. This results in velocity fields with 2160×1280 pixel resolution, with a cylinder diameter of approximately 128 pixels.

⁴Experimental data were acquired and processed by Jessica Shang (Princeton University).

For $Re = 413$, a sampling rate of 20 Hz easily resolves the wake shedding frequency. Thus we can use the time-resolved PIV data to compute DMD modes and eigenvalues. These provide a basis of comparison for our method, as they are in effect the true oscillatory modes and frequencies that we are trying to approximate using compressed sensing. The resulting DMD spectrum is shown in Figure 6.6. We observe that there is a dominant frequency at $f_{\text{wake}} = 0.889$. There are also harmonic peaks in the spectrum at approximately $2f_{\text{wake}}$ and $3f_{\text{wake}}$. We note that the peaks in the spectrum are somewhat broad, and that the harmonic peaks are significantly lower than the dominant one. Thus while we can identify three spectral peaks, one could argue that the flow is in fact dominated by a single frequency. The DMD modes corresponding to the wake frequency show strong coherence (Figure 6.7 (a)), as do those corresponding to $3f_{\text{wake}}$ (Figure 6.7 (c)). The modes corresponding to

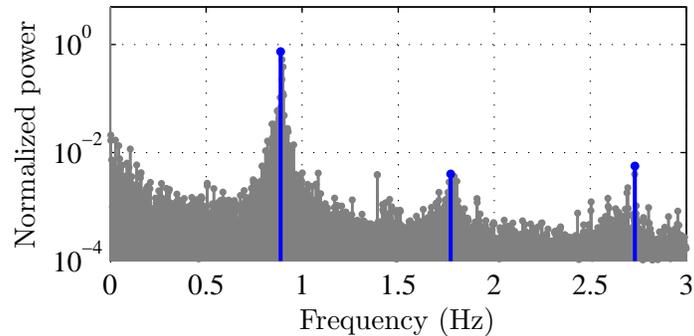


Figure 6.6: DMD spectrum computed from the flow past a two-dimensional cylinder using time-resolved PIV data. Peaks corresponding to modes shown in Figure 6.7 are highlighted in blue. There is a dominant spectral peak at $f = 0.889$ Hz, corresponding to the wake shedding frequency. Superharmonics of this frequency also appear in the spectrum, but with much lower power.

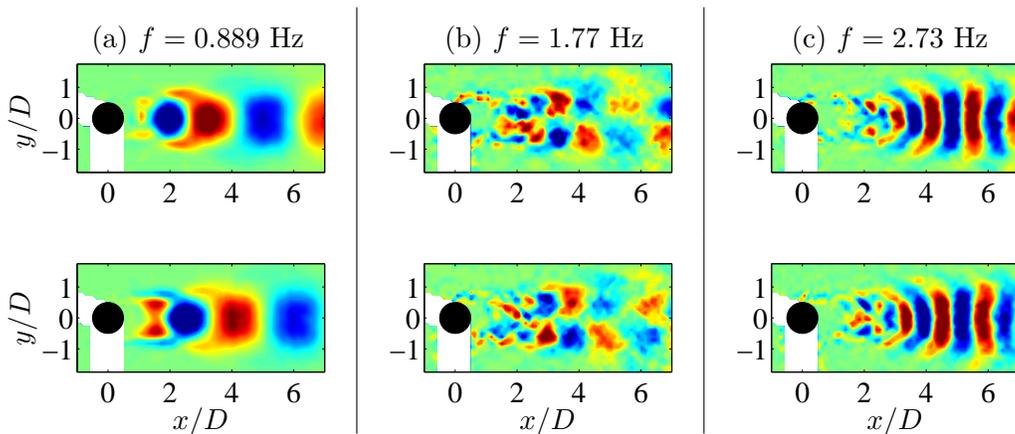


Figure 6.7: DMD modes computed from the flow past a two-dimensional cylinder, illustrated using contours of vorticity. The modes resemble those shown in Figures 2.9 and 5.8. This is expected, as all are computed from PIV data of bluff body wakes.

$2f_{\text{wake}}$ show features reminiscent of the top-bottom anti-symmetric modes seen in Figures 2.9 and 5.8, but here the structures are less coherent.

Figure 6.8 shows the energy distribution among POD modes computed from the time-resolved PIV data. We see that the flow is dominated by a single pair of POD modes. There is a sharp drop-off in energy content thereafter, with 12 modes required to capture 75 % of the energy contained in the dataset. We choose these first 12 modes as our low-dimensional basis for compressed sensing.⁵ From Figure 6.9 (a) and (b), we see that the dominant POD modes resemble the DMD modes corresponding to the wake shedding frequency (Figure 6.7 (a)). The remainder of the first six modes contain coherent structures, but do not resemble DMD modes. However, if we consider even lower energy modes, we do find some that resemble higher-frequency DMD modes; these POD modes are shown in Figure 6.10.

For the compressed sensing computation, we downsample the time-resolved PIV data, rather than acquiring a new dataset using a trigger. We sample the data using the minimum/maximum separation strategy, choosing $s_{\text{min}} = 50$ and $s_{\text{max}} = 70$, in comparison to $s_{\text{Nyq}} = 10$. This results in 33 total samples, out of the original $n = 2000$. Due to the similarity of the first POD mode and the dominant DMD mode, we expect that a time history the first POD coefficient will contain oscillations at the wake shedding frequency. Figure 6.11 shows the sample points overlaid on this history. We see that again, the sample points are so infrequent that traditional methods would not be able to reconstruct the original signal.

For this larger computation, we perform compressed sensing using OMP rather than ℓ_1 minimization. We compute the first ten DFT modes using this greedy approach; the resulting spectrum is shown in Figure 6.12. Once again, a dominant peak is identified, here corresponding to the wake shedding frequency. The error in this frequency is again less than 2.5 %. Unfortunately, the harmonic peaks observed in the DMD spectrum (Figure 6.6) do not appear here. This is the case even as we vary the sampling rate and the total number of samples. However, if we look again at Figure 6.6, we see that this may be caused by the broad peak around

⁵In practice, we would compute the POD modes using only the sampled data, and not the time-resolved data. However, for this flow we do not expect the POD basis to change much if computed from the sampled data, due to the strong attraction of the dynamics onto the low-dimensional POD subspace.

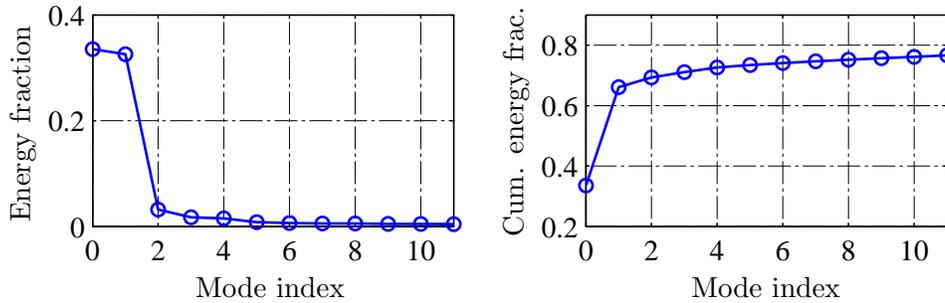


Figure 6.8: POD energy content for the flow past a two-dimensional cylinder. The first two modes dominate, followed by a slow roll-off in energy content. 12 modes are required to capture 75 % of the energy in the dataset. (Left) Energy fraction per mode; (right) cumulative energy fraction.

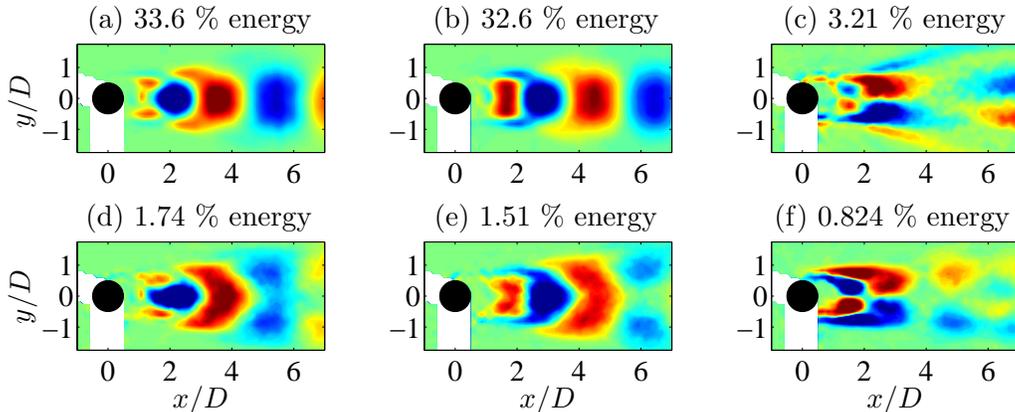


Figure 6.9: First six POD modes computed from the flow past a two-dimensional cylinder, illustrated using contours of vorticity. The dominant two modes ((a) and (b)) resemble the DMD modes corresponding to the wake shedding frequency (Figure 6.7 (a)). The next four modes contain coherent structures, but do not resemble DMD modes.

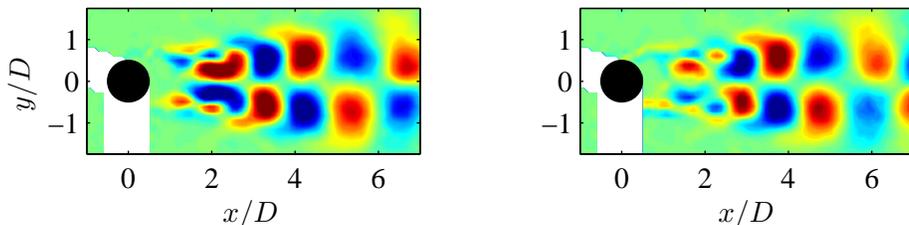


Figure 6.10: DMD-like POD modes computed from the flow past a two-dimensional cylinder, illustrated using contours of vorticity. These two modes resemble the higher-frequency DMD modes shown in Figure 6.7 (b). However, they contain very little energy. (Left) Mode 8, 0.646 % energy; (right) Mode 9, 0.579 % energy.

$f_{\text{wake}} = 0.889$ Hz. OMP iteratively finds basis vectors that contribute significantly to the dataset. While the harmonic peaks may stand out in the spectrum, the relatively low height of these peaks makes it difficult to identify them compared to the frequencies in the broad peak around f_{wake} . As such, it is not a failure of our method that these peaks are not identified; our method correctly characterizes the system as having only one dominant frequency.

The dominant OMP mode is depicted in Figure 6.13. We see that OMP correctly pairs the dominant POD modes with the wake shedding frequency, as expected based on DMD analysis. (Recall that the dominant POD modes closely resemble the dominant DMD modes.) Though the OMP modes do not exactly match the DMD modes (Figure 6.7), they capture the main coherent structures. We note that in theory, one could compute the POD modes using non-time-resolved data and then independently measure the dominant flow frequency using a hot-wire (which is much faster than PIV and can resolve the wake shedding frequency). One could then pair these together to arrive at the same conclusions as we get using OMP. However, the compressed sensing/OMP approach identifies the oscillatory modes and corre-

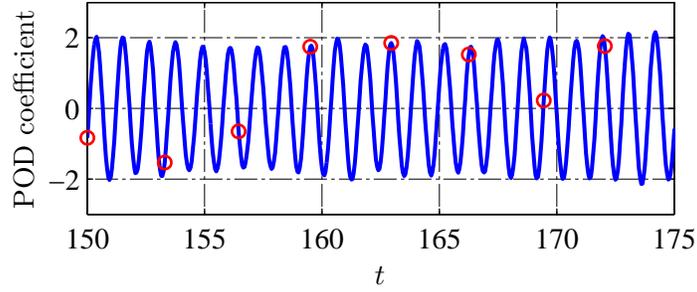


Figure 6.11: Random sampling for the flow past a two-dimensional cylinder. The first eight sample points are plotted over the time history of the first POD coefficient. We see that the sample points clearly do not resolve the fastest oscillations in this signal.

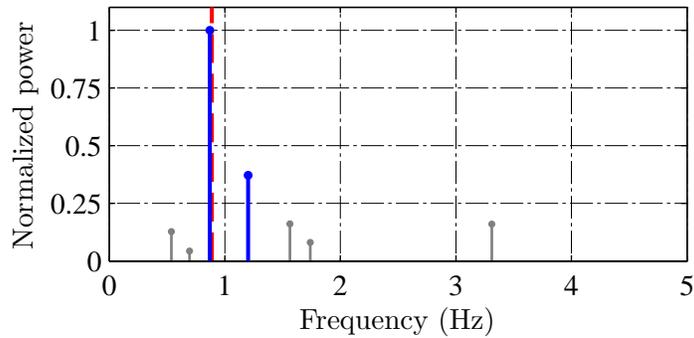


Figure 6.12: OMP spectrum for the flow past a two-dimensional cylinder. OMP correctly identifies the wake shedding frequency. (The true frequency is denoted by a red, dashed line.)

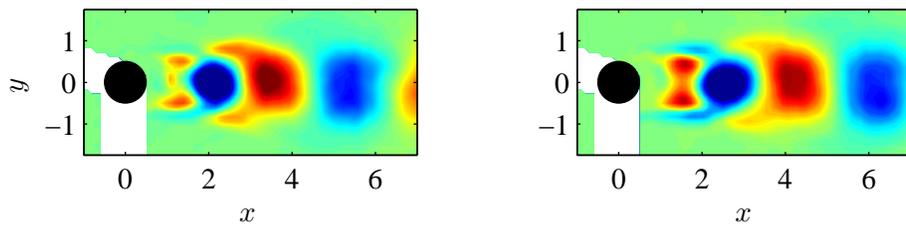


Figure 6.13: Dominant OMP mode for the flow past a two-dimensional cylinder, illustrated using contours of vorticity. Comparing to Figures 6.7, we see that OMP identifies the general structure of the true oscillatory mode.

sponding frequency directly from the data and does not require a priori knowledge of the flow dynamics (aside from an intuition that the signal is compressible). As such, it is generalizable to more complex flows, where it may not be obvious how to pair the dominant POD modes with characteristic flow frequencies.

6.4 Conclusions

We have demonstrated a method for computing oscillatory spatial modes from sub-Nyquist-rate data. This method combines concepts from compressed sensing, formulated for vector-valued signals, and DMD. The result is a method that is similar to the DFT, as the modes each correspond to a particular oscillatory frequency, but no temporal growth or decay. An interesting future direction would be to relax this assumption, either by choosing a more general basis Ψ or by adaptively searching for the correct growth/decay rates. Doing so would yield a method that could truly be called “compressed DMD.”

We demonstrated the capabilities of our method using both numerical and experimental data. The compressed sensing computations were done using ℓ_1 minimization and greedy algorithms (specifically OMP), respectively. In both cases, the correct frequencies and spatial modes were identified. This verifies not only the compressed sensing approach, but also the random sampling strategies proposed in Section 6.2.5.

Chapter 7

Conclusions and future directions

This thesis details a number of scientific contributions related to dynamic mode decomposition (DMD). Some of these are direct contributions to the understanding of DMD, whereas others make use of DMD as a means to other ends. In this chapter, we summarize the main findings and describe future directions in which this work can be extended.

7.1 Summary

There is presently a lack of clarity regarding the differences between DMD and Koopman spectral analysis, as evidenced by the fact that the terms “DMD mode” and “Koopman mode” are often used interchangeably in the fluids community. A key contribution of this thesis is the introduction of a precise definition of DMD (see Chapter 2) based solely on linear algebra. From this definition, we were able to develop a theoretical framework that encompasses a more general class of datasets than the existing theory. In particular, we used this framework to understand the potential benefits and pitfalls of applying DMD to rank-deficient datasets. We also developed variants of the standard DMD algorithm that extend it to non-sequential time-series. Through numerical examples, we demonstrated the effectiveness of these algorithms in increasing computational efficiency and reducing the effects of noise, respectively. While extensions of the DMD algorithm have been discussed among DMD practitioners for some time, such methods have generally been implemented in an ad hoc manner. Ours is the first attempt to build a general, theoretical framework from which such extensions arise naturally.

We also explored a variety of ways in which DMD can be applied. In Chapter 3, we made use of DMD as a snapshot-based method for identifying eigenvectors. This allows for efficient estimation of the slowly decaying eigenvectors that dominate the impulse responses of linear systems. Using those eigenvectors, along with their corresponding eigenvalues, we can model the long-term behavior of impulse responses analytically, avoiding the need for long impulse response simulations. We took advantage of this in developing a variant of balanced proper orthogonal decomposition (BPOD) that offers both increased accuracy and decreased computational cost.

Chapter 4 explores the use of DMD as a way to investigate the physics of oscillatory fluid flows. Two examples were discussed: a model for separated airfoil flows and shock-turbulent boundary layer interactions (STBLIs). In the former, we ran multiple simulations of a separated flow, varying the frequency at which we applied zero-net-mass-flux forcing. We analyzed each case using proper orthogonal decomposition (POD) and DMD, looking for trends in the modal characteristics that correlated with control effectiveness. We found that a decrease in the height of the mean separation bubble was correlated with a more even distribution of energy among the POD modes. We also saw that the most effective control was achieved when the flow was dominated by lock-on between the wake and separation bubble. In the STBLI example, the goal was to identify coherent structures that might elucidate the cause of low-frequency oscillations in the shock. We found that the DMD mode corresponding to the shock motion frequency resembles previously identified structures, suggesting a linear instability may lead to the observed oscillations. Both examples leveraged DMD as a tool to process large datasets and identify structures of physical interest.

Though DMD is a powerful tool, it is limited by the fact that it requires data samples that satisfy the Nyquist-Shannon sampling criterion. While not problematic for numerical simulations, in experiments it is often impossible to acquire such data. For instance, DMD is typically applied to snapshots of fluid velocity fields, which are typically captured using particle image velocimetry (PIV). Standard PIV systems have capture rates on the order of 15 Hz, which can easily be exceeded in flows of interest. (Faster PIV systems exist, but are expensive and thus relatively rare.)

In Chapter 5 we proposed a method that overcomes this limitation using dynamic estimation. We made use of fast, localized velocity measurements and slowly acquired PIV data to estimate the time evolution of a bluff-body wake. We found that by using a dynamic estimation approach, we outperformed stochastic estimation techniques, which are commonly used in the fluids community. Not only are the dynamic estimation results more accurate in the mean, but they are also more robust to noise. We applied DMD to the time-resolved velocity field estimates, finding that DMD based on the dynamically estimated snapshots more accurately identifies oscillatory structures in the flow. This approach allowed for DMD analysis of a fluid flow for which the natural frequencies (around 90 Hz) far exceed those accessible with typical PIV sampling rates.

We took a different approach in Chapter 6, using compressed sensing concepts to overcome sampling rate limitations. Both ℓ_1 minimization and greedy algorithms (specifically orthogonal matching pursuit) were implemented. To make these algorithms tractable for large datasets, we first projected the data onto POD modes. We also proposed strategies for random sampling that take into account measurement limitations. The overall method was applied to both a numerical and an experimental dataset (the flow past a two-dimensional cylinder). In both cases, the correct frequencies and oscillatory spatial modes were identified, using only sub-Nyquist-rate data and in the presence of noise.

7.2 Suggestions for future work

While the work presented in this thesis has answered a number of questions about DMD, it has also led to a number of new ones. For instance, while the theory discussed in Chapter 2 is independent of Koopman operator theory, it still allows for the interpretation of DMD modes as an approximation of Koopman modes (when they are computed from linearly independent data). There are a number of questions related to this approximation that merit further study. These include the following:

- **How good is the DMD approximation of Koopman modes?** Rowley et al. [138] suggest that if we can identify modes and eigenvalues that satisfy (2.21), then these modes and eigenvalues are indistinguishable from the true Koopman modes and eigenvalues, given the available data. Suppose we could compute the true Koopman modes and eigenvalues. How would the DMD approximation compare? Is there a way to measure the approximation error without knowing the true Koopman decomposition, perhaps using properties of the underlying dynamical system? (We note that Bagheri [14] studies the relationship between DMD and Koopman eigenvalues for a two-dimensional cylinder flow; it remains to be seen if one could generalize those findings.)
- **Are there canonical examples for which DMD and Koopman spectral analysis differ greatly?** Given that there is currently no guarantee on the accuracy of DMD as an approximation to Koopman spectral analysis, there may exist examples for which the two methods produce drastically different results. If such examples exist, it would be important to try and identify the causes of the differences, for instance using properties of the data or the underlying dynamics. Once these causes are identified, one could create canonical examples that illustrate the potential pitfalls in approximating Koopman decompositions using DMD.

A number of future directions also arise from the applications of DMD discussed in Chapters 3–5:

- **Can the analytic tail method be extended to the eigensystem realization algorithm (ERA)?** Ma et al. (2011) [99] showed that reduced-order models computed using the ERA are theoretically equivalent to those computed using BPOD. As such, it seems natural that the analytic tail method discussed in Chapter 3 should also extend to the ERA. Preliminary work suggests that this is a nontrivial exercise, due to the fact that the ERA is an adjoint-free method.
- **Can DMD analysis of the canonical separated flow be used to write down a coupled-oscillator model of the dynamics?** The canonical separated flow is characterized by a number of distinct frequencies, which interact and synchronize to varying degrees depending on the flow configuration. As such, it seems natural to model its dynamics using a coupled-oscillator model, perhaps with individual oscillators for the wake, separation bubble, shear layer, and actuator. One could use DMD analysis, similar to that described in Section 4.1, to posit how these oscillators should be coupled.

- **Do DMD modes outperform POD modes for Galerkin projection of the Navier–Stokes equations?** DMD modes are favored for their ability to capture nonlinear dynamics. As such, one might think that they would be a good choice of basis for Galerkin projection. In flows where the DMD modes differ from the POD modes, it would be informative to investigate the relative performance of reduced-order models computed using one set of modes versus the other.

Appendix A

Background on POD

Proper orthogonal decomposition (POD) is a data analysis method that identifies the dominant structures in a dataset [78, 96, 155]. More precisely, suppose we wish to project the dataset $\{\boldsymbol{\xi}_k\}_{k=0}^m$ onto an r -dimensional subspace. Let \mathbb{P}_r be the corresponding projection operator. Then the first r POD modes form the orthogonal basis that minimizes the sum-squared error

$$\sum_{k=0}^m \|\boldsymbol{\xi}_k - \mathbb{P}_r \boldsymbol{\xi}_k\|^2.$$

The image of each projection is spanned by the POD modes, with $\text{Image}(\mathbb{P}_k) \subset \text{Image}(\mathbb{P}_{k+1})$ for all k . As such, POD modes are naturally ordered, with a smaller mode index indicating a larger contribution to the accuracy of the projection. We can also say that the POD modes are ordered in terms of their energy content (where we equate the L_2 inner product to energy).

When analyzing an incompressible fluid flow, we generally take the data elements to be mean-subtracted velocity fields at given instants in time. That is, we let $\boldsymbol{\xi}_k = \mathbf{u}'_k = \mathbf{u}'(t_k)$. These elements are commonly referred to as “snapshots.” Each velocity field is discretized and reshaped into a one-dimensional vector, and then stacked in a data matrix

$$\mathbf{X} = \begin{bmatrix} | & | & \cdots & | \\ \mathbf{u}'_0 & \mathbf{u}'_1 & \cdots & \mathbf{u}'_m \\ | & | & & | \end{bmatrix}. \quad (\text{A.1})$$

The POD modes can then be computed efficiently using the “method of snapshots” [155], as described in Section 2.3.1. For convenience, we summarize the method here, including a matrix of inner product weights for generality.

The first step in the method of snapshots is to solve the $(m+1) \times (m+1)$ eigenvalue problem

$$\mathbf{X}^T \mathbf{M} \mathbf{X} \mathbf{W} = \mathbf{W} \boldsymbol{\Sigma}^2, \quad (\text{A.2})$$

where \mathbf{M} is the matrix of inner product weights. This matrix typically contains grid weights, for instance the scaled identity matrix $\mathbf{I} dx dy dz$. Including \mathbf{M} allows us

to interpret the vector norm as the integrated kinetic energy:

$$\|\mathbf{u}'_k\|^2 = (\mathbf{u}'_k)^T \mathbf{M} \mathbf{u}'_k = \iiint \left(u'(x, y, t_k)^2 + v'(x, y, t_k)^2 + w'(x, y, t_k)^2 \right) dx dy dz.$$

Having solved (A.2) for \mathbf{W} and $\mathbf{\Sigma}$, the matrix of POD modes is computed via

$$\mathbf{\Phi} = \mathbf{X} \mathbf{W} \mathbf{\Sigma}^{-1}, \quad (\text{A.3})$$

where the POD mode ϕ_j is given by the $j + 1$ th column of $\mathbf{\Phi}$. (In this thesis, we start our indexing from zero, and as such, the “first” mode corresponds to $j = 0$, the “second” to $j = 1$, and so on.)

The modes form an orthonormal set, satisfying the identity

$$\phi_j^T \mathbf{M} \phi_i = \delta_{ij}, \quad (\text{A.4})$$

where δ_{ij} is the Kronecker delta function. As such, the projection of a snapshot \mathbf{u}'_k onto the first r POD modes is given by

$$\mathbb{P}_r \mathbf{u}'_k = \mathbf{\Phi}_r \mathbf{a}_k, \quad (\text{A.5})$$

where $\mathbf{\Phi}_r$ contains only the first r columns of $\mathbf{\Phi}$, and

$$\mathbf{a}_k = \mathbf{\Phi}_r^T \mathbf{M} \mathbf{u}'_k. \quad (\text{A.6})$$

We refer to \mathbf{a}_k as the vector of POD coefficients corresponding to the snapshot \mathbf{u}'_k . For a spatially discretized velocity field, the dimension of a POD mode ϕ_j is the number of spatial dimensions times the number of grid points. In contrast, \mathbf{a}_k has only dimension r .

We observe that due to the orthogonality of the POD modes (see (A.4)), the energy in any POD approximation of a velocity field is simply given by

$$\|\mathbb{P}_r \mathbf{u}'_k\|^2 = \mathbf{a}_k^T \mathbf{\Phi}_r^T \mathbf{M} \mathbf{\Phi}_r \mathbf{a}_k = \mathbf{a}_k^T \mathbf{a}_k = \|\mathbf{a}_k\|_2^2. \quad (\text{A.7})$$

The kinetic energy captured by the projection itself can be computed as

$$\sum_{k=0}^m \|\mathbb{P}_r \boldsymbol{\xi}_k\|^2 = \sum_{j=0}^{r-1} \sigma_j, \quad (\text{A.8})$$

where the values σ_j are the singular values lying on the diagonal of $\mathbf{\Sigma}$. If all of the singular values in $\mathbf{\Sigma}$ are included in the right-hand sum, then the above equation yields the maximum possible energy that can be captured by a POD projection.

Bibliography

- [1] R. J. Adrian. On the role of conditional averages in turbulence theory. In *Proceedings of the 5th Biennial Symposium on Turbulence in Liquids*, pages 322–332. AFOSR, October 1977.
- [2] R. J. Adrian. Stochastic estimation of conditional structure — a review. *Appl. Sci. Res.*, 53(3–4):291–303, December 1994.
- [3] R. J. Adrian and P. Moin. Stochastic estimation of organized turbulent structure: Homogeneous shear-flow. *J. Fluid Mech.*, 190:531–559, May 1988.
- [4] S. Ahuja and C. W. Rowley. Feedback control of unstable steady states of flow past a flat plate using reduced-order estimators. *J. Fluid Mech.*, 645:447–478, February 2010.
- [5] J. C. Akers and D. S. Bernstein. Time-domain identification using AR-MARKOV/Toeplitz models. In *Proceedings of the American Control Conference*, pages 191–195, 1997.
- [6] E. Åkervik, L. Brandt, D. S. Henningson, J. Hoeffner, O. Marxen, and P. Schlatter. Steady solutions of the Navier–Stokes equations by selective frequency damping. *Phys. Fluids*, 18(6), June 2006.
- [7] M. Amitay, D. R. Smith, V. Kibens, D. E. Parekh, and A. Glezer. Aerodynamic flow control over an unconventional airfoil using synthetic jet actuators. *AIAA J.*, 39(3):361–370, March 2001.
- [8] J. D. Anderson. *Hypersonic and high-temperature gas dynamics*. AIAA Education Series, 2nd edition, 2006.
- [9] A. C. Antoulas. *Approximation of Large-Scale Dynamical Systems*. SIAM, 2005.
- [10] E. Aram, R. Mittal, J. Griffin, and L. Cattafesta. Towards effective ZNMF jet based control of a canonical separated flow. AIAA Paper 2010-4705, 5th Flow Control Conference, June 2010.
- [11] E. B. Arik and J. Carr. Digital particle image velocimetry system for real-time wind tunnel measurements. In *International Congress on Instrumentation in Aerospace Simulation Facilities*, pages 267–277, 1997.
- [12] N. Aubry, P. Holmes, J. L. Lumley, and E. Stone. The dynamics of coherent structures in the wall region of a turbulent boundary layer. *J. Fluid Mech.*, 192:115–173, July 1988.

- [13] S. Bagheri. Computational hydrodynamic stability and flow control based on spectral analysis of linear operators. *Arch. Comput. Method E.*, 19(3):341–379, September 2012.
- [14] S. Bagheri. Koopman-mode decomposition of the cylinder wake. *J. Fluid Mech.*, 726:596–623, July 2013.
- [15] S. Bagheri, L. Brandt, and D. S. Henningson. Input-output analysis, model reduction and control of the flat-plate boundary layer. *J. Fluid Mech.*, 620:263–298, February 2009.
- [16] S. Bagheri, D. S. Henningson, J. Hoeffner, and P. J. Schmid. Input-output analysis and control design applied to a linear model of spatially developing flows. *Appl. Mech. Rev.*, 62(2), March 2009.
- [17] A. Bar-Sever. Separation control on an airfoil by periodic forcing. *AIAA J.*, 27(6):820–821, June 1989.
- [18] R. G. Baraniuk. Compressive sensing. *IEEE Signal Proc. Mag.*, 24(4):118–124, July 2007.
- [19] R. G. Baraniuk. More is less: Signal processing and the data deluge. *Science*, 331(6018):717–719, February 2011.
- [20] A. Barbagallo, D. Sipp, and P. J. Schmid. Closed-loop control of an open cavity flow using reduced-order models. *J. Fluid Mech.*, 641:1–50, December 2009.
- [21] B. A. Belson, J. H. Tu, and C. W. Rowley. A parallelized model reduction library. *ACM T. Math. Software*, (submitted).
- [22] S. J. Beresh, N. T. Clemens, and D. S. Dolling. Relationship between upstream turbulent boundary-layer velocity fluctuations and separation shock unsteadiness. *AIAA J.*, 40(12):2412–2422, December 2002.
- [23] M. B. Blumenthal. Predictability of a coupled ocean-atmosphere model. *J. Climate*, 4(8):766–784, August 1991.
- [24] J. P. Bonnet, D. R. Cole, J. Delville, M. N. Glauser, and L. S. Ukeiley. Stochastic estimation and proper orthogonal decomposition: Complementary techniques for identifying structure. *Exp. Fluids*, 17(5):307–314, September 1994.
- [25] E. J. Candes and T. Tao. Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE T. Inform. Theory*, 52(12):5406–5425, December 2006.
- [26] E. J. Candes and M. B. Wakin. An introduction to compressive sampling. *IEEE Signal Proc. Mag.*, 25(2):21–30, March 2008.
- [27] E. J. Candes, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE T. Inform. Theory*, 52(2):489–509, February 2006.

- [28] F. J. Chen and G. B. Beeler. Virtual shaping of a two-dimensional NACA 0015 airfoil using synthetic jet actuator. AIAA Paper 2002-3273, 1st Flow Control Conference, June 2002.
- [29] J. Chen and X. Huo. Theoretical results on sparse representations of multiple-measurement vectors. *IEEE T. Signal Proces.*, 54(12):4634–4643, December 2006.
- [30] K. K. Chen and C. W. Rowley. H_2 optimal actuator and sensor placement in the linearised complex Ginzburg-Landau system. *J. Fluid Mech.*, 681:241–260, August 2011.
- [31] K. K. Chen, J. H. Tu, and C. W. Rowley. Variants of dynamic mode decomposition: Boundary condition, Koopman, and Fourier analyses. *J. Nonlinear Sci.*, 22(6):887–915, December 2012.
- [32] J. M. Chomaz. Global instabilities in spatially developing flows: Non-normality and nonlinearity. *Annu. Rev. Fluid Mech.*, 37:357–392, 2005.
- [33] N. T. Clemens and V. Narayanaswamy. Shock/turbulent boundary layer interactions: Review of recent work on sources of unsteadiness. AIAA Paper 2009-3710, 39th AIAA Fluid Dynamics Conference, June 2009.
- [34] K. Cohen, S. Siegel, and T. McLaughlin. A heuristic approach to effective sensor placement for modeling of a cylinder wake. *Comput. Fluids*, 35(1): 103–120, January 2006.
- [35] D. R. Cole, M. N. Glauser, and Y. G. Guezennec. An application of the stochastic estimation to the jet mixing layer. *Phys. Fluids A-Fluid*, 4(1):192–194, January 1992.
- [36] T. Colonius and K. Taira. A fast immersed boundary method using a nullspace approach and multi-domain far-field boundary conditions. *Comput. Method Appl. M.*, 197(25–28):2131–2146, 2008.
- [37] C. Cossu and J. M. Chomaz. Global measures of local convective instabilities. *Phys. Rev. Lett.*, 78(23):4387–4390, June 1997.
- [38] S. F. Cotter, B. D. Rao, K. Engan, and K. Kreutz-Delgado. Sparse solutions to linear inverse problems with multiple measurement vectors. *IEEE T. Signal Proces.*, 53(7):2477–2488, July 2005.
- [39] L. Cullen, B. Nishri, D. Greenblatt, and I. Wygnanski. The effect of curvature on boundary layers subjected to strong adverse pressure gradients. AIAA Paper 2002-0578, 40th AIAA Aerospace Sciences Meeting and Exhibition, January 2002.
- [40] A. Darabi and I. Wygnanski. On the transient process of flow reattachment by external excitation. AIAA Paper 2002-3163, 1st Flow Control Conference, June 2002.

- [41] A. E. Deane, I. G. Kevrekidis, G. E. Karniadakis, and S. A. Orszag. Low-dimensional models for complex-geometry flows: Application to grooved channels and circular-cylinders. *Phys. Fluids A-Fluid*, 3(10):2337–2354, October 1991.
- [42] J. Détery and J. G. Marvin. *Shock-Wave Boundary Layer Interactions*. 280. AGARDograph, 1986.
- [43] G. Dergham, D. Sipp, and J.-C. Robinet. Accurate low dimensional models for deterministic fluid systems driven by uncertain forcing. *Phys. Fluids*, 23(9), September 2011.
- [44] D. S. Dolling. Fifty years of shock-wave/boundary-layer interaction research: What next? *AIAA J.*, 39(8):1517–1531, August 2001.
- [45] D. L. Donoho. Compressed sensing. *IEEE T. Inform. Theory*, 52(4):1289–1306, April 2006.
- [46] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk. Single-pixel imaging via compressive sampling. *IEEE Signal Proc. Mag.*, 25(2):83–91, March 2008.
- [47] D. Duke, D. Honnery, and J. Soria. Experimental investigation of nonlinear instabilities in annular liquid sheets. *J. Fluid Mech.*, 691:594–604, January 2012.
- [48] D. Duke, J. Soria, and D. Honnery. An error analysis of the dynamic mode decomposition. *Exp. Fluids*, 52(2):529–542, February 2012.
- [49] G. E. Dullerud and F. Paganini. *A Course in Robust Control Theory: A Convex Approach*, volume 36 of *Texts in Applied Mathematics*. Springer-Verlag, 2000.
- [50] P. Dupont, C. Haddad, and J. F. Debieve. Space and time organization in a shock-induced separated boundary layer. *J. Fluid Mech.*, 559:255–277, July 2006.
- [51] V. Durgesh and J. W. Naughton. Multi-time-delay LSE-POD complementary approach applied to unsteady high-Reynolds-number near wake flow. *Exp. Fluids*, 49(3):571–583, September 2010.
- [52] Y. C. Eldar and M. Mishali. Robust recovery of signals from a structured union of subspaces. *IEEE T. Inform. Theory*, 55(11):5302–5316, November 2009.
- [53] M. E. Erengil and D. S. Dolling. Unsteady wave structure near separation in a Mach-5 compression ramp interaction. *AIAA J.*, 29(5):728–735, May 1991.
- [54] R. Everson and L. Sirovich. Karhunen–Loève procedure for gappy data. *J. Opt. Soc. Am. A*, 12(8):1657–1664, August 1995.

- [55] D. Ewing and J. H. Citriniti. Examination of a LSE/POD complementary technique using single and multi-time information in the axisymmetric shear layer. In *Proceedings of the IUTAM Symposium on Simulation and Identification of Organized Structures in Flows*, pages 375–384, 1999.
- [56] R. Funk, D. Parekh, T. Crittenden, and A. Glezer. Transient separation control using pulse combustion actuation. AIAA Paper 2002-3166, 1st Flow Control Conference, June 2002.
- [57] U. Gamper, P. Boesiger, and S. Kozerke. Compressed sensing in dynamic MRI. *Magn. Reson. Med.*, 59(2):365–373, February 2008.
- [58] B. Ganapathisubramani, N. T. Clemens, and D. S. Dolling. Effects of upstream boundary layer on the unsteadiness of shock-induced separation. *J. Fluid Mech.*, 585:369–394, August 2007.
- [59] B. Ganapathisubramani, N. T. Clemens, and D. S. Dolling. Low-frequency dynamics of shock-induced separation in a compression ramp interaction. *J. Fluid Mech.*, 636:397–425, October 2009.
- [60] J. Gantz and D. Reinsel. The digital universe decade — Are you ready? IDC white paper, May 2010.
- [61] J. Gerhard, M. Pastoor, R. King, B. R. Noack, A. Dillmann, M. Morzyński, and G. Tadmor. Model-based control of vortex shedding using low-dimensional Galerkin models. AIAA Paper 2003-4262, 33rd AIAA Fluid Dynamics Conference and Exhibit, June 2003.
- [62] J. L. Gilarranz and O. K. Rediniotis. Compact, high-power synthetic jet actuators for flow separation control. AIAA Paper 2001-0737, 39th AIAA Aerospace Sciences Meeting and Exhibit, January 2001.
- [63] M. N. Glauser, H. Higuchi, J. Ausseur, and J. Pinier. Feedback control of separated flows. AIAA Paper 2004-2521, 2nd Flow Control Conference, June 2004.
- [64] S. V. Gordeyev and F. O. Thomas. A temporal proper decomposition (TPOD) for closed-loop flow control. *Exp. Fluids*, 54(3), March 2013.
- [65] P. J. Goulart, A. Wynn, and D. Pearson. Optimal mode decomposition for high dimensional systems. In *Proceedings of the 51st IEEE Conference on Decision and Control*, December 2012.
- [66] M. Grant and S. P. Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. P. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, volume 371 of *Lecture Notes in Control and Information Sciences*, pages 95–110. Springer-Verlag, 2008.
- [67] M. Grant and S. P. Boyd. CVX: Matlab software for disciplined convex programming, version 2.0 beta. <http://cvxr.com/cvx>, September 2012. URL <http://cvxr.com/cvx>.

- [68] D. Greenblatt and I. Wygnanski. Parameters affecting dynamic stall control by oscillatory excitation. AIAA Paper 1999-3121, AIAA Fluid Dynamics Conference, June 1999.
- [69] D. Greenblatt and I. Wygnanski. Parameters affecting dynamic stall control by oscillatory excitation. AIAA Paper 2002-3271, 1st Flow Control Conference, June 2002.
- [70] D Greenblatt and I Wygnanski. Effect of leading-edge curvature on airfoil separation control. *J. Aircraft*, 40(3):473–481, May 2003.
- [71] J. Griffin, T. Schultz, R. Holman, L. S. Ukeiley, and L. N. Cattafesta III. Application of multivariate outlier detection to fluid velocity measurements. *Exp. Fluids*, 49(1):305–317, July 2010.
- [72] J. Griffin, M. Oyarzun, L. N. Cattafesta III, J. H. Tu, C. W. Rowley, and R. Mittal. Control of a canonical separated flow. AIAA Paper 2013-2968, 43rd AIAA Fluid Dynamics Conference, June 2013.
- [73] M. Grilli, P. J. Schmid, S. Hickel, and N. A. Adams. Analysis of unsteady behaviour in shockwave turbulent boundary layer interaction. *J. Fluid Mech.*, 700:16–28, June 2012.
- [74] Y. G. Guezennec. Stochastic estimation of coherent structures in turbulent boundary-layers. *Phys. Fluids A-Fluid*, 1(6):1054–1060, June 1989.
- [75] K. Hasselmann. PIPs and POPs: the reduction of complex dynamical-systems using Principal Interaction and Oscillation Patterns. *J. Geophys. Res.-Atmos.*, 93(D9):11015–11021, September 1988.
- [76] Y. Y. He, A. W. Cary, and D. A. Peters. Parametric and dynamic modeling for synthetic jet control of a post-stall airfoil. AIAA Paper 2001-0733, 39th AIAA Aerospace Sciences Meeting and Exhibit, January 2001.
- [77] M. A. Herman and T. Strohmer. High-resolution radar via compressed sensing. *IEEE T. Signal Proces.*, 57(6):2275–2284, June 2009.
- [78] P. Holmes, J. L. Lumley, G. Berkooz, and C. W. Rowley. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge University Press, 2012.
- [79] H. Hotelling. Analysis of a complex of statistical variables into principal components. *J. Educ. Psychol.*, 24:417–441, September 1933.
- [80] H. Hotelling. Analysis of a complex of statistical variables into principal components. *J. Educ. Psychol.*, 24:498–520, October 1933.
- [81] M. Hubert and E. Vandervieren. An adjusted boxplot for skewed distributions. *Comput. Stat. Data An.*, 52(12):5186–5201, August 2008.
- [82] R. A. Humble, G. E. Elsinga, F. Scarano, and B. W. van Oudheusden. Three-dimensional instantaneous structure of a shock wave/turbulent boundary layer interaction. *J. Fluid Mech.*, 622:33–62, March 2009.

- [83] D. C. Hyland. Neural network architecture for online system identification and adaptively optimized control. In *Proceedings of the 30th IEEE Conference on Decision and Control*, December 1991.
- [84] M. Ilak and C. W. Rowley. Modeling of transitional channel flow using balanced proper orthogonal decomposition. *Phys. Fluids*, 20(3), March 2008.
- [85] J. N. Juang and R. S. Pappa. An eigensystem realization-algorithm for modal parameter-identification and model-reduction. *J. Guid. Control Dynam.*, 8(5): 620–627, 1985.
- [86] J. N. Juang, M. Phan, L. G. Horta, and R. W. Longman. Identification of observer/Kalman filter Markov parameters: Theory and experiments. Technical memorandum 104069, NASA, 1991.
- [87] H.-J. Kaltenbach, H. Fatica, R. Mittal, T. S. Lund, and P. Moin. Study of flow in a planar asymmetric diffuser using large-eddy simulation. *J. Fluid Mech.*, 390:151–185, July 1999.
- [88] R. King, K. Aleksić, G. Gelbert, N. Losse, R. Muminovic, A. Brunn, W. Nitsche, M. R. Bothien, J. P. Moeck, C. O. Paschereit, B. R. Noack, U. Rist, and M. Zengl. Model predictive flow control. AIAA Paper 2008-3975, 38th Fluid Dynamics Conference and Exhibit, June 2008.
- [89] B. O. Koopman. Hamiltonian systems and transformations in Hilbert space. *P. Natl. Acad. Sci. USA*, 17:315–318, 1931.
- [90] R. B. Kotapati, R. Mittal, O. Marxen, F. Ham, D. You, and L. N. Cattafesta III. Nonlinear dynamics and synthetic-jet-based control of a canonical separated flow. *J. Fluid Mech.*, 654:65–97, July 2010.
- [91] M. Legrand, J. Nogueira, and A. Lecuona. Flow temporal reconstruction from non-time-resolved data part I: Mathematic fundamentals. *Exp. Fluids*, 51(4): 1047–1055, October 2011.
- [92] M. Legrand, J. Nogueira, S. Tachibana, A. Lecuona, and S. Nauri. Flow temporal reconstruction from non time-resolved data part II: Practical implementation, methodology validation, and applications. *Exp. Fluids*, 51(4): 861–870, October 2011.
- [93] F. Li and N. Aubry. Feedback control of a flow past a cylinder via transverse motion. *Phys. Fluids*, 15(8):2163–2176, August 2003.
- [94] M. Loève. *Probability Theory II*. Springer-Verlag, 4th edition, 1978.
- [95] E. N. Lorenz. Empirical orthogonal functions and statistical weather prediction. Technical report, Massachusetts Institute of Technology, December 1956.
- [96] J. L. Lumley. The structure of inhomogeneous turbulence. In A.M. Yaglom and V.I. Tatarski, editors, *Atmospheric Turbulence and Wave Propagation*, pages 166–178. Nauka, Moscow, 1967.

- [97] J. L. Lumley. *Stochastic Tools in Turbulence*. Academic Press, New York, 1970.
- [98] M. Lustig, D. L. Donoho, and J. M. Pauly. Sparse MRI: the application of compressed sensing for rapid MR imaging. *Magn. Reson. Med.*, 58(6):1182–1195, December 2007.
- [99] Z. Ma, S. Ahuja, and C. W. Rowley. Reduced-order models for control of fluids using the eigensystem realization algorithm. *Theor. Comp. Fluid Dyn.*, 25(1–4):233–247, June 2011.
- [100] D. Malioutov, M. Çetin, and A. S. Willsky. A sparse signal reconstruction perspective for source localization with sensor arrays. *IEEE T. Signal Proces.*, 53(8):3010–3022, August 2005.
- [101] S. Margalit, D. Greenblatt, and I. Wygnanski. Active flow control of a delta wing at high active flow control of a delta wing at high incidence using segmented piezoelectric actuators. AIAA Paper 2002-3270, 1st Flow Control Conference, June 2002.
- [102] L. Massa, R. Kumar, and P. Ravindran. Dynamic mode decomposition analysis of detonation waves. *Phys. Fluids*, 24(6), June 2012.
- [103] I. Mezić. Spectral properties of dynamical systems, model reduction and decompositions. *Nonlin. Dynam.*, 41(1–3):309–325, August 2005.
- [104] I. Mezić. Analysis of fluid flows via spectral properties of the Koopman operator. *Annu. Rev. Fluid Mech.*, 45:357–378, 2013.
- [105] S. Miranda and D. Telionis. Flow control of a sharp edged airfoil. AIAA Paper 2001-0119, 39th AIAA Aerospace Sciences Meeting and Exhibit, January 2001.
- [106] R. Mittal, R. Kotapati, and L. N. Cattafesta III. Numerical study of resonant interactions and flow control in a canonical separated flow. AIAA Paper 2005-1261, 43rd AIAA Aerospace Sciences Meeting and Exhibit, January 2005.
- [107] R. Mittal, H. Dong, M. Bozkurttas, F. M. Najjar, A. Vargas, and A. Von Loebbecke. A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries. *J. Comput. Phys.*, 227(10):4825–4852, May 2008.
- [108] B. C. Moore. Principal component analysis in linear-systems: Controllability, observability, and model-reduction. *IEEE T. Automat. Contr.*, 26(1):17–32, 1981.
- [109] T. W. Muld, G. Efraimsson, and D. S. Henningson. Flow structures around a high-speed train extracted using proper orthogonal decomposition and dynamic mode decomposition. *Comput. Fluids*, 57:87–97, March 2012.
- [110] N. E. Murray and L. S. Ukeiley. Estimation of the flowfield from surface pressure measurements in an open cavity. *AIAA J.*, 41(5):969–972, May 2003.

- [111] N. E. Murray and L. S. Ukeiley. Modified quadratic stochastic estimation of resonating subsonic cavity flow. *J. Turbul.*, 8(53):1–23, 2007.
- [112] Nathan E. Murray and Lawrence S. Ukeiley. An application of gappy POD; for subsonic cavity flow PIV data. *Exp. Fluids*, 42(1):79–91, January 2007.
- [113] A. M. Naguib, C. E. Wark, and O. Juckenhöfel. Stochastic estimation and flow sources associated with surface pressure events in a turbulent boundary layer. *Phys. Fluids*, 13(9):2611–2626, September 2001.
- [114] D. Needell and J. A. Tropp. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Appl. Comput. Harmon. A.*, 26(3):301–321, May 2009.
- [115] B. R. Noack, K. Afanasiev, M. Morzyński, G. Tadmor, and F. Thiele. A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. *J. Fluid Mech.*, 497:335–363, December 2003.
- [116] B. R. Noack, G. Tadmor, and M. Morzyński. Actuation models and dissipative control in empirical Galerkin models of fluid flows. In *Proceedings of the American Control Conference*, pages 5722–5727, 2004.
- [117] H. Nyquist. Certain topics in telegraph transmission theory. *T. AIEE*, 47(2): 617–644, 1928.
- [118] L. G. Pack and A. Seifert. Dynamics of active separation control at high Reynolds numbers. AIAA Paper 2000-0409, 38th AIAA Aerospace Sciences Meeting and Exhibit, January 2000.
- [119] L. G. Pack, N. Schaeffler, and C. S. Yao. Active control of flow separation from the slat shoulder of a supercritical airfoil. AIAA Paper 2002-3156, 1st Flow Control Conference, June 2002.
- [120] A. Papoulis. *Probability, Random Variables and Stochastic Theory*. McGraw-Hill, 1984.
- [121] M. Pastoor, B. R. Noack, R. King, and G. Tadmor. Spatiotemporal waveform observers and feedback in shear layer control. AIAA Paper 2006-1402, 44th AIAA Aerospace Sciences Meeting and Exhibit, January 2006.
- [122] M. Pastoor, L. Henning, B. R. Noack, R. King, and G. Tadmor. Feedback shear layer control for bluff body drag reduction. *J. Fluid Mech.*, 608:161–196, August 2008.
- [123] K. Pearson. LIII. on lines and planes of closest fit to systems of points in space. *Philos. Mag.*, 2(7–12):559–572, July 1901.
- [124] C. Penland. Random forcing and forecasting using Principal Oscillation Pattern analysis. *Mon. Weather Rev.*, 117(10):2165–2185, October 1989.
- [125] C. Penland and T. Magorian. Prediction of Niño 3 sea-surface temperatures using linear inverse modeling. *J. Climate*, 6(6):1067–1076, June 1993.

- [126] T. Penzl. A cyclic low-rank Smith method for large sparse Lyapunov equations. *SIAM J. Sci. Comput.*, 21(4):1401–1418, April 2000.
- [127] M. Phan, L. G. Horta, J. N. Juang, and R. W. Longman. Linear-system identification via an asymptotically stable observer. *J. Optimiz. Theory App.*, 79(1):59–86, October 1993.
- [128] S. Piponnier, J. P. Dussauge, J. F. Debieve, and P. Dupont. A simple model for low-frequency unsteadiness in shock-induced separation. *J. Fluid Mech.*, 629:87–108, June 2009.
- [129] L. C. Potter, E. Ertin, J. T. Parker, and M. Çetin. Sparsity and compressed sensing in radar imaging. *P. IEEE*, 98(6):1006–1020, June 2010.
- [130] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes*. Cambridge University Press, 3rd edition, 2007.
- [131] S. Priebe. *Direct Numerical Simulation of Two Shock Wave/Turbulent Boundary Layer Interactions*. PhD thesis, Princeton University, 2012.
- [132] S. Priebe and M. P. Martín. Low-frequency unsteadiness in shock wave-turbulent boundary layer interaction. *J. Fluid Mech.*, 699:1–49, May 2012.
- [133] B. Protas. Linear feedback stabilization of laminar vortex shedding based on a point vortex model. *Phys. Fluids*, 16(12):4473–4488, December 2004.
- [134] R. Raju, E. Aram, R. Mittal, and L. N. Cattafesta III. Simple models of zero-net mass-flux jets for flow control simulations. *Int. J. Flow Contr.*, 1(3):179–197, 2009.
- [135] S. S. Ravindran. Active control of flow separation over an airfoil. Technical memorandum 209838, NASA, 1999.
- [136] J. Romberg. Imaging via compressive sampling. *IEEE Signal Proc. Mag.*, 25(2):14–20, March 2008.
- [137] C. W. Rowley. Model reduction for fluids, using balanced proper orthogonal decomposition. *Int. J. Bifurcat. Chaos*, 15(3):997–1013, March 2005.
- [138] C. W. Rowley, I. Mezic, S. Bagheri, P. Schlatter, and D. S. Henningson. Spectral analysis of nonlinear flows. *J. Fluid Mech.*, 641:115–127, December 2009.
- [139] Y. Saad. Variations on Arnoldi’s method for computing eigenvalues of large unsymmetric matrices. *Linear Algebra Appl.*, 34(December):269–295, 1980.
- [140] M. Schlegel, B. R. Noack, P. Jordan, A. Dillmann, E. Gröschel, W. Schröder, M. Wei, J. B. Freund, O. Lehmann, and G. Tadmor. On least-order flow representations for aerodynamics and aeroacoustics. *J. Fluid Mech.*, 697:367–398, April 2012.
- [141] P. J. Schmid. Dynamic mode decomposition of numerical and experimental data. *J. Fluid Mech.*, 656:5–28, August 2010.

- [142] P. J. Schmid. Application of the dynamic mode decomposition to experimental data. *Exp. Fluids*, 50(4):1123–1130, April 2011.
- [143] P. J. Schmid and J. Sesterhenn. Dynamic mode decomposition of numerical and experimental data. In *61st Annual Meeting of the APS Division of Fluid Dynamics*, San Antonio, Texas, November 2008. American Physical Society.
- [144] P. J. Schmid, L. Li, M. P. Juniper, and O. Pust. Applications of the dynamic mode decomposition. *Theor. Comp. Fluid Dyn.*, 25(1–4):249–259, June 2011.
- [145] P. J. Schmid, D. Violato, and F. Scarano. Decomposition of time-resolved tomographic PIV. *Exp. Fluids*, 52(6):1567–1579, June 2012.
- [146] A. Seena and H. J. Sung. Dynamic mode decomposition of turbulent cavity flows for self-sustained oscillations. *Int. J. Heat Fluid Fl.*, 32(6):1098–1110, December 2011.
- [147] A. Seifert and L. G. Pack. Oscillatory control of separation at high Reynolds numbers. *AIAA J.*, 37(9):1062–1071, September 1999.
- [148] A. Seifert, T. Bachar, D. Koss, M. Shephelovich, and I. Wygnanski. Oscillatory blowing: A tool to delay boundary-layer separation. *AIAA J.*, 31(11):2052–2060, November 1993.
- [149] A. Seifert, A. Darabi, and I. Wygnanski. Delay of airfoil stall by periodic excitation. *J. Aircraft*, 33(4):691–698, July 1996.
- [150] O. Semeraro, S. Bagheri, L. Brandt, and D. S. Henningson. Feedback control of three-dimensional optimal disturbances using reduced-order models. *J. Fluid Mech.*, 677:63–102, June 2011.
- [151] O. Semeraro, G. Bellani, and F. Lundell. Analysis of time-resolved PIV measurements of a confined turbulent jet using POD and Koopman modes. *Exp. Fluids*, 53(5):1203–1220, November 2012.
- [152] C. E. Shannon. Communication in the presence of noise. *P. IRE*, 37(1):10–21, 1949.
- [153] S. G. Siegel, J. Seidel, C. Fagley, D. M. Luchtenburg, K. Cohen, and T. McLaughlin. Low-dimensional modelling of a transient cylinder wake using double proper orthogonal decomposition. *J. Fluid Mech.*, 610:1–42, September 2008.
- [154] D. Simon. *Optimal State Estimation*. John Wiley and Sons, 2006.
- [155] L. Sirovich. Turbulence and the dynamics of coherent structures parts I–III. *Q. Appl. Math.*, 45(3):561–590, October 1987.
- [156] R. A. Smith. Matrix equation $XA + BX = C$. *SIAM J. Appl. Math.*, 16(1):198–201, 1968.
- [157] A. J. Smits and J.-P. Dussauge. *Turbulent Shear Layers in Supersonic Flow*. Springer-Verlag, 2nd edition, 2006.

- [158] K. Taira and T. Colonius. The immersed boundary method: a projection approach. *J. Comput. Phys.*, 225(2):2118–2137, August 2007.
- [159] D. S. Taubman and M. W. Marcellin. *JPEG 2000: Image Compression Fundamentals, Standards and Practice*. Kluwer, 2001.
- [160] J. A. Taylor and M. N. Glauser. Towards practical flow sensing and control via POD and LSE based low-dimensional tools. *J. Fluids Eng.*, 126(3):337–345, May 2004.
- [161] F. O. Thomas, C. M. Putnam, and H. C. Chu. On the mechanism of unsteady shock oscillation in shock-wave turbulent boundary-layer interactions. *Exp. Fluids*, 18(1–2):69–81, December 1994.
- [162] Y. Tian, L. N. Cattafesta III, and R. Mittal. Adaptive control of separated flow. AIAA Paper 2006-1401, 44th AIAA Aerospace Sciences Meeting and Exhibit, January 2006.
- [163] C. E. Tinney, F. Coiffet, J. Delville, A. M. Hall, P. Jordan, and M. N. Glauser. On spectral linear stochastic estimation. *Exp. Fluids*, 41(5):763–775, November 2006.
- [164] C. E. Tinney, L. S. Ukeiley, and M. N. Glauser. Low-dimensional characteristics of a transonic jet. Part 2. Estimate and far-field prediction. *J. Fluid Mech.*, 615:53–92, November 2008.
- [165] E. Touber and N. D. Sandham. Large-eddy simulation of low-frequency unsteadiness in a turbulent shock-induced separation bubble. *Theor. Comp. Fluid Dyn.*, 23(2):79–107, June 2009.
- [166] L. N. Trefethen and D. Bau III. *Numerical Linear Algebra*. SIAM, Philadelphia, 1997.
- [167] J. A. Tropp. Greed is good: Algorithmic results for sparse approximation. *IEEE T. Inform. Theory*, 50(10):2231–2242, October 2004.
- [168] J. A. Tropp. Algorithms for simultaneous sparse approximation. Part II: Convex relaxation. *Signal Process.*, 86(3):589–602, March 2006.
- [169] J. A. Tropp and A. C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE T. Inform. Theory*, 53(12):4655–4666, December 2007.
- [170] J. A. Tropp, A. C. Gilbert, and M. J. Strauss. Algorithms for simultaneous sparse approximation. Part I: Greedy pursuit. *Signal Process.*, 86(3):572–588, March 2006.
- [171] J. H. Tu and C. W. Rowley. An improved algorithm for balanced POD through an analytic treatment of impulse response tails. *J. Comput. Phys.*, 231(16):5317–5333, June 2012.

- [172] J. H. Tu, C. W. Rowley, E. Aram, and R. Mittal. Koopman spectral analysis of separated flow over a finite-thickness flat plate with elliptical leading edge. AIAA Paper 2011-38, 49th AIAA Aerospace Sciences Meeting and Exhibit, January 2011.
- [173] J. H. Tu, J. Griffin, A. Hart, C. W. Rowley, L. N. Cattafesta III, and L. S. Ukeiley. Integration of non-time-resolved PIV and time-resolved velocity point sensors for dynamic estimation of velocity fields. *Exp. Fluids*, 54(2):1429, February 2013.
- [174] F. Tubino and G. Solari. Double proper orthogonal decomposition for representing and simulating turbulence fields. *J. Eng. Mech.-ASCE*, 131(12):1302–1312, December 2005.
- [175] T. C. Tung and R. J. Adrian. Higher-order estimates of conditional eddies in isotropic turbulence. *Phys. Fluids*, 23(7):1469–1470, 1980.
- [176] L. S. Ukeiley, N. E. Murray, Q. Song, and L. Cattafesta. Dynamic surface pressure based estimation for flow control. In *IUTAM Symposium on Flow Control and MEMS*, pages 183–189. Springer, 2008.
- [177] H. von Storch, G. Bürger, R. Schnur, and J. S. von Storch. Principal oscillation patterns: a review. *J. Climate*, 8(3):377–400, March 1995.
- [178] A. W. Vreman. An eddy-viscosity subgrid-scale model for turbulent shear flow: Algebraic theory and applications. *Phys. Fluids*, 16(10):3670–3681, October 2004.
- [179] J. Westerweel. Efficient detection of spurious vectors in particle image velocimetry data. *Exp. Fluids*, 16(3–4):236–247, February 1994.
- [180] J. Westerweel and F. Scarano. Universal outlier detection for PIV data. *Exp. Fluids*, 39(6):1096–1100, December 2005.
- [181] J. H. Williamson. Low-storage Runge–Kutta schemes. *J. Comput. Phys.*, 35(1):48–56, 1980.
- [182] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE T. Pattern Anal.*, 31(2):210–227, February 2009.
- [183] J. Z. Wu, X. Y. Lu, A. G. Denny, M. Fan, and J. M. Wu. Post-stall flow control on an airfoil by local unsteady forcing. *J. Fluid Mech.*, 371:21–58, September 1998.
- [184] M. Wu and M. P. Martín. Analysis of shock motion in shockwave and turbulent boundary layer interaction using direct numerical simulation data. *J. Fluid Mech.*, 594:71–83, January 2008.
- [185] I. Wygnanski. Some new observations affecting the control of separation by periodic forcing. AIAA Paper 2000-2314, Fluids 2000, June 2000.

- [186] A. Wynn, D. Pearson, B. Ganapathisubramani, and P. J. Goulart. Optimal mode decomposition for unsteady flows. *J. Fluid Mech.*, 2012 (submitted).
- [187] H. Yu, M. Leeser, G. Tadmor, and S. Siegel. Real-time particle image velocimetry for feedback loops using FPGA implementation. *J. Aerosp. Comput. Inf. Commun.*, 3(2):52–62, 2006.