




Universitetet
i Stavanger

FACULTY OF SCIENCE AND TECHNOLOGY

MASTER'S THESIS

Study programme/specialization: Master of Science in Petroleum Engineering/ Drilling Engineering	Spring semester, 2020. Open access
Author: MohammadAli Shahmoradi	 <hr/> <p>(Author's signature)</p>
Supervisor(s): Dr. Mahmoud Khalifeh Dr. Eric Cayeux	
Title of master's thesis: Predictive Accuracy of Data-Driven Solutions to the Transient Response of the Drilling System	
Credits: 30 ECTS	
Keywords: DMD SVD Data-driven Drilling dynamics Prediction Data reconstruction	Number of pages: 133 Stavanger, 15th July 2020

**Predictive accuracy of data-driven solutions to the transient response of
the drilling system**

By
MohammadAli Shahmoradi

Master's Thesis
Presented to the Faculty of Science and Technology
University of Stavanger

UNIVERSITY OF STAVANGER

JULY 2020

Acknowledgement

This master thesis is written in spring of 2020 as final work of a Master of Science in Petroleum Engineering specialization in Drilling and Well Engineering from University of Stavanger (UiS), Norway.

First of all, I would like to express my gratitude towards NORCE Norwegian Research Centre, especially drilling department for providing data, office and supporting me in any possible way. I would like to thank my supervisor Dr. Mahmoud Khalifeh, Professor at Department of Energy and Petroleum Engineering, University of Stavanger, co-supervisor Dr. Eric Cayeux, Chief Scientist at NORCE for your support, guidance, professional insight and positive feedback along the process of writing this thesis. I must thank Elie Magnon at Total who has guided me in the process of this thesis.

Finally, and most importantly, I must thank my wife and parents for all that they have done. I am forever indebted to them for their many sacrifices over the years which can never be repaid, but I hope that someday I can return the favor.

Abstract

One of the utmost desires of the drilling industry is full drilling automation to minimize well construction cost, optimize well performance and many other advantages. Dynamical system modeling and control of complex systems is undergoing a renaissance, with appearance of data-driven approaches as a result of unprecedentedly availability of high-fidelity measurements from historical records, numerical simulations and experimental data. In this master thesis, one of modal decomposition techniques called Dynamic Mode Decomposition (DMD), a data-driven regression and machine learning method, is introduced and performance of the algorithm in a dynamic of drilling parameter evaluated as few existing literatures focus on drilling applications of this algorithm.

First, basic DMD theory, definition and its classified applications are introduced. In particular, a synthetic example was presented to check the accuracy of the algorithm and become familiar with terms and conditions of it. Afterward, annular fluid velocity dataset is used to characterize the basic DMD algorithm capabilities. Among the analyzed DMD applications are DMD data reconstruction, DMD data interpolation and DMD data prediction. The basic DMD algorithm is able to reconstruct different datasets under various conditions and limitations, however, the primary desire has been the reconstruction of whole dataset as it is the base of all DMD applications. Various advantages and drawbacks of the algorithm are evaluated and good understandings of method are achieved. Furthermore, based on the successful reconstructed intervals, short studies of DMD data interpolation and extrapolation are accomplished, aiming to learn more about the algorithm capabilities. DMD interpolation and extrapolation could be satisfactory as long as the performance of the DMD reconstruction is preferable. Last but not least, as a recommendation, for commercializing DMD algorithm, WDP providers should increase the number of sensors run on WDP to transmit high quality and high-density real-time data, for the reason that the number of measurements in each timestep is an important factor to apply DMD algorithm, which is corresponding to number of sensors along drill-sting.

Acronyms

AFV	Annular Fluid Velocity
ASM	Along-string Measurement
BHA	Bottom Hole Assembly
BOP	Blowout Preventer
DMD	Dynamic Mode Decomposition
DMS	Drilling Modeling and Simulation
LWD	Logging While Drilling
MWD	Measurement While Drilling
NPT	Non-Productive Time
POD	Proper Orthogonal Decomposition
SVD	Singular Value Decomposition
WDP	Wired Drill Pipe

Table of Contents

- Acknowledgement..... iii
- Abstract iv
- Acronyms v
- Table of Contents vi
- List of Figures viii
- List of Tables..... xiii
- 1 Introduction 14
- 2 Background Theory and Solution Method 15
 - 2.1 Dynamic mode decomposition (DMD)..... 18
 - 2.2 Formulating the DMD algorithm 19
 - 2.2.1 Step by step guide for DMD algorithm in practice 21
 - 2.2.2 DMD example 25
 - 2.3 Optimal amplitudes of DMD modes 31
- 3 Case Studies, Results and Discussions..... 33
 - 3.1 Simulated well..... 34
 - 3.2 Case study: Annular fluid velocity (AFV) during trip out 36
 - 3.2.1 DMD data reconstruction 38
 - 3.2.1.1 DMD reconstruction of single sub dataset 39
 - 3.2.1.2 DMD reconstruction of multiple sub datasets 47
 - 3.2.1.3 DMD reconstruction of multiple sub datasets without acceleration 74
 - 3.2.2 DMD data interpolation 80
 - 3.2.3 DMD data extrapolation or prediction 85
 - 3.2.4 Summary of DMD implementation in AFV case study 89
 - 3.3 Future studies 89
 - 3.4 DMD applicability and current technologies in the industry 89
- 4 Conclusions and Recommendations..... 91

Appendix A – Eigenvalue and eigenvector.....	93
Appendix B – Singular value decomposition (SVD).....	94
Appendix C – Overdetermined and underdetermined system	97
Appendix D – DMD reconstructed result of single sub dataset, interval No. 2 – 5.....	98
Appendix E – Python codes	122
References	132

List of Figures

FIGURE 2.1 ELEMENTS OF MATHEMATICAL MODELS FOR OBTAINING SATISFACTORY RESULTS (BJØRKEVOLL, 2015B).....	17
FIGURE 2.2 SCHEMATIC REVIEW OF THE DMD ALGORITHM (KUTZ ET AL., 2016).....	21
FIGURE 2.3 STEP BY STEP DMD ALGORITHM SUMMARY (KUTZ ET AL., 2016).....	24
FIGURE 2.4 SPATIOTEMPORAL DYNAMICS OF f_1, f_2 AND F	25
FIGURE 2.5 THE SINGULAR VALUE DECOMPOSITION (SVD) OF THE DATASET.....	26
FIGURE 2.6 THE SPECTRUM OF SINGULAR VALUES OF $F(x,T)$	27
FIGURE 2.7 EIGENVALUES OF THE DYNAMIC SYSTEM ON THE UNIT CIRCLE.....	28
FIGURE 2.8 THE SPATIAL MODES AND THEIR DYNAMICS IN TIME.....	28
FIGURE 2.9 THE DOMINANT DMD MODES AND DMD RECONSTRUCTED DATASET.....	29
FIGURE 2.10 THE FREQUENCY DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR.....	30
FIGURE 2.11 THE DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR.....	30
FIGURE 3.1 DOWNHOLE REAL TIME DATA SOURCES(ISRAEL ET AL., 2018).....	33
FIGURE 3.2 SIMULATED WELL ARCHITECTURE.....	34
FIGURE 3.3 WELLBORE TRAJECTORY.....	35
FIGURE 3.4 DRILL-STRING AND BHA DESCRIPTION.....	35
FIGURE 3.5 DRILLING FLUID REPORT.....	36
FIGURE 3.6 ANNULAR FLUID VELOCITY DATASET.....	37
FIGURE 3.7 ANNULAR FLUID VELOCITY, INTERVAL No. 1 (0 - 43).....	40
FIGURE 3.8 SINGULAR VALUE DECOMPOSITION (SVD) OF INTERVAL 1(0 - 43).....	41
FIGURE 3.9 SINGULAR VALUES OF INTERVAL No. 1 (0 - 43).....	41
FIGURE 3.10 THE SPATIAL MODES AND THEIR DYNAMICS IN TIME, INTERVAL 1 (0 - 43).....	42
FIGURE 3.11 EIGENVALUES OF INTERVAL 1 (0 - 43).....	43
FIGURE 3.12 DMD RECONSTRUCTION OF THE INTERVAL No. 1 (0 - 43).....	44
FIGURE 3.13 ORIGINAL DATA INTERVAL No. 1 (0 - 43).....	44
FIGURE 3.14 DMD RECONSTRUCTED DATA ERROR INTERVAL No. 1 (0 - 43).....	45
FIGURE 3.15 THE FREQUENCY DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, INTERVAL No. 1 (0 - 43).....	45

FIGURE 3.16 THE DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, INTERVAL No. 1 (0 - 43)	46
FIGURE 3.17 ANNULAR FLUID VELOCITY, MERGED INTERVAL No. 1 (0 - 111)	48
FIGURE 3.18 SINGULAR VALUE DECOMPOSITION (SVD) OF MERGED INTERVAL 1 (0 - 111)	49
FIGURE 3.19 SINGULAR VALUES OF MERGED INTERVAL No. 1 (0 - 111)	49
FIGURE 3.20 THE SPATIAL MODES AND THEIR DYNAMICS IN TIME, MERGED INTERVAL 1 (0 - 100)	50
FIGURE 3.21 EIGENVALUES OF MERGED INTERVAL 1 (0 - 111)	51
FIGURE 3.22 DMD RECONSTRUCTION OF MERGED INTERVAL No. 1 (0 - 111)	52
FIGURE 3.23 ORIGINAL DATA, MERGED INTERVAL No. 1 (0 - 111)	52
FIGURE 3.24 DMD RECONSTRUCTED DATA ERROR, MERGED INTERVAL No. 1 (0 - 111)	53
FIGURE 3.25 THE FREQUENCY DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, MERGED INTERVAL No. 1 (0 - 111)	54
FIGURE 3.26 THE DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, MERGED INTERVAL No. 1 (0 - 111)	54
FIGURE 3.27 ANNULAR FLUID VELOCITY, MERGED INTERVAL No. 2 (0 - 140)	55
FIGURE 3.28 SINGULAR VALUE DECOMPOSITION (SVD) OF MERGED INTERVAL 2 (0 - 140)	56
FIGURE 3.29 SINGULAR VALUES OF MERGED INTERVAL No. 2 (0 - 140)	56
FIGURE 3.30 THE SPATIAL MODES AND THEIR DYNAMICS IN TIME, MERGED INTERVAL 2 (0 - 140)	57
FIGURE 3.31 EIGENVALUES OF MERGED INTERVAL 2 (0 - 140)	57
FIGURE 3.32 DMD RECONSTRUCTION OF MERGED INTERVAL No. 2 (0 - 140)	58
FIGURE 3.33 ORIGINAL DATA, MERGED INTERVAL No. 2 (0 - 140)	58
FIGURE 3.34 DMD RECONSTRUCTED DATA ERROR, MERGED INTERVAL No. 2 (0 - 140)	59
FIGURE 3.35 THE FREQUENCY DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, MERGED INTERVAL No. 2 (0 - 140)	60
FIGURE 3.36 THE DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, MERGED INTERVAL No. 2 (0 - 140)	60
FIGURE 3.37 ANNULAR FLUID VELOCITY, MERGED INTERVAL No. 3 (0 - 161)	61
FIGURE 3.38 SINGULAR VALUE DECOMPOSITION (SVD) OF MERGED INTERVAL 3 (0 - 161)	62
FIGURE 3.39 SINGULAR VALUES OF MERGED INTERVAL No. 3 (0 - 161)	62
FIGURE 3.40 THE SPATIAL MODES AND THEIR DYNAMICS IN TIME, MERGED INTERVAL 3 (0 - 161)	63
FIGURE 3.41 EIGENVALUES OF MERGED INTERVAL 3 (0 - 161)	64

FIGURE 3.42 DMD RECONSTRUCTION OF MERGED INTERVAL No. 3 (0 - 161).....	65
FIGURE 3.43 ORIGINAL DATA, MERGED INTERVAL No. 3 (0 - 161).....	65
FIGURE 3.44 DMD RECONSTRUCTED DATA ERROR, MERGED INTERVAL No. 3 (0 – 161)	66
FIGURE 3.45 THE FREQUENCY DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, MERGED INTERVAL No. 3 (0 – 161)	66
FIGURE 3.46 THE DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, MERGED INTERVAL No. 3 (0 - 161).....	67
FIGURE 3.47 THE SPATIAL MODES AND THEIR DYNAMICS IN TIME, CORRECTED MERGED INTERVAL 3 (0 - 156) ...	68
FIGURE 3.48 EIGENVALUES OF CORRECTED MERGED INTERVAL 3 (0 - 156)	69
FIGURE 3.49 DMD RECONSTRUCTION OF CORRECTED MERGED INTERVAL No. 3 (0 - 156).....	70
FIGURE 3.50 ORIGINAL DATA, CORRECTED MERGED INTERVAL No. 3 (0 - 156).....	70
FIGURE 3.51 DMD RECONSTRUCTED DATA ERROR, CORRECTED MERGED INTERVAL No. 3 (0 – 156).....	71
FIGURE 3.52 THE FREQUENCY DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, CORRECTED MERGED INTERVAL No. 3 (0 – 156)	72
FIGURE 3.53 THE DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, CORRECTED MERGED INTERVAL No. 3 (0 - 156)	72
FIGURE 3.54 ANNULAR FLUID VELOCITY, MERGED INTERVAL No. 1 (45 - 140) WITHOUT ACCELERATION DATA ..	74
FIGURE 3.55 SINGULAR VALUE DECOMPOSITION (SVD) OF MERGED INTERVAL 1(45 - 140) WITHOUT ACCELERATION.....	75
FIGURE 3.56 SINGULAR VALUES OF MERGED INTERVAL No. 1 (45 - 140) WITHOUT ACCELERATION DATA	75
FIGURE 3.57 THE SPATIAL MODES AND THEIR DYNAMICS IN TIME, MERGED INTERVAL 1 (45 - 140) WITHOUT ACCELERATION DATA	76
FIGURE 3.58 EIGENVALUES OF MERGED INTERVAL 1 (45 - 140) WITHOUT ACCELERATION DATA	76
FIGURE 3.59 DMD RECONSTRUCTION OF MERGED INTERVAL No. 1 (45 - 140) WITHOUT ACCELERATION DATA ..	77
FIGURE 3.60 ORIGINAL DATA, MERGED INTERVAL No. 1 (45 - 140) WITHOUT ACCELERATION DATA	77
FIGURE 3.61 DMD RECONSTRUCTED DATA ERROR MERGED INTERVAL No. 1 (45 – 140) WITHOUT ACCELERATION DATA	78
FIGURE 3.62 THE FREQUENCY DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, MERGED INTERVAL No. 1 (45 – 140) WITHOUT ACCELERATION DATA	79
FIGURE 3.63 THE DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, MERGED INTERVAL No. 1 (45 - 140) WITHOUT ACCELERATION DATA	79
FIGURE 3.64 DMD RECONSTRUCTED DATA ERROR, (45 – 225) ODD COLUMNS.....	81

FIGURE 3.65 THE FREQUENCY DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, (45 – 225) ODD COLUMNS.....	82
FIGURE 3.66 THE DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, (45 - 225) ODD COLUMNS	82
FIGURE 3.67 DMD RECONSTRUCTION (ODD TIME SNAPSHOTS) AND INTERPOLATION (EVEN TIME SNAPSHOTS) , (45 - 225).....	83
FIGURE 3.68 ORIGINAL DATA, (45 - 225)	83
FIGURE 3.69 DMD INTERPOLATED DATA ERROR, (45 - 225).....	84
FIGURE 3.70 THE FREQUENCY DISTRIBUTION OF THE ABSOLUTE DMD INTERPOLATED DATA ERROR, (45 - 225) .	84
FIGURE 3.71 THE DISTRIBUTION OF THE ABSOLUTE DMD INTERPOLATED DATA ERROR, (45 - 225)	85
FIGURE 3.72 DMD RECONSTRUCTED (0 - 140) AND EXTRAPOLATED (141 - 148) DATA	86
FIGURE 3.73 ORIGINAL DATA INTERVAL (0 - 148).....	86
FIGURE 3.74 EIGENVALUES OF RECONSTRUCTED DATASET (45 – 140) FOR EXTRAPOLATION	87
FIGURE 3.75 THE FREQUENCY DISTRIBUTION OF THE ABSOLUTE DMD EXTRAPOLATED DATA ERROR	88
FIGURE 3.76 THE DISTRIBUTION OF THE ABSOLUTE DMD EXTRAPOLATED DATA ERROR	88
APPEND. B.1 FULL SINGULAR VALUE DECOMPOSITION (SVD)((BRUNTON & KUTZ, 2019))	94
APPEND. B.2 ECONOMY SINGULAR VALUE DECOMPOSITION (SVD)(BRUNTON & KUTZ, 2019).....	95
APPEND. B.3 TRUNCATED SINGULAR VALUE DECOMPOSITION (SVD)(BRUNTON & KUTZ, 2019)	95
APPEND. B. 4 GEOMETRIC ILLUSTRATION OF THE SVD AS MAPPING FROM A SPHERE TO AN ELLIPSOID (BRUNTON & KUTZ, 2019)	96
APPEND. D.1 ANNULAR FLUID VELOCITY, INTERVAL NO. 2 (44 - 111).....	98
APPEND. D.2 SINGULAR VALUE DECOMPOSITION (SVD) OF INTERVAL 2(44 - 111).....	99
APPEND. D.3 SINGULAR VALUES OF INTERVAL NO. 2 (44 - 111)	99
APPEND. D.4 THE SPATIAL MODES AND THEIR DYNAMICS IN TIME, INTERVAL 2 (44 - 111).....	100
APPEND. D.5 EIGENVALUES OF INTERVAL 2 (44 - 111)	100
APPEND. D.6 DMD RECONSTRUCTION OF THE INTERVAL NO. 2 (44 - 111)	101
APPEND. D.7 ORIGINAL DATA INTERVAL NO. 2 (44 - 111)	101
APPEND. D.8 DMD RECONSTRUCTED DATA ERROR INTERVAL NO. 2 (44 - 111).....	102
APPEND. D.9 THE FREQUENCY DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, INTERVAL No. 2 (44 – 111)	102

APPEND. D.10 THE DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, INTERVAL No. 2 (44 - 111).....	103
APPEND. D.11 ANNULAR FLUID VELOCITY, INTERVAL No. 3 (112 - 140).....	104
APPEND. D.12 THE SPATIAL MODES AND THEIR DYNAMICS IN TIME, INTERVAL 3 (112 - 140).....	105
APPEND. D.13 SINGULAR VALUE DECOMPOSITION (SVD) OF INTERVAL 3 (112 - 140).....	105
APPEND. D.14 SINGULAR VALUES OF INTERVAL No. 3 (112 - 140).....	106
APPEND. D.15 EIGENVALUES OF INTERVAL 3 (112 - 140).....	106
APPEND. D.16 DMD RECONSTRUCTION OF THE INTERVAL No. 3 (112 - 140).....	107
APPEND. D.17 ORIGINAL DATA INTERVAL No. 3 (112 - 140).....	107
APPEND. D.18 DMD RECONSTRUCTED DATA ERROR INTERVAL No. 3 (112 - 140).....	108
APPEND. D.19 THE FREQUENCY DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, INTERVAL No. 3 (112 – 140).....	108
APPEND. D.20 THE DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, INTERVAL No. 3 (112 - 140).....	109
APPEND. D.21 THE DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, INTERVAL No. 4 (141 - 161).....	110
APPEND. D.22 THE SPATIAL MODES AND THEIR DYNAMICS IN TIME, INTERVAL 4 (141 - 161).....	111
APPEND. D.23 SINGULAR VALUE DECOMPOSITION (SVD) OF INTERVAL 4 (141 - 161).....	111
APPEND. D.24 SINGULAR VALUES OF INTERVAL No. 4 (141 - 161).....	112
APPEND. D.25 EIGENVALUES OF INTERVAL 4 (141 - 161).....	112
APPEND. D.26 DMD RECONSTRUCTION OF THE INTERVAL No. 4 (141 - 161).....	113
APPEND. D.27 ORIGINAL DATA INTERVAL No. 4 (141 - 161).....	113
APPEND. D.28 DMD RECONSTRUCTED DATA ERROR INTERVAL No. 4 (141 - 161).....	114
APPEND. D.29 THE FREQUENCY DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, INTERVAL No. 4 (141 – 161).....	114
APPEND. D.30 THE DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, INTERVAL No. 4 (141 - 161).....	115
APPEND. D.31 ANNULAR FLUID VELOCITY, INTERVAL No. 5 (162 - 185).....	116
APPEND. D.32 THE SPATIAL MODES AND THEIR DYNAMICS IN TIME, INTERVAL 5 (162 - 185).....	117
APPEND. D.33 SINGULAR VALUE DECOMPOSITION (SVD) OF INTERVAL 5 (162 - 185).....	117
APPEND. D.34 SINGULAR VALUES OF INTERVAL No. 5 (162 - 185).....	118
APPEND. D.35 EIGENVALUES OF INTERVAL 5 (162 - 185).....	118

APPEND. D.36 DMD RECONSTRUCTION OF THE INTERVAL No. 5 (162 - 185).....	119
APPEND. D.37 ORIGINAL DATA INTERVAL No. 5 (162 - 185)	119
APPEND. D.38 DMD RECONSTRUCTED DATA ERROR INTERVAL No. 5 (162 - 185).....	120
APPEND. D.39 THE FREQUENCY DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, INTERVAL No. 5 (162 –185).....	120
APPEND. D.40 THE DISTRIBUTION OF THE ABSOLUTE DMD RECONSTRUCTED DATA ERROR, INTERVAL No. 5 (162 - 185).....	121

List of Tables

TABLE 1. ANNULAR FLUID VELOCITY SUB DATASETS.....	38
TABLE 2. ERROR DISTRIBUTION OF DMD RECONSTRUCTION SUB DATASETS	46
TABLE 3. ERROR DISTRIBUTION OF DMD RECONSTRUCTION DATA, MERGED INTERVALS	73
TABLE 4. ERROR DISTRIBUTION OF DMD RECONSTRUCTION DATA, MERGED INTERVALS WITHOUT ACCELERATION	80

1 Introduction

Drilling of wells is a basic process in hydrocarbon production. It involves multi-physics aspects such as hydraulics, mechanics, heat transfer and mass transfer with non-linear abrupt changes of the surface and downhole boundary conditions. Primary analysis of drilling experiences is often being used in designing of the conventional drilling, which is not effective in modern drilling and it has caused loss of assets, health, safety and environment. Complex real-time high-fidelity models of the drilling process have been developed, as a result of the advancement in downhole measurement while drilling and experimental data. These models may add significant value if implemented in a way that handles the various related challenges adequately (Bjørkevoll, 2015a). However, it is difficult to run such complex models much faster than real-time as a result of the high-dimensional, nonlinear dynamical systems. All challenges and complexities of the mathematical models have induced research in other techniques such as data-driven techniques.

The recently developed dynamic mode decomposition (DMD) algorithm, originated from fluid dynamic community, is an innovative tool for integrating data with dynamical systems theory and has become a widely used technique in fluid dynamics (Schmid, 2011, p. 31). Jets, cavity flow, wakes, channel flow, boundary layers are examples of flow geometries that has been studied by DMD to understand mixing, acoustics, and combustion among other phenomena, and it may be used for short-time future state prediction and control (Kutz, Brunton, Brunton, & Proctor, 2016). Bao and Gildin (2017) successfully used DMD to capture and predict the behavior of reservoir fluid flow in porous media.

The main objective of this master thesis is to investigate the possibility of applying DMD method for drilling applications in the scope of the project that will be explain in the following. Chapter 2 reviews the DMD theory and background. For better understanding of the algorithm, a synthetic dataset is analyzed to demonstrate the utility of this generalized theory. The rest of this master thesis focuses on applications of DMD in practice. In chapter 3, the DMD algorithm is applied to a case study in the dynamics of drilling parameters and accuracy of the DMD for different applications such as original data reconstruction, interpolation and prediction is analyzed finally applicability of the current downhole drilling data acquisition technologies for the DMD applications are discussed. In Chapter 4, the main findings of this master thesis are summarized and conclusions are drawn on whether the DMD is viable solution for drilling applications or not.

2 Background Theory and Solution Method

Drilling industry is always looking for solutions for improving the drilling performance, increasing safety and reducing non-productive time (NPT). Drilling Modeling and Simulation (DMS) for analysis and control of the drilling process have become an attractive approach to deliver solutions. DMS involves modeling and simulating the behavior of drilling systems and/or processes, and it provides crucial information without actually constructing a well. DMS has widely been used for improving drilling systems automation and control, managed pressure drilling and drilling optimization by understanding and predicting downhole dynamics, thanks to the aggressive development. The mechanical, hydraulic and heat transfer domains are the usual sections of drilling modelling. Changing boundary conditions of the drilling system encourage using transient models instead of steady state models, which leads to perform accurate simulation of the real-time operation according to the detailed wellbore properties (Dvergsnes & Cayeux, 2019; Sugiura et al., 2015).

The drilling process is described by mass, momentum, and energy conservation equations, and force and torque equilibriums that are used for mechanistic modeling. One-dimensional versions of these equations are succinctly mentioned.

The mass conservation can be expressed as follows:

$$\frac{\partial}{\partial t}(A\rho) + \frac{\partial}{\partial s}(A\rho v) = \dot{m}, \quad (2.1)$$

where t , s are time and space dimensions, A , ρ , v are the cross-sectional area, density and velocity and \dot{m} is the mass flux through the wellbore wall.

The Navier-Stokes equation for a single-component, single-phase fluid can be simplified to:

$$\frac{\partial}{\partial t}(A\rho v) + \frac{\partial}{\partial s}(A\rho v^2) + A \frac{\partial p}{\partial s} = A\rho g \cos \theta + AK(v) + \dot{m}v_{\dot{m}}, \quad (2.2)$$

where p is the pressure, g is the gravitational acceleration, θ is the inclination, $v_{\dot{m}}$ is velocity of the mass flux along wellbore wall, $K(v)$ represents the friction loss and it is a positive function of the velocity, depending on the fluid's rheological behavior and the local geometry.

The Fourier equation in a volumetric form can be written as:

$$\rho C_p \frac{\partial T}{\partial t} + \rho V(1 - \alpha T) \frac{\partial p}{\partial t} - \rho v C_p \frac{\partial T}{\partial s} - \rho v V(1 - \alpha T) \frac{\partial p}{\partial s} - \nabla(\lambda \nabla T) = q_s \quad (2.3)$$

where T is the temperature, C_p is the specific heat capacity, V is the elementary volume, $\alpha = \frac{1}{V} \left(\frac{\partial V}{\partial T} \right)_p$ is the coefficient of thermal expansion, λ is the thermal conductivity and q_s is the heat generated by mechanical and hydraulic friction per unit volume.

Newton's equations of motion on force and torques along the drill-string can be expressed as:

$$EA \frac{\partial \vec{u}}{\partial s} + \vec{F}_g + \vec{F}_p + \vec{F}_a + \vec{F}_v + \vec{F}_{\mu_k} + \vec{F}_c + \vec{F}_e + \vec{R} = \rho A \frac{\partial^2 \vec{u}}{\partial t^2} \quad (2.4)$$

$$GI \frac{\partial \vec{\beta}}{\partial s} + \vec{t} \times \vec{T} + \vec{C}_v + \vec{C}_{\mu_k} + \vec{C}_e = \rho I \frac{\partial^2 \vec{\beta}}{\partial t^2} \quad (2.5)$$

where E is the Young modulus, \vec{u} is the displacement, $\vec{F}_g, \vec{F}_p, \vec{F}_a, \vec{F}_v, \vec{F}_{\mu_k}, \vec{F}_c, \vec{F}_e$ are the gravitational, pressure, fluid acceleration, viscous friction, kinetic friction, centrifugal, and Euler forces applied to the control element, \vec{R} is the reaction force at the contact with the borehole wall, G is the shear modulus, I is the second moment of area around the axis of rotation, $\vec{\beta}$ is the rotation of a section of the control element, \vec{T} is tension vector, \vec{t} is the tangential unit vector of the Frenet-Serret coordinate system associated to the control element, \times denotes the vectoral cross-product, $\vec{C}_v, \vec{C}_{\mu_k}, \vec{C}_e$ are viscous friction, kinetic friction and Euler torques applied to the control element. It should be noted that many of the coefficients in above equations are a function of pressure and/or temperature such as density $\rho(p, T)$, the hydraulic friction $K(v, p, T)$, the specific heat capacity $C_p(p, T)$, the thermal conductivity $\lambda(p, T)$, the coefficient of thermal expansion $\alpha(p, T)$, young modulus $E(T)$ and shear modulus $G(T)$ (Cayeux et al., 2018).

In order to develop qualified mathematical models with high level of accuracy and proper for real time purposes, number of elements need to be developed (see Figure 2.1).

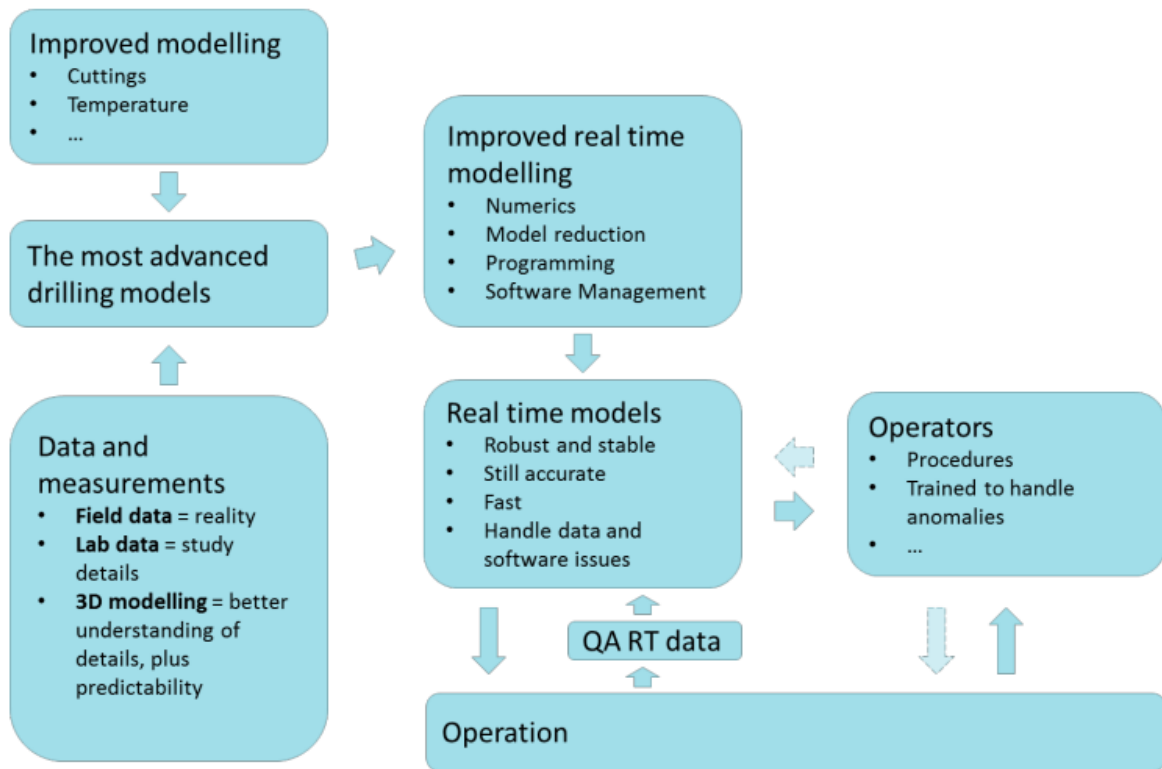


Figure 2.1 Elements of mathematical models for obtaining satisfactory results (Bjørkevoll, 2015b).

As shown in Figure 2.1, number of items need to be addressed to improve real-time models, and they should be implemented in a robust way that model characteristics such as accuracy, predictability, calculation speed, stability and others are not sacrificed. It is important to have a model that works in spite of the complexities of the modelling (Bjørkevoll, 2015b). As mentioned, mass conservation equation (2.16), Navier Stokes equation (2.2), Fourier equation (2.3), and Newton’s equations (2.4, 2.5) have parameters that are function of pressure, temperature and velocity, which influence each other, and add more complexities to the modelling calculations. These equations also indicate that we are dealing with chaotic systems, which are known as sensitive dependence on initial and boundary conditions. According to chaos theory, deterministic predictability is false for most systems, and small uncertainties in initial states can indeed become large errors at last and as a result this causes the appearance of randomness in the system. So this means, at any time one of two or more events can happen randomly next (Motter & Campbell, 2013). Furthermore, for control applications, sometimes it is necessary to have fast response of the non-linear models in real-time situation. All challenges and computational complexities of the mathematical models have induced research in other techniques for real time applications.

Dynamical system modeling and control of complex systems is undergoing a renaissance, with appearance of data-driven approaches as a result of unprecedentedly availability of high-fidelity measurements from historical records, numerical simulations and experimental data. Manipulating such data to find most obvious trends needs a skillset that can take a dataset and characterize it in meaningful ways. In this master thesis, one of modal decomposition techniques called Dynamic Mode Decomposition (DMD), a data-driven regression and machine learning method, is introduced. This technique takes a set of data and discover dynamical systems from the data. DMD is a new data-based algorithm, which computes a set of modes from data and identifies features that explain the underlying physics in a dynamical system (Kutz et al., 2016; Jonathan H. Tu, 2013). In this chapter, the core DMD algorithm is presented. It is the foundation for different DMD methods and future innovations and applications of this algorithm. Wherever possible, attempts are made to be clear and simple as possible, and some preliminary mathematical concepts are covered in appendices.

2.1 Dynamic mode decomposition (DMD)

Dynamic mode decomposition was developed by Schmid in the fluid dynamic community to find spatio-temporal coherent structures from measured data (Brunton & Kutz, 2019). DMD is a combination of spatial dimensionality-reduction technique, using the computationally efficient singular value decomposition (SVD¹), with Fourier transform which decomposes experimental data into a set of dynamic modes that are derived from collected snapshots of data, and also provides a model for how these modes evolve in time. Basically, DMD is a regression method to find the best-fit linear dynamical system from measurement data in time, even if the dynamics is nonlinear (Kutz, 2013; Kutz et al., 2016). Considerable interest has been generated to DMD algorithm as it only relies on measured data without any knowledge of the governing equations (Brunton & Kutz, 2019).

The DMD algorithm has three main applications:

- I. **Diagnostics.** Ability of the algorithm to extract low-rank features of high-dimensional systems, allowing for physical interpretation of system in case of spatial structures and temporal response.
- II. **State estimation and future-state prediction.** By using dominant spatiotemporal structures of data in DMD algorithm, it is possible to construct dynamical models of the

¹ This notion is explained more precise in Appendix B

underlying processes and predict the state of the system in future where no measurements were made.

- III. Control.** As linear dynamical model is used to predict the future state of a nonlinear dynamical system, there is limited short-time window in future where DMD model and real dynamics agree. As a result, control goal of the DMD algorithm is highly challenging. If precise prediction window is long enough, it is possible to use DMD for control strategies (Kutz et al., 2016).

2.2 Formulating the DMD algorithm

A dynamical system is generally defined

$$\frac{\partial x}{\partial t} = f(x, t, \mu), \quad (2.6)$$

where $x(t) \in \mathbb{R}^n$ is an n -dimensional vector ($n \gg 1$) representing the state of our dynamical system at time t , μ is parameters of the system, and f represents the dynamics which is often nonlinear differential equation. In general, it is impossible to find a governing equation for nonlinear dynamics of dynamical system (2.6). DMD is used to approximate the dynamics, without using any equation, just by using measured data.

The first step is to collect a number of snapshots of the state of systems in equal time interval Δt so that $x_k = x(t_k)$ at $t_k = k\Delta t$ for $k=1, 2, 3, \dots, m$. The measured data matrix could be like:

$$\begin{bmatrix} x_1(t_1) & x_1(t_2) & x_1(t_3) & \dots & x_1(t_m) \\ x_2(t_1) & x_2(t_2) & x_2(t_3) & \dots & x_2(t_m) \\ x_3(t_1) & x_3(t_2) & x_3(t_3) & \dots & x_3(t_m) \\ \vdots & \vdots & \vdots & \dots & \vdots \\ x_n(t_1) & x_n(t_2) & x_n(t_3) & \dots & x_n(t_m) \end{bmatrix}_{n \times m} \quad (2.7)$$

which $x_{(t_1)}$ is the initial conditions of the dynamical system. The timestep Δt should be small enough to record the highest frequencies in the measurements.

The DMD method approximate dynamical system (2.6) as a locally linear dynamical system

$$\frac{\partial x}{\partial t} = Ax \quad (2.8)$$

which searches the leading spectral decomposition (eigenvalues, eigenvectors)² of A , the best-fit linear operator that relates two modes in time. Assuming uniform sampling of modes in time, the discrete version of equation (2.8):

² This notion is explained more precise in Appendix A

$$x_{(t_{k+1})} \approx Ax_{(t_k)} \quad (2.9)$$

By calculating eigenvalues (λ_k) and eigenvectors (Φ_k) of the operator A, the solution for dynamical system (2.6) could be expressed as:

$$x_k = \sum_{j=1}^r \Phi_j \lambda_j^k b_j = \Phi \Lambda^k b \quad k = 1, 2, \dots, m-1 \quad (2.10)$$

which $x_1 = \Phi b$ is the initial condition in eigenvector basis and b are the coefficients of the initial condition, and Λ is eigenvalues in reduced version of A. DMD algorithm is capable to provide a low-rank eigendecomposition of matrix A which fits x_k for $k=1,2,3,\dots, m$ in least-square sense:

$$\|X_{k+1} - AX_k\|_2 \quad (2.11)$$

So, it is minimized across all points for $k=1, 2, 3, \dots, m-1$. To minimize the equation (2.11), approximation error, measured data matrix (2.7) is then arranged into two matrices, X and X', for the DMD algorithm.

$$X = \begin{bmatrix} x_1(t_1) & x_1(t_2) & x_1(t_3) & \dots & x_1(t_{m-1}) \\ x_2(t_1) & x_2(t_2) & x_2(t_3) & \dots & x_2(t_{m-1}) \\ x_3(t_1) & x_3(t_2) & x_3(t_3) & \dots & x_3(t_{m-1}) \\ \vdots & \vdots & \vdots & \dots & \vdots \\ x_n(t_1) & x_n(t_2) & x_n(t_3) & \dots & x_n(t_{m-1}) \end{bmatrix}_{n \times (m-1)} \quad (2.12)$$

$$X' = \begin{bmatrix} x_1(t_2) & x_1(t_3) & x_1(t_4) & \dots & x_1(t_m) \\ x_2(t_2) & x_2(t_3) & x_2(t_4) & \dots & x_2(t_m) \\ x_3(t_2) & x_3(t_3) & x_3(t_4) & \dots & x_3(t_m) \\ \vdots & \vdots & \vdots & \dots & \vdots \\ x_n(t_2) & x_n(t_3) & x_n(t_4) & \dots & x_n(t_m) \end{bmatrix}_{n \times (m-1)} \quad (2.13)$$

The linear approximation of the dynamical system (2.9) could be written:

$$X' \approx AX \quad (2.14)$$

And Matrix A is achieved by:

$$A = X'X^\dagger \quad (2.15)$$

which the error is $\|X' - AX\|_F$, Frobenius norm, and could be calculated by:

$$\|X\|_F = \sqrt{\sum_{j=1}^n \sum_{k=1}^m X_{jk}^2} \quad (2.16)$$

The size of the measured data matrix depends on how many time snapshots is planned to record, but for DMD algorithm, it is supposed to have overdetermined system³, which means the number of rows or constraints of X is greater than the number of columns or variables. Matrix A may have high dimension, and as a result the decomposition of A becomes difficult. To solve the problem, measured data is projected onto a low-rank subspace defined by the $m-1$ modes, computed by singular value decomposition (SVD), and calculations continue with a low-rank matrix \tilde{A} . The DMD algorithm reconstructs the eigenvalues and eigenvectors of matrix A by using low-rank operator \tilde{A} (Brunton & Kutz, 2019; Kutz, 2013; Kutz et al., 2016). Schematic of DMD method shown in Figure 2.2.

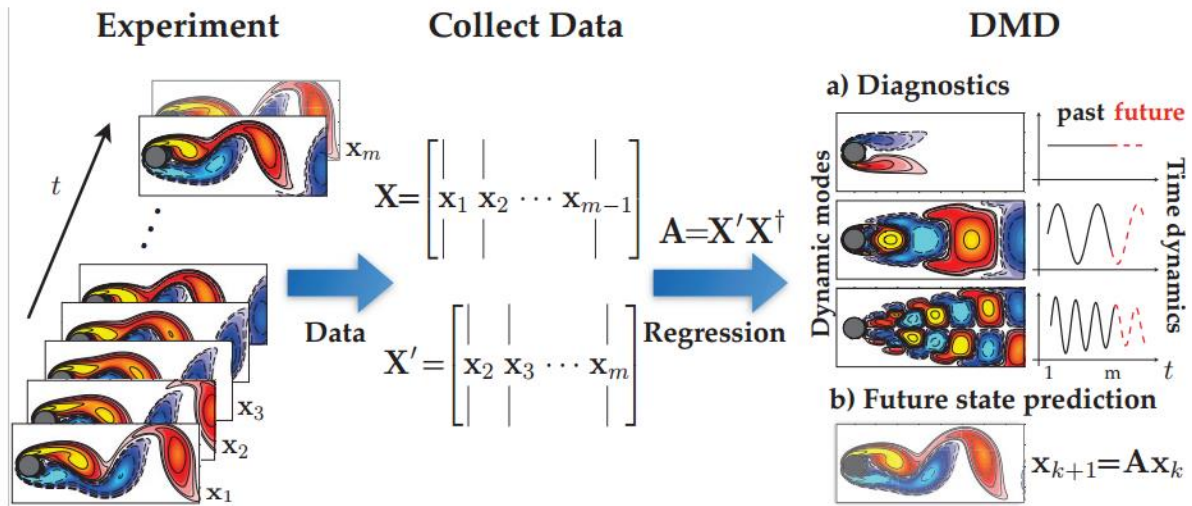


Figure 2.2 Schematic review of the DMD algorithm (Kutz et al., 2016).

DMD definition in short: For the prepared X (2.12) and X' (2.13) dataset of dynamical system (2.6), which $x_{k+1} = F(x_k)$ and F is the map of time snapshots in time evolution, DMD computes the leading eigendecomposition of the best-fit linear operator A that $X' \approx AX$. Then the DMD modes (or the eigenvectors of A) corresponding to particular eigenvalues of A is achieved (Jonathan H Tu, Rowley, Luchtenburg, Brunton, & Kutz, 2013). DMD modes are used for different applications such as original data reconstruction, interpolation, and future state or mode prediction.

2.2.1 Step by step guide for DMD algorithm in practice

1. Take the singular value decomposition (SVD) of X :

³This notion is explained more precise in Appendix C

$$X \approx U\Sigma V^* \quad (2.17)$$

where $*$ denotes conjugate transpose, $U \in \mathbb{C}^{n \times r}$, $\Sigma \in \mathbb{C}^{r \times r}$, $V \in \mathbb{C}^{m \times r}$, $r \leq m$ is either the exact or approximate rank of the data matrix X . The columns of U are known as proper orthogonal decomposition (POD) modes and they satisfy $U^*U=I$ as the columns are orthonormal. In a similar way, $V^*V=I$.

2. Compute A by using pseudoinverse of X obtained by SVD:

$$A = X'V\Sigma^{-1}U^* \quad (2.18)$$

As mentioned before, it is not efficient to directly calculate A , so a low-dimensional linear model of the dynamical system, \tilde{A} , is obtained by using $r \times r$ projection of the full matrix A onto POD modes in U :

$$\tilde{A} = U^*AU = U^*X'V\Sigma^{-1} \quad (2.19)$$

$$\tilde{x}_{k+1} = \tilde{A}\tilde{x}_k \quad (2.20)$$

It should be mentioned that full dataset X could be reconstructed from the reduced states:

$$\tilde{x}_k: x = U\tilde{x} \quad (2.21)$$

3. Compute the eigendecomposition of \tilde{A} :

$$\tilde{A}W = W\Lambda \quad (2.22)$$

where diagonal matrix Λ contains the DMD eigenvalues (λ_k) of \tilde{A} , which correspond to eigenvalues of the full A matrix. The columns of W are eigenvectors of \tilde{A} and provide a coordinate transformation that diagonalizes the matrix. The columns of eigenvectors are linear combinations of mode amplitudes that behave linearly with single temporal given by λ .

4. Reconstruct eigenvectors of the A matrix using eigenvectors of the reduced version W (Brunton & Kutz, 2019; Kutz, 2013; Kutz et al., 2016):

$$\Phi = X'V\Sigma^{-1}W \quad (2.23)$$

As mentioned in Step 3, eigenvalues of matrix A and reduced version \tilde{A} are equal which is demonstrated as follows (Jonathan H Tu et al., 2013):

$$\begin{aligned} A\Phi &= (X'V\Sigma^{-1}U^*)(X'V\Sigma^{-1}W) \\ &= X'V\Sigma^{-1}\tilde{A}W \\ &= X'V\Sigma^{-1}W\Lambda \\ &= \Phi\Lambda \end{aligned}$$

5. Approximate solution at all future times:

$$x(t) \approx \sum_{k=1}^r \Phi_k \exp(\omega_k) b_k = \Phi \exp(\Omega t) b, \quad (2.24)$$

where b_k is the initial amplitude of each mode, Φ is matrix whose columns are the DMD modes or eigenvectors Φ_k , and $\Omega = \text{diag}(\omega)$ is a diagonal matrix that contains eigenvalues $\omega_k = \ln(\lambda_k)/\Delta t$. The b is a vector contains coefficients of $b_k = \Phi^\dagger x_1$. Step by step DMD algorithm summary is shown in Figure 2.3.

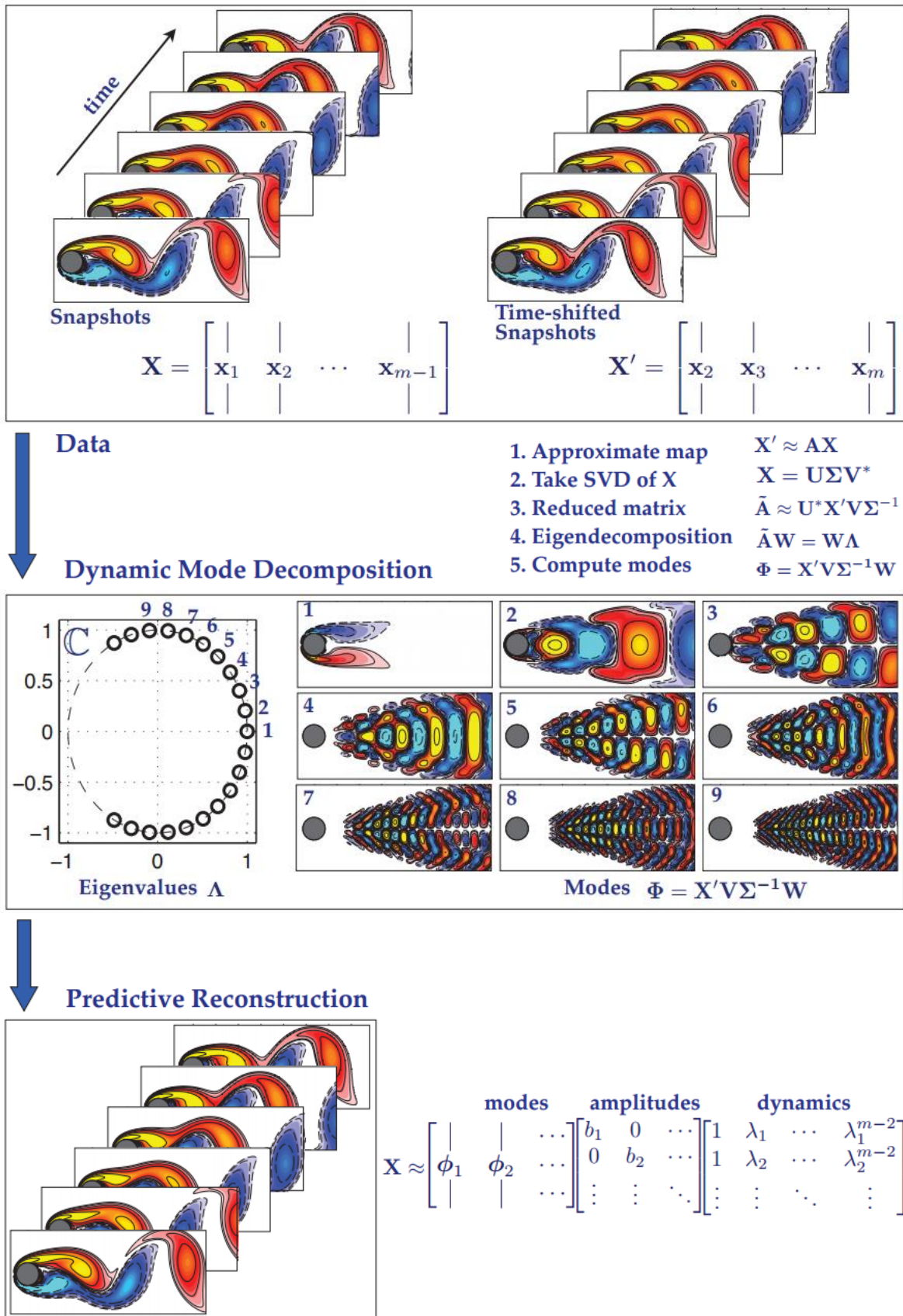


Figure 2.3 Step by step DMD algorithm summary (Kutz et al., 2016)

2.2.2 DMD example

To understand and test the DMD algorithm, two simple functions are considered as a dynamical system, which are added together. The objective is to demonstrate the ability of DMD algorithm to decompose the summation of the signals into the constituent signals.

The two functions of interests are

$$f_1(x, t) = \sin(x) \exp(i10.3t) \quad (2.25)$$

$$f_2(x, t) = \cos(x) \exp(i1.8t) \quad (2.26)$$

The mixed signal function is equal:

$$F(x, t) = f_1(x, t) + f_2(x, t) = \sin(x) \exp(i10.3t) + \cos(x) \exp(i1.8t) \quad (2.27)$$

Each spatiotemporal function is illustrated in Figure 2.4. The frequency of f_1 (freq.1=10.3) is higher than f_2 (freq.2=1.8), and this is clearly seen in the illustration. For this example, $x \in [0 \ 10]$ and $t \in [0 \ 4\pi]$ are divided into 128 equally spaced distances.

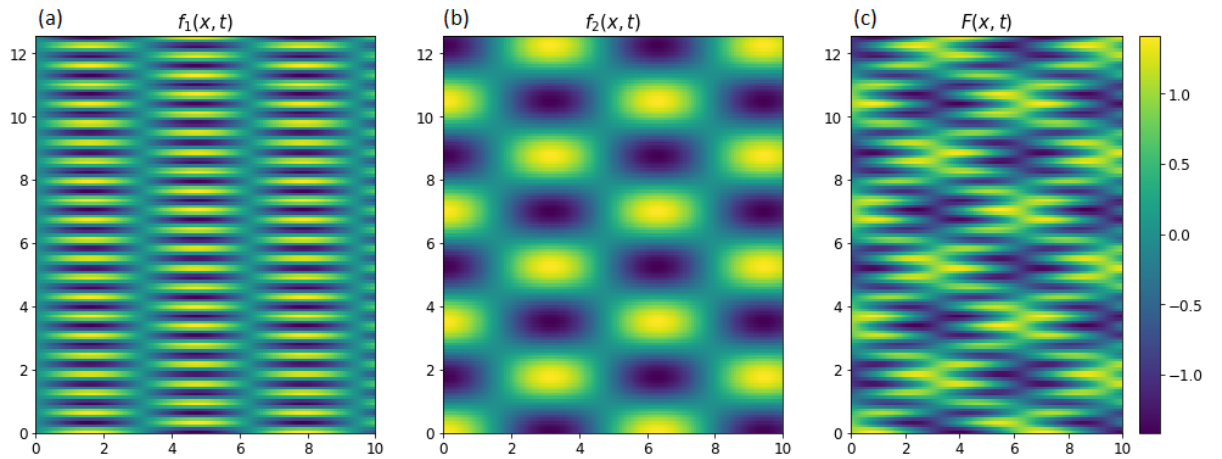


Figure 2.4 Spatiotemporal dynamics of f_1 , f_2 and F

After preparing data matrix of $F(x,t)$, first step is to compute the SVD, which transforms data matrix into a number of constitutive components, all of which has a specific meaning. In other words, the SVD produces the diagonal matrix of ordered singular values along with the two unitary matrices U and V . The SVD is represented in Figure 2.5.

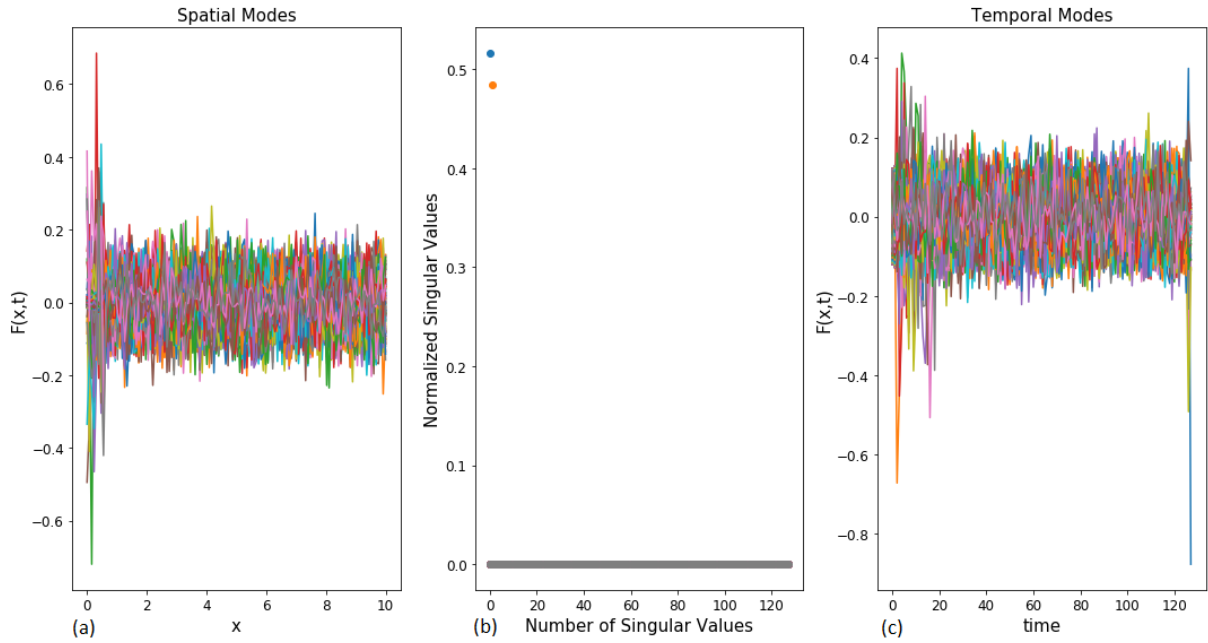


Figure 2.5 The singular value decomposition (SVD) of the dataset

The Figure 2.5a shows all of the spatial modes in the dataset which are the columns of U , left singular vector. The time dynamics of the modes in U are shown in Figure 2.5c which are columns of the matrix V , right singular vector. The normalized singular values of the modes are illustrated in Figure 2.5b which represent the energy of each mode in whole data matrix. As mentioned in Appendix B, for faster computation, it is possible to reduce the data matrix either by the economy SVD, removing zero singular values rows, or the truncated SVD by keeping the leading r singular values and vectors and discarding the rest. The entire spectrum of the singular values of $F(x,t)$ is plotted in Figure 2.6.

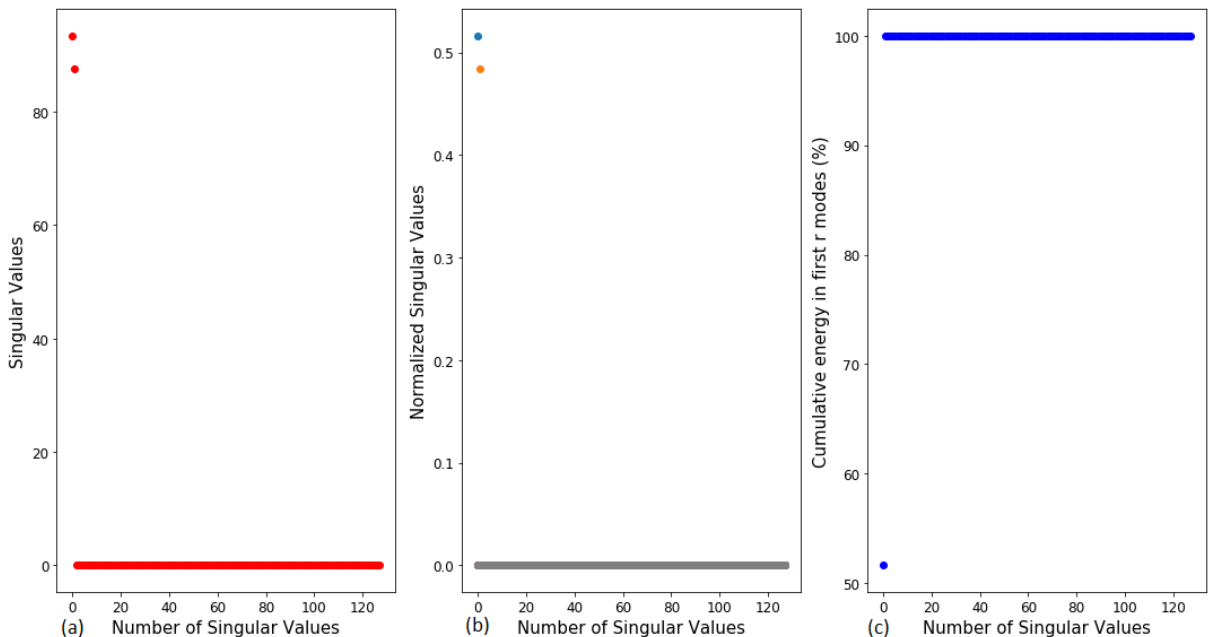


Figure 2.6 *The spectrum of singular values of $F(x,t)$*

As is shown in Figure 2.6a, most of singular values are almost zero except two, which are 93.41 and 87.53 corresponding first and second mode in U . These two singular values contain 0.51 and 0.48 of the normalized energy of the system according to Figure 2.6b, which means that each mode captures 51% and 48% of dynamics of the system, respectively. The cumulative energy amount of these two modes, Figure 2.6c, capture almost 100% of the energy and dynamics of the system $F(x,t)$ could be approximated using these two dominant modes. Thus, the rest of the modes are unnecessary.

SVD explored that the system could be characterize by two modes according to the energy of the singular values. Now the reduced version of the DMD algorithm is applied and these two modes are achieved. The next step is to know about the temporal behavior of these modes. Stability of the modes is one of the important characteristics, which influences the applicability of this method and could be checked thanks to the eigenvalues and position of them on the unit circle. The unit circle and eigenvalues for the detected DMD modes of the dynamical system $F(x,t)$ is illustrated in Figure 2.7.

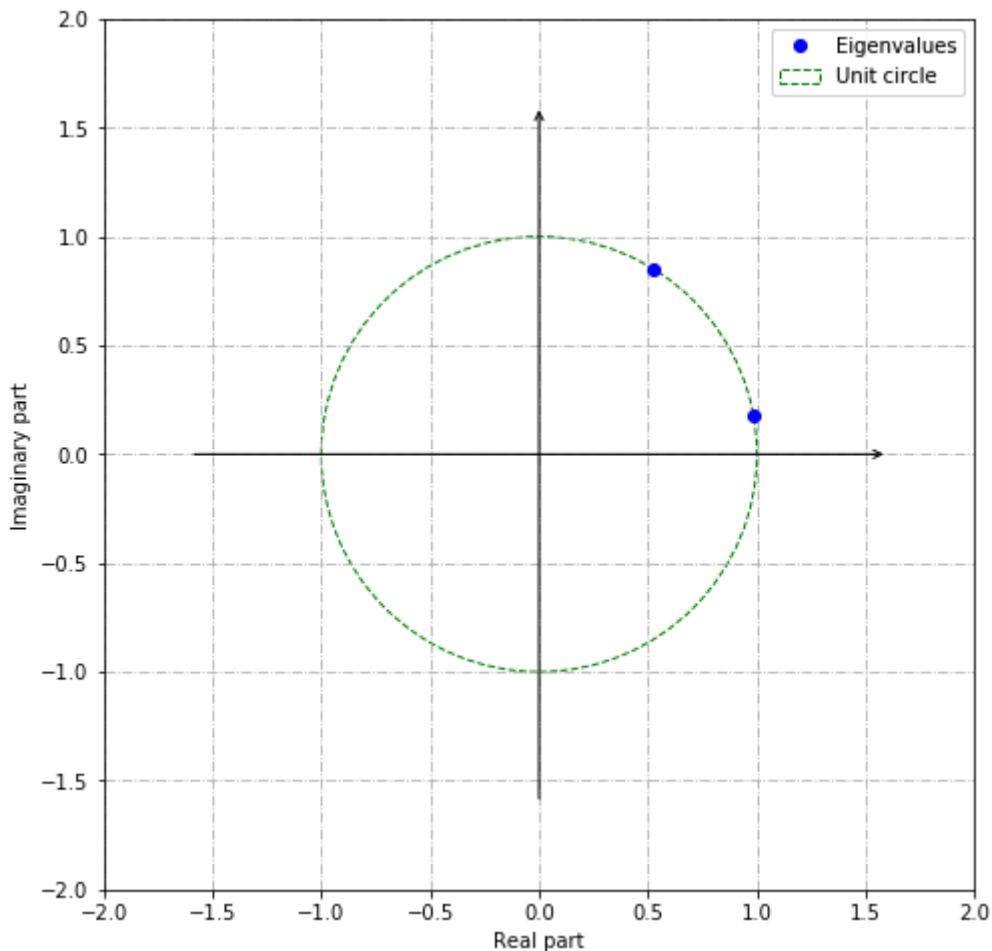


Figure 2.7 Eigenvalues of the dynamic system on the unit circle

To understand the behavior of modes using unit circle, several points should be considered:

- If eigenvalue is a complex number, the corresponding mode will oscillate. The higher imaginary part results in the higher frequency.
- If the real part of the eigenvalue is positive, the corresponding mode will diverge in time and if the real part is negative, the corresponding mode will converge in time.
- When eigenvalue is close to the origin, the time takes the mode converges or diverges will increase and the resulting mode is called slow mode. As the eigenvalue move away from the unit circle, the corresponding mode will converge or diverge very fast and the mode based on this behavior is called fast mode (Demo et al., 2018).

Considering mentioned points, a signal is stable when it does not diverge or converge too fast or too slow and this is obtained when an eigenvalue is close or on the unit circle. As shown in Figure 2.7, the DMD eigenvalues are located on the top right side of the coordinate system. The expected dynamics of modes should diverge and oscillate in time as their eigenvalues have positive real part and non-zero imaginary part respectively. The one with higher imaginary value oscillates more with higher frequency. The spatial DMD modes and their dynamics are shown in Figure 2.8.

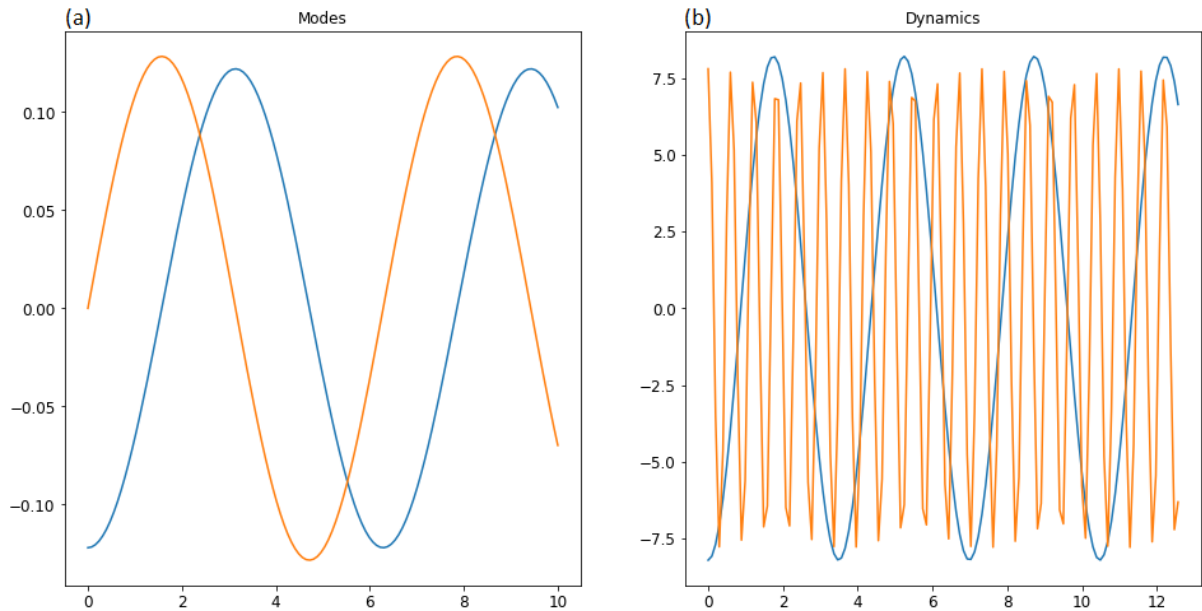


Figure 2.8 The Spatial modes and their dynamics in time

Figure 2.8a shows the DMD detected modes which catch the 100 % energy of the system. Figure 2.8b clearly supports what was mentioned about the expected dynamics based on the

eigenvalues and the unit circle. Final approval of these DMD modes and dynamics could be achieved by comparing them with the constituent signals f_1 (2.25) and f_2 (2.26), which are two diverging trigonometric functions with $\frac{\pi}{2}$ phase difference and different frequencies.

Finally, the original dataset is reconstructed using the dominant modes and their dynamics. The dominant DMD modes and DMD reconstructed dataset are shown in Figure 2.9.

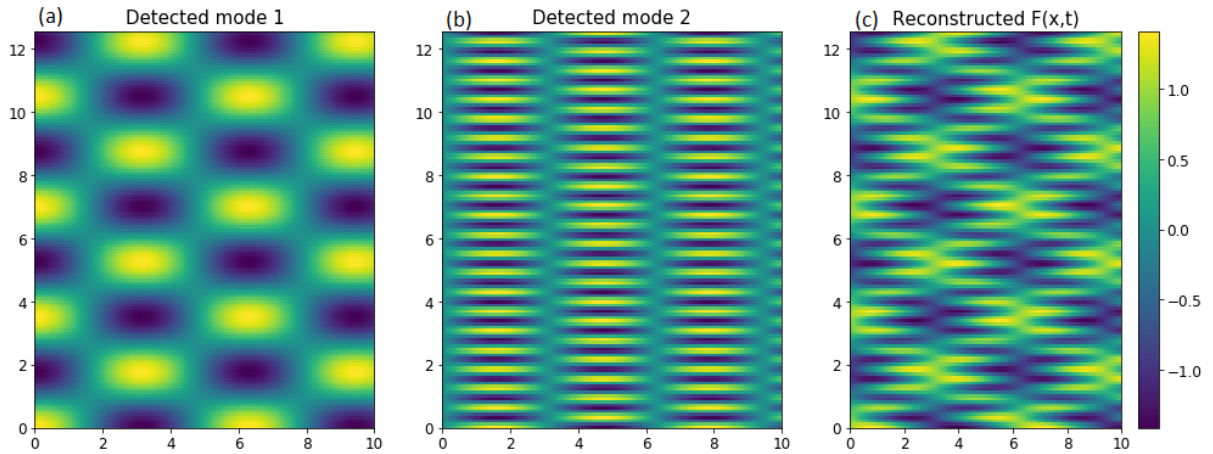


Figure 2.9 *The dominant DMD modes and DMD reconstructed dataset*

By comparing Figure 2.9 and Figure 2.4, the similarities between the evolution of the original dataset and the DMD reconstructed data are identified. To understand the precision of the DMD algorithm, the absolute error between the approximated DMD data and the original data is plotted in Figure 2.10 and Figure 2.11.

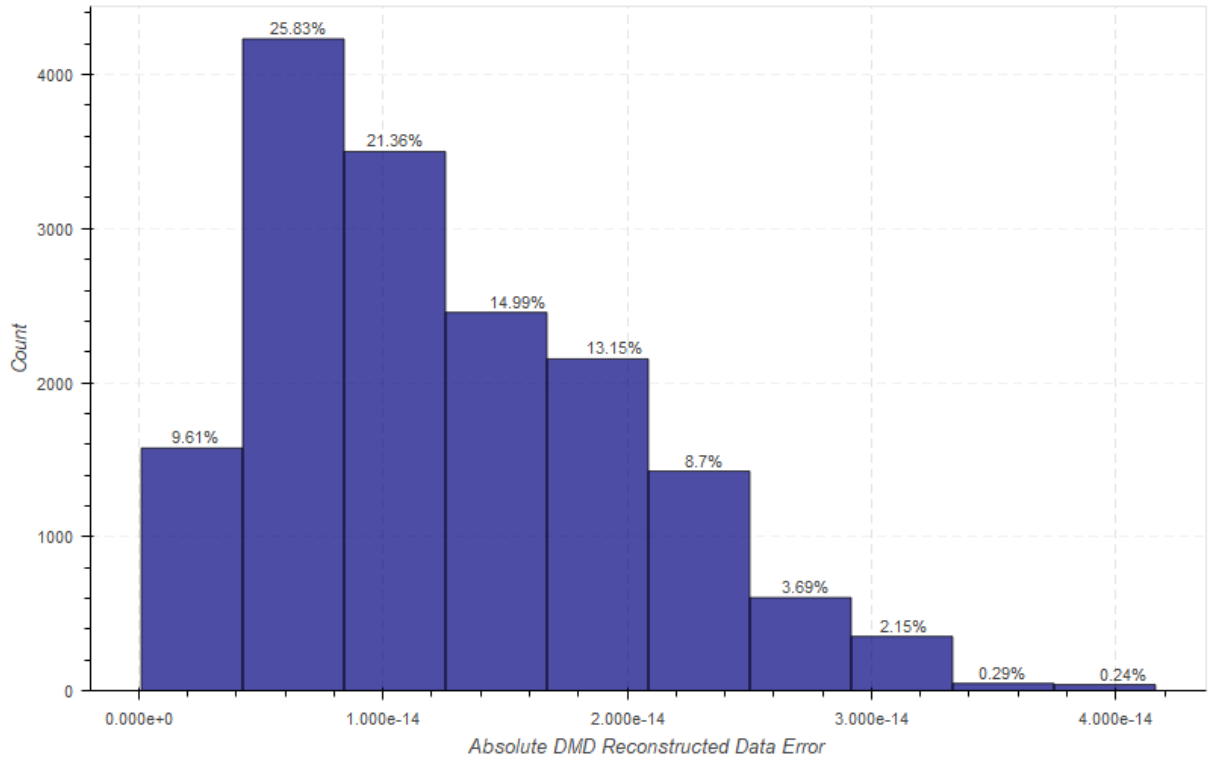


Figure 2.10 The frequency distribution of the absolute DMD reconstructed data error

As it is shown in Figure 2.10, the maximum error for the DMD reconstruction data is $4.000e-14$, which is negligible.

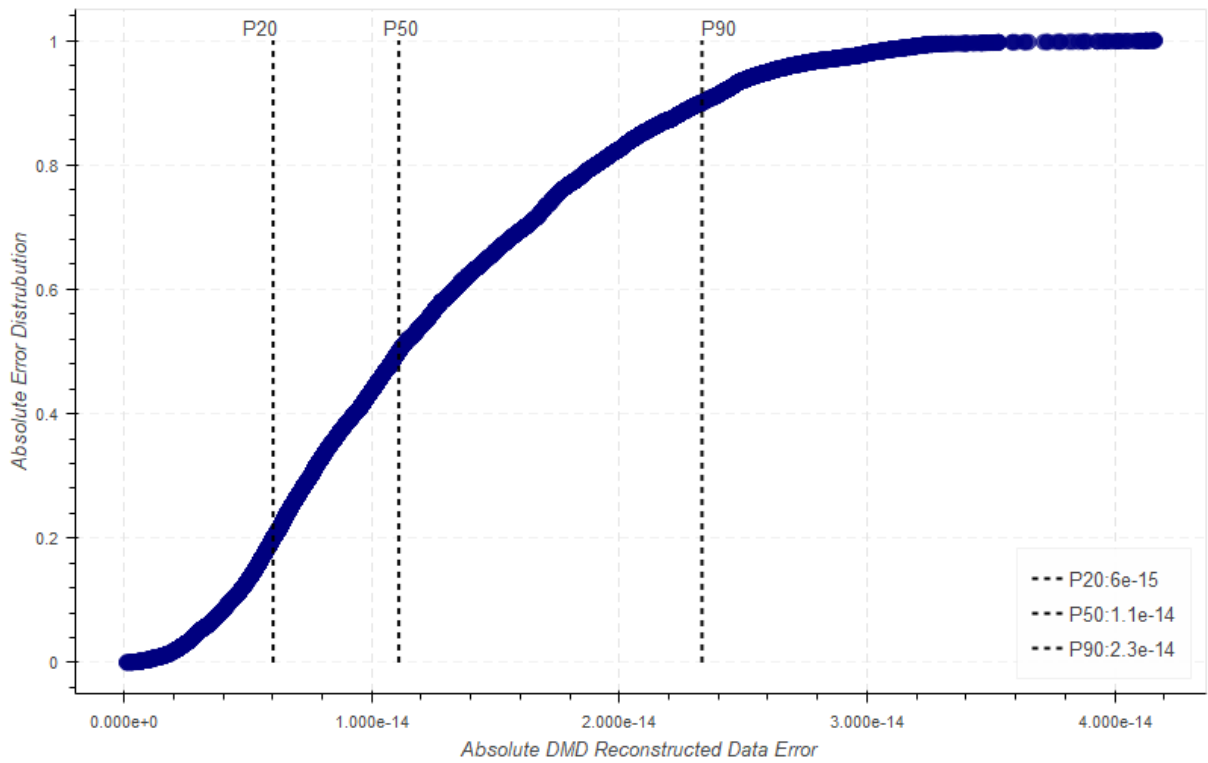


Figure 2.11 The distribution of the absolute DMD reconstructed data error

In Figure 2.11, the values for P20, P50 and P90 are presented as 6e-15, 1.1e-14 and 2.3e-14, respectively. P90 shows that 90 percent of the absolute DMD reconstructed data error are less than 2.3e-14.

The overall success of the DMD algorithm is highly dependent upon what applications one is attempting to achieve. In some applications, it may be reasonable to use DMD as a diagnostic tool and using it for other applications is limited. For the analyzed DMD example, the basic DMD algorithm computed several diagnostic features of the ideal dynamical system $F(x,t)$ such as DMD modes, eigenvalues and the reconstruction of the original data was almost perfect and matched the original dataset. In reality, dynamical systems are not behaving like an ideal system, and as a result applying the DMD is challenging. This makes various powerful extensions to DMD technique based on the different dynamical systems and unsatisfactory performances of basic DMD for those dynamical systems, thanks to the simple and well-established formulation of the DMD algorithm in linear algebra (Kutz et al., 2016).

PyDMD is a python package which is used for implementing DMD algorithm in this master thesis. In PyDMD, the majority of the DMD extensions are coded for different applications such as multiresolution DMD, compressed DMD, forward backward DMD, higher order DMD and so forth (Demo et al., 2018). Any extension used with DMD in this master thesis is explained in the following.

2.3 Optimal amplitudes of DMD modes

This extension is aimed to optimize the quality of approximation and the number of modes that are used to approximate dataset which are achieved by regularization of the least square deviation between the data matrix and linear combination of the DMD modes.

In section 2.2.1 after implementing step by step the DMD algorithm, approximation solution $x(t)$ (2.24) is achieved.

$$X \approx \underbrace{\begin{bmatrix} | & | & \cdots \\ \Phi_1 & \Phi_2 & \cdots \\ | & | & \cdots \end{bmatrix}}_{\Phi} \underbrace{\begin{bmatrix} b_1 & 0 & \cdots \\ 0 & b_2 & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}}_b \underbrace{\begin{bmatrix} 1 & \lambda_1 & \cdots & \lambda_{1(m-2)} \\ 1 & \lambda_2 & \cdots & \lambda_{2(m-2)} \\ \vdots & \vdots & \ddots & \vdots \end{bmatrix}}_{V_{and}} \quad (2.28)$$

where Φ is a matrix whose columns are the DMD modes or eigenvectors Φ_k , b is a matrix with the initial amplitude of each mode b_k and Vandermonde matrix (V_{and}) which is determined by eigenvalues and provides information about the underlying temporal frequencies. The solution

of the following optimization problem is way to determine the unknown amplitudes of 2.28 which can be illustrated as vector $D_b = \text{diag}(b) = [b_1 \ \cdots \ b_r]^T$:

$$\text{minimize}_b \|X - \Phi D_b V_{and}\|_F^2 \quad (2.29)$$

Using $X = U\Sigma V^*$ and $\Phi = UW$ the problem become:

$$\text{minimize}_b J(b) = \|\Sigma V^* - W D_b V_{and}\|_F^2 \quad (2.30)$$

which is an optimization problem and can be solved by standard methods. After solving the problem, function $J(b)$ is represented as:

$$J(b) = b^* P b - q^* b - b^* q + s \quad (2.31)$$

$$P = (W^* W) \circ (\overline{V_{and} V_{and}^*}), \quad q = \overline{\text{diag}(V_{and} V_{and}^* \Sigma^* W)}, \quad s = \text{trace}(\Sigma^* \Sigma)$$

The overline signifies the complex conjugate of a vector (matrix), and \circ is the elementwise multiplication for matrices.

The optimal DMD amplitudes, which solve optimization problem 2.29), can be achieved by minimizing function 2.31 with respect to b . If the DMD approximation solution 2.28) is properly weighted by optimal amplitudes of the modes, better result will be achieved. For detailed information, it is recommended to read the original paper (Jovanović, Schmid, & Nichols, 2014).

In chapter 2, the basic DMD theory, definition and its classified applications were introduced. In particular, a synthetic example was presented to check the accuracy of the algorithm and become familiar with terms and conditions of it.

The following chapter focuses on implementing DMD in practice. The DMD algorithm will be applied to a case study in dynamics of drilling parameters. The accuracy of different DMD applications such as the original data reconstruction, interpolation, prediction will be analyzed.

3 Case Studies, Results and Discussions

We have drilled many wells without the benefit of downhole real-time data and this is no longer a necessary limitation since the introduction of wired drill pipe (WDP) technology. The WDP system has been commercialized since 2006 which creates a high-speed data network from the drill-string enabling bi-directional data transmission at speeds up to 57000 bits per second. The WDP allows for different data sources such as Logging while drilling (LWD), measurement while drilling (MWD) and along-string measurement (ASM) tools, to stream to surface in real time and makes real time decisions-making possible (see Figure 3.1) (Foster & Macmillan, 2018; Schils, Teelken, van Burkleo, Rossa, & Edwards, 2016).

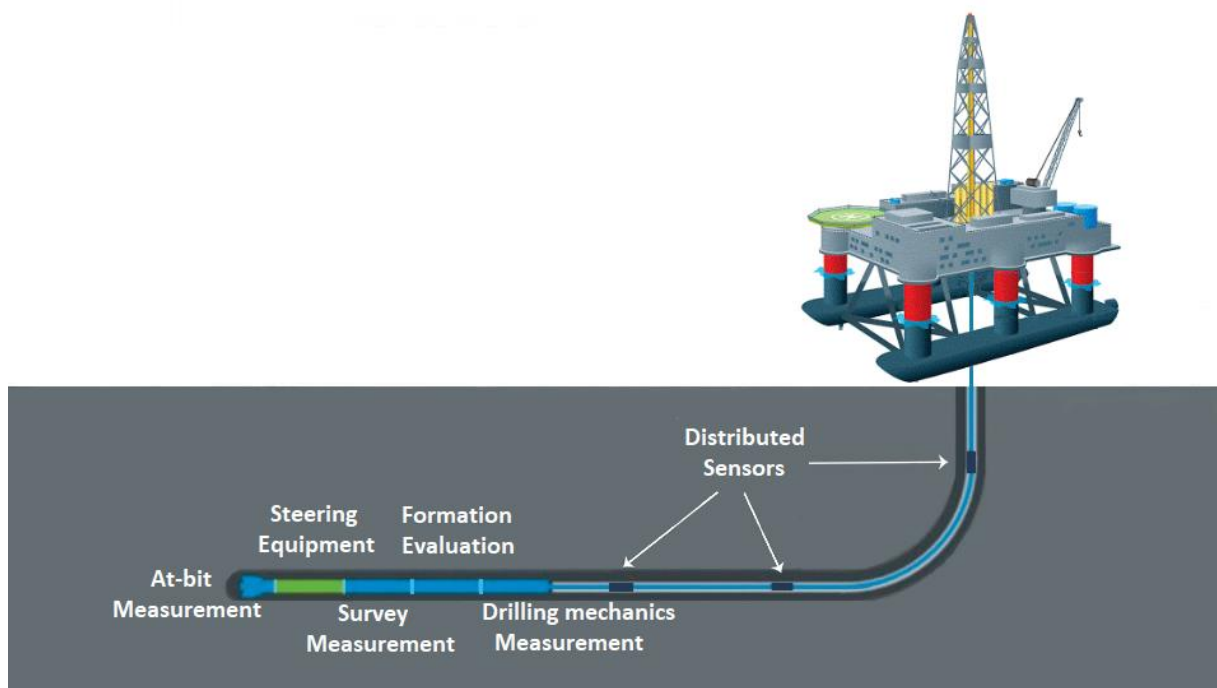


Figure 3.1 Downhole real time data sources (Israel et al., 2018)

Since the introduction of high-speed telemetry to the drilling market, operators have recognized the potential benefits of WDP on their drilling processes in many different ways which push the technology forward (Foster & Macmillan, 2018).

Along string measurement (ASM) sensors which can be placed anywhere along the WDP, collect dynamic data from downhole and transmit it to the surface through WDP communication network for analysis and modeling to improve efficiency and avoid problems related to downhole conditions. ASM tools which run on WDP are limited to mechanical measurements including, pressure, temperature, vibration and rotational speed. There have been continuous attempts by service companies to improve or add ASMs to measure different

parameters such as bending, torque, tension and borehole caliper (NOV, 2020). Also, it should be noted that despite the high bandwidth of the wired pipe telemetry network, measurements from the ASMs are sent in packets. The refresh interval of packets is about 2.5s and only average, min/max and standard deviations corresponding to the time window are transferred (Eric, Per, Lars Jørgen, Håvard, & Espen, 2019). In practice, there are from 0 to 4-5 ASMs along the drill-string.

In this chapter, DMD algorithm will be implemented on a drilling dynamic dataset received from NORCE’s simulator, which generates synthetic measurements along the drill-string at every tool-joints and transfer them without delays and at a refresh rate of 10Hz. This would require a bandwidth largely greater than the one of currently commercialized WDP solutions but which would be compatible with newer solutions such as the one described by Temizel et al., 2019 . A brief information about the simulated well configuration and other parameters are presented in the following before going into the main analysis.

3.1 Simulated well

For a short overview of the simulated well, some of the input data and simulator visualizations are presented. Figure 3.2 shows different information about the depth of blowout preventer (BOP) and surface sensor, size and depth of casings, riser and open-hole section.

The screenshot displays a software interface for well architecture simulation, organized into several sections:

- Riser:** Includes input fields for Body OD (21 in), Body ID (19 1/4 in), Length (153,000 m), and a checkbox for "Has Riser Telescopic Joint".
- Blowout Preventer:** Includes input fields for Top BOP depth (153,00 m), Length, Max. pressure rating, and Max. thru OD.
- Sensors:** Features an "Add" button, a "Delete" button, a "Sensor Name" field (AnnulusSurfaceSen), and a "Sensor depth (MD)" field (1,00 m).
- Casing Architecture:** A table with columns for Susp. Depth (m), Shoe Depth (m), OD (in), and ID (in). The first row is highlighted in blue.
- Open hole:** Includes input fields for Open hole length (2 200,000 m) and Open hole diameter (8 1/2 in).

Susp. Depth (m)	Shoe Depth (m)	OD (in)	ID (in)
153,00	602,00	24	21 1/2
569,00	1 017,00	17	16 1/8
153,00	1 545,00	13 5/8	12 3/8
1 499,00	4 179,00	10 3/4	9 9/16

Figure 3.2 Simulated well architecture

As can be seen from wellbore trajectory in Figure 3.3, the simulated well is horizontal. The corresponding TVD for vertical, build-up and horizontal sections are shown in left curve. The orientation of well is also illustrated in the azimuth curve on the right side.

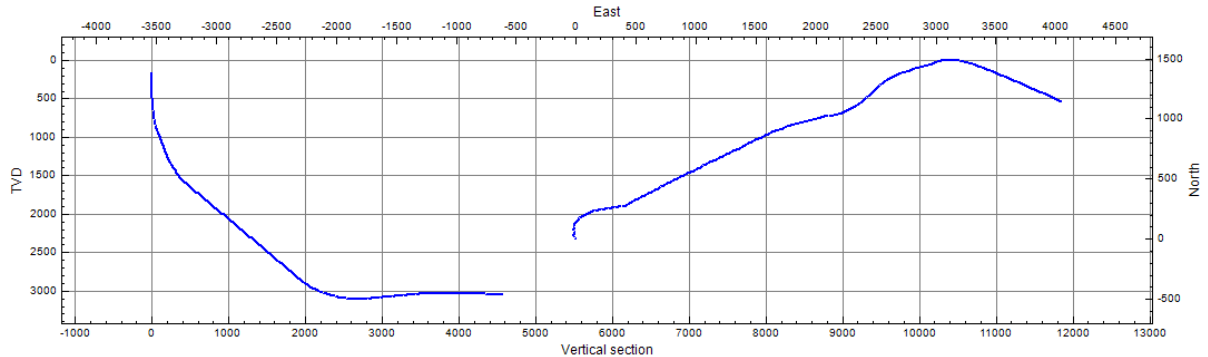


Figure 3.3 wellbore trajectory

Schematic and description of the drill-string components (drill pipes and bottom hole assembly (BHA)) are shown in Figure 3.4.

Drill-string/BHA description			#	Length (ft)	OD (in)	ID (in)	Lin. Weight (lb/ft)	Cum. Length (ft)	Cum. Mass (lb)
Type									
Bit	▼		1	0,98	8 1/2	3	274,4	0,98	268,96
Steerable rotary t...	▼		1	20,16	7 5/8	1 1/2	109,5	21,14	2 476,48
MWD	▼		1	7,17	7 1/8	2 1/32	109,5	28,31	3 261,59
Drill-collar	▼		1	9,20	6 3/4	3 1/8	98,0	37,51	4 163,19
Stabilizer	▼		1	2,20	6 3/4	3	109,5	39,71	4 404,09
LWD	▼		1	18,98	6 3/4	1 29/32	109,5	58,69	6 482,40
Stabilizer	▼		1	2,20	6 3/4	3	109,5	60,89	6 723,30
LWD	▼		1	24,16	6 3/4	1 29/32	109,5	85,05	9 368,82
Stabilizer	▼		1	2,20	6 3/4	3	109,5	87,25	9 609,72
LWD	▼		1	18,98	6 3/4	1 29/32	109,5	106,23	11 688,03
Stabilizer	▼		1	3,26	6 3/4	3 1/8	109,5	109,49	12 045,00
LWD	▼		1	10,69	6 3/4	1 29/32	109,5	120,18	13 215,56
Stabilizer	▼		1	1,87	6 3/4	3	109,5	122,05	13 420,32
LWD	▼		1	11,82	6 3/4	1 29/32	109,5	133,87	14 714,61
LWD	▼		1	18,87	6 3/4	1 29/32	109,5	152,74	16 780,87
Stabilizer	▼		1	2,20	6 3/4	1 29/32	109,5	154,94	17 021,77
Drill-collar	▼		1	1,87	6 3/4	3	98,0	156,81	17 205,03
MWD	▼		1	27,77	6 29/32	3 1/4	109,5	184,58	20 245,85
LWD	▼		1	25,57	6 3/4	1 29/32	109,5	210,15	23 045,76
Stabilizer	▼		1	2,20	6 3/4	1 29/32	109,5	212,35	23 286,66
Drill-collar	▼		1	1,87	6 3/4	3	98,0	214,22	23 469,92
LWD	▼		1	4,54	6 3/4	1 29/32	109,5	218,76	23 967,05
LWD	▼		1	4,97	6 3/4	1 29/32	109,5	223,73	24 511,27
Stabilizer	▼		1	5,60	6 3/4	3	19,6	229,33	24 620,77
Float-sub	▼		1	3,00	6 3/4	3	109,5	232,33	24 949,27
Drill-collar	▼		1	82,02	6 3/4	3	109,5	314,35	33 930,46
HWDP	▼		1	475,72	5	3	53,1	787,07	58 855,91
Drill-pipe	▼		1	6 699,48	5	4 9/32	23,5	7 486,55	215 970,11
Drill-pipe	▼		1	13 123,36	5 7/8	5 1/32	30,7	20 609,91	618 975,12

Figure 3.4 Description of Drill-string components

Figure 3.5 shows some of the properties of drilling fluid used for this study.

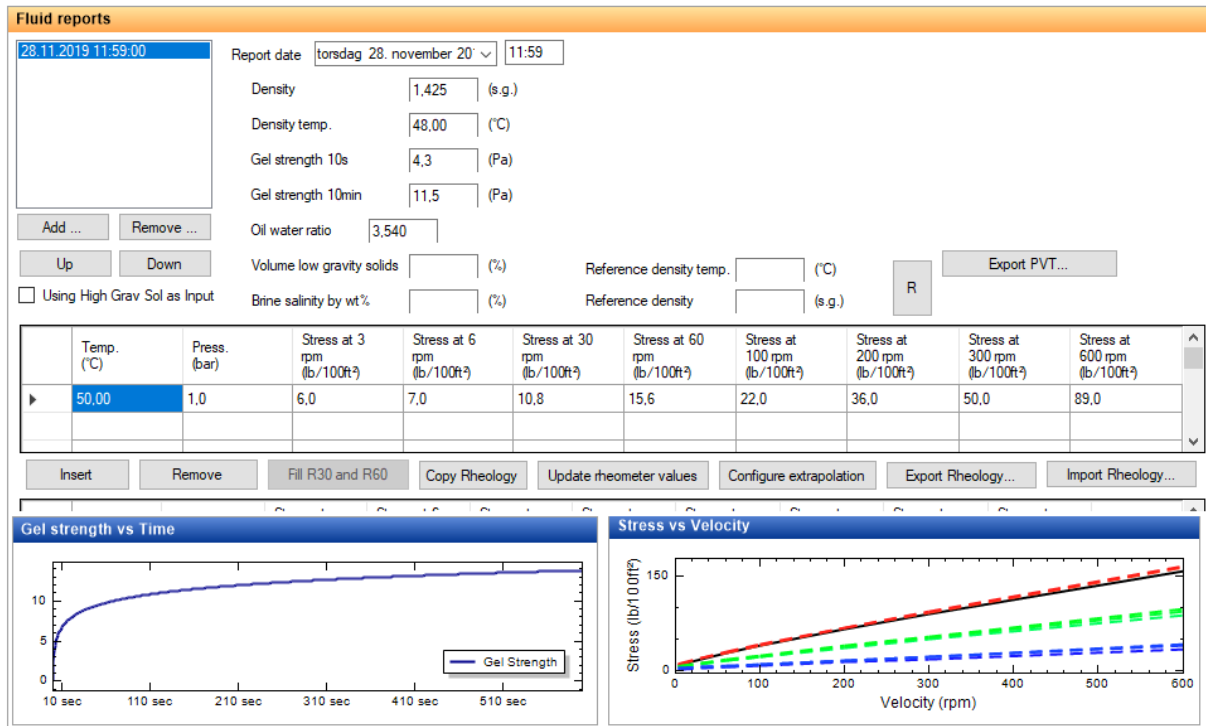


Figure 3.5 Drilling fluid report

3.2 Case study: Annular fluid velocity (AFV) during trip out

During drilling operation, it is often necessary to pull the drill-string out of the wellbore and run it back (tripping operation) for different reasons. The typical reasons for tripping pipe are replacing worn-out bit, replacing damaged downhole equipment, performing logging operation and so forth. When tripping, the drill-string acts like a spring-mass system, composed of a multitudes of individual masses attached to each other's by springs, which oscillates axially with complex patterns as a consequence of mechanical friction between the moving string and wellbore (Cayeux, 2018) and swab and surge engendered by fluid movement (Cayeux et al., 2020). In other words, the drill-string axial movement is most often not periodic, and oscillates with larger or shorter periods and also nonuniform timing of process in different time intervals (Sethares, 2001). This oscillatory behavior of the drill-string disturbs the fluid velocity profile in an annulus. Because of the no slip at the wall condition, the fluid in contact with the drill-string moves at the same velocity as the drill-string, while the fluid velocity at the wellbore is zero. This results in a fluid velocity field in the annulus where in the same cross-section the fluid velocity can be positive at a certain radial distance and negative at another radial position. This is called the clinging effect (Whittaker, 1985).

The simulated annular fluid velocity while lifting up one stand is used to test the DMD algorithm. The dataset contains 388 rows and 1669 columns, which corresponds to the number of sensors placed along the drill-string, and number of measurements in different time snapshots, respectively. It should be noted that there is no ASM to measure annular fluid velocity directly and it is calculated using differential pressure measured between sensors, distances between sensors, fluid characteristics, wellbore geometry. Figure 3.6 visualizes the annular fluid velocity along the wellbore for every 0.2 sec time snapshots. Positive annular fluid velocity during tripping out means that fluid moving upward as a result of clinging effect and negative annular fluid velocity indicating downward movement of fluid.

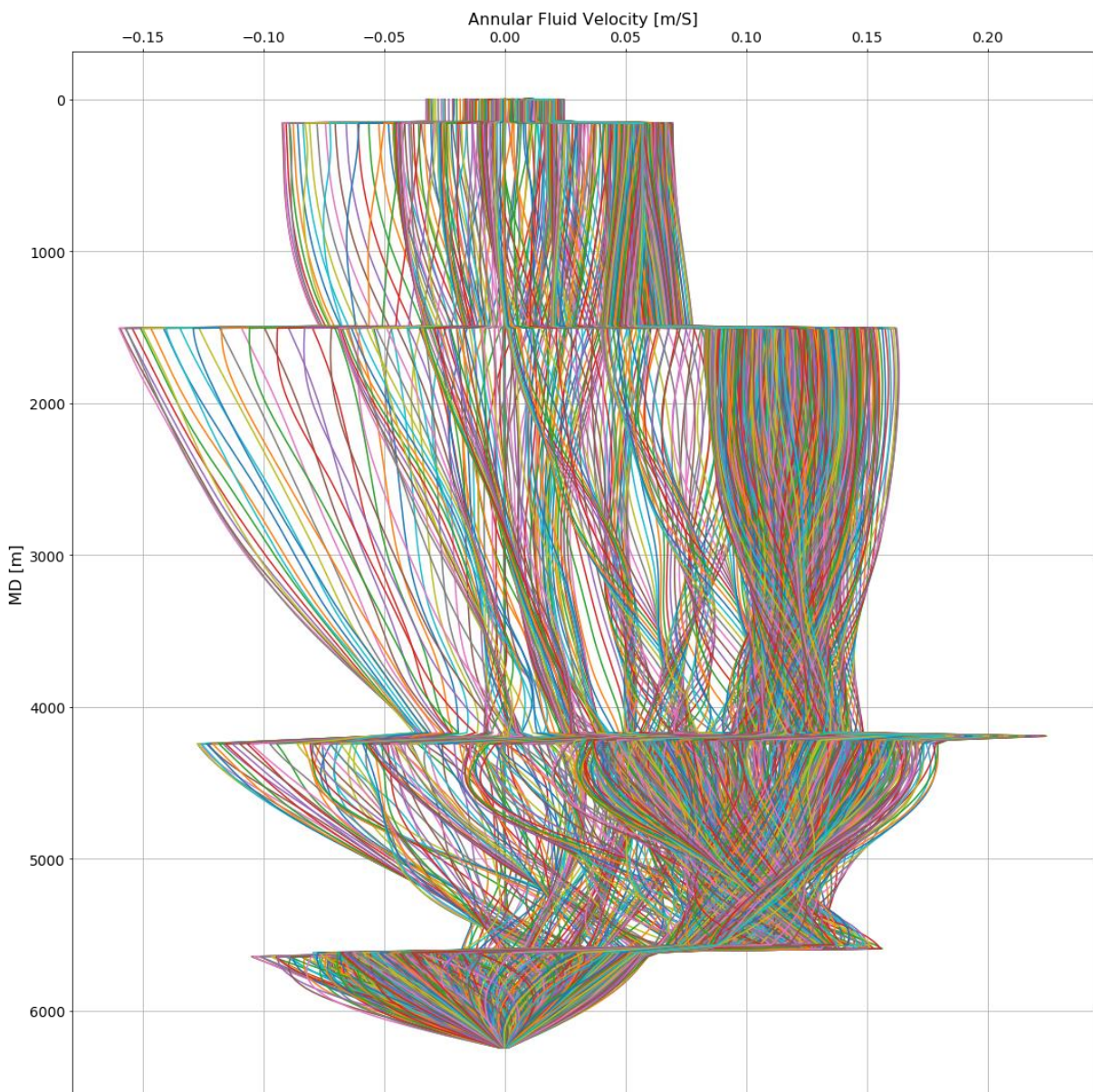


Figure 3.6 *Annular Fluid Velocity dataset*

As shown in Figure 3.6, the annular fluid velocity values change at each measured depth, which represent the dynamic behavior of the AFV. The dynamic behavior of AFV in different time intervals shows variable oscillations with larger or shorter periods and also nonuniform timing. There is a dominance of positive velocities compared to the negative ones, which can be seen by the darker part of the curve in the righthand side, showing that fluid mostly moves downward. Abrupt changes or jumps of the velocity in annuls could be caused by an abrupt change in drill-string movement, drill-string configuration, different borehole dimensions and inclination of each section. Now this data set is going to be used for DMD implementation.

3.2.1 DMD data reconstruction

In order to use DMD algorithm for any application, it is necessary to reconstruct the dynamics over the interval used for the algorithm. In chapter 2, DMD algorithm was successfully tested for an ideal dynamical system. As the annular fluid velocity data contains dynamics with different periods in nonequal time intervals, the first use of DMD is to check the reconstruction of dynamics in short intervals and then larger ones. Less than 388 time-snapshots are enough to test DMD reconstruction, as mentioned in the previous chapter, overdetermined system is critical for DMD algorithm, meaning that the number of columns should be less than the number of rows. Here, 395 time-snapshots are used.

According to the similar patterns observed in the Figure 3.6, time snapshots 0 to 394 are divided into 14 intervals with the similar dynamics, however for some intervals it is difficult to find the transition from one behavior to another (see Table 1).

Table 1. *Annular fluid velocity sub datasets*

Interval No.	Interval range
1	0 – 43
2	44 – 111
3	112 – 140
4	141 – 161
5	162 – 185
6	186 – 211
7	212-230
8	231 – 253

9	254 – 278
10	279 – 307
11	308 – 332
12	333 – 347
13	348 – 372
14	373 - 394

3.2.1.1 DMD reconstruction of single sub dataset

DMD algorithm, without reduction, is implemented on the 14 sub datasets in Table 1.

- Interval No. 1: 0 – 43

First interval shows the acceleration of velocity as tripping out is started (see Figure 3.7). As can be seen, velocity values are small and changing gently. At the beginning, fluid velocities are positive with small amplitudes and it moves upward in the annulus because of clinging effect.

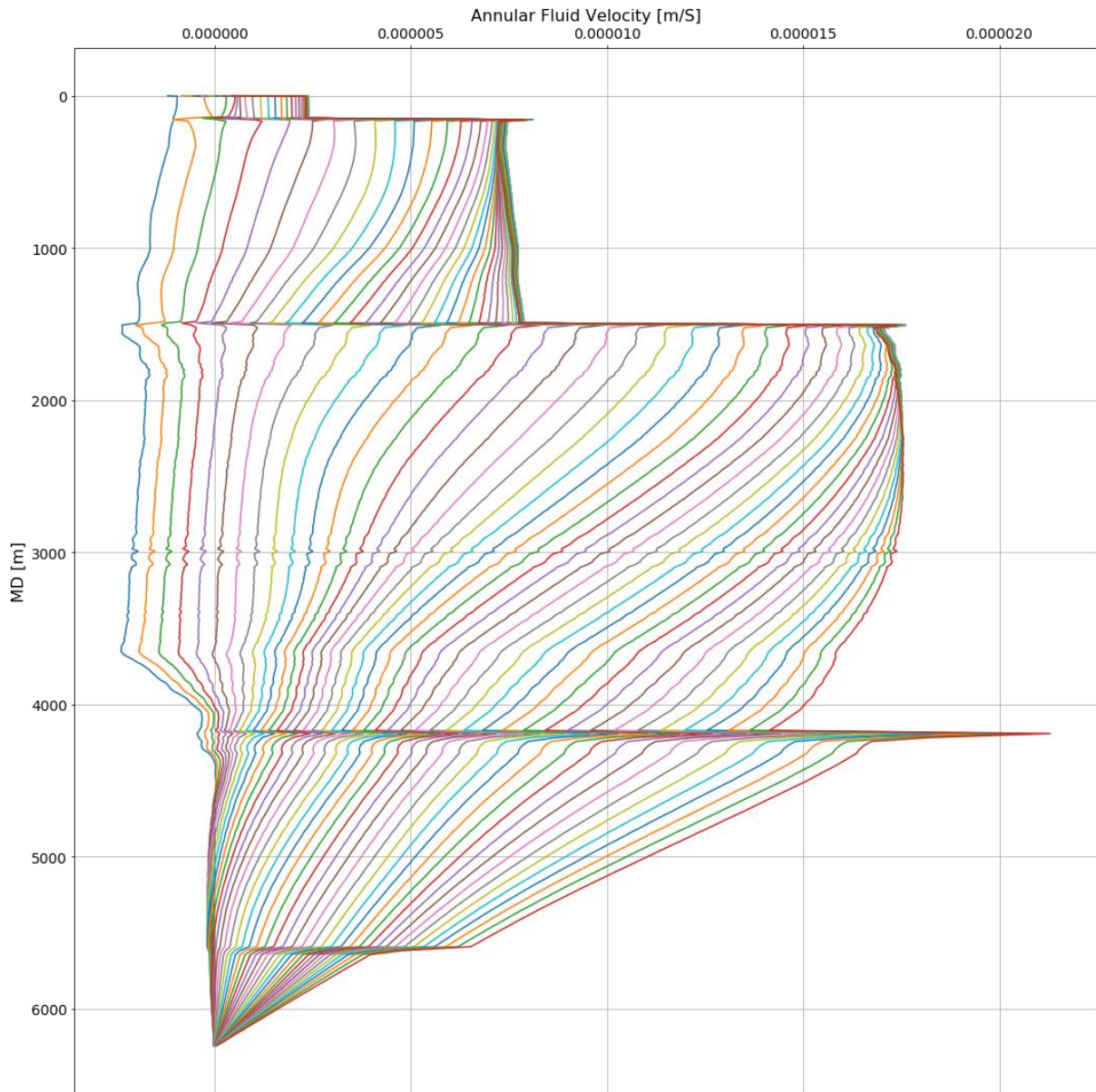


Figure 3.7 Annular fluid velocity, interval No. 1 (0 - 43)

Singular value decomposition (SVD) and Singular values of the interval No. 1 are shown in Figure 3.8 and Figure 3.9.

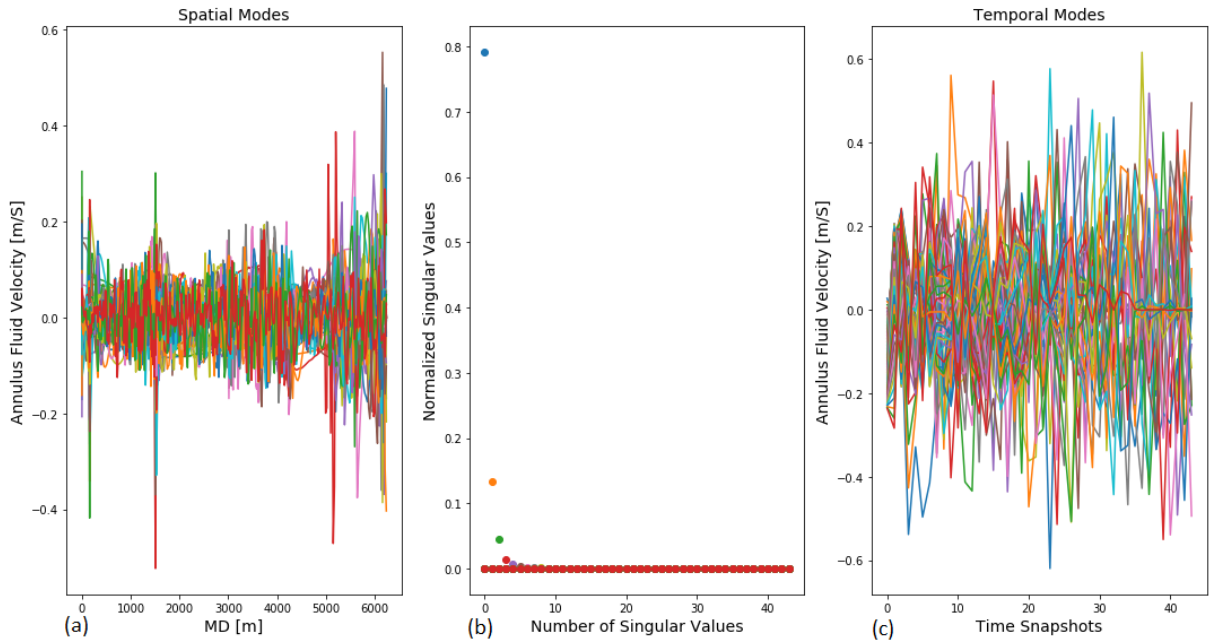


Figure 3.8 Singular value decomposition (SVD) of interval 1(0 - 43)

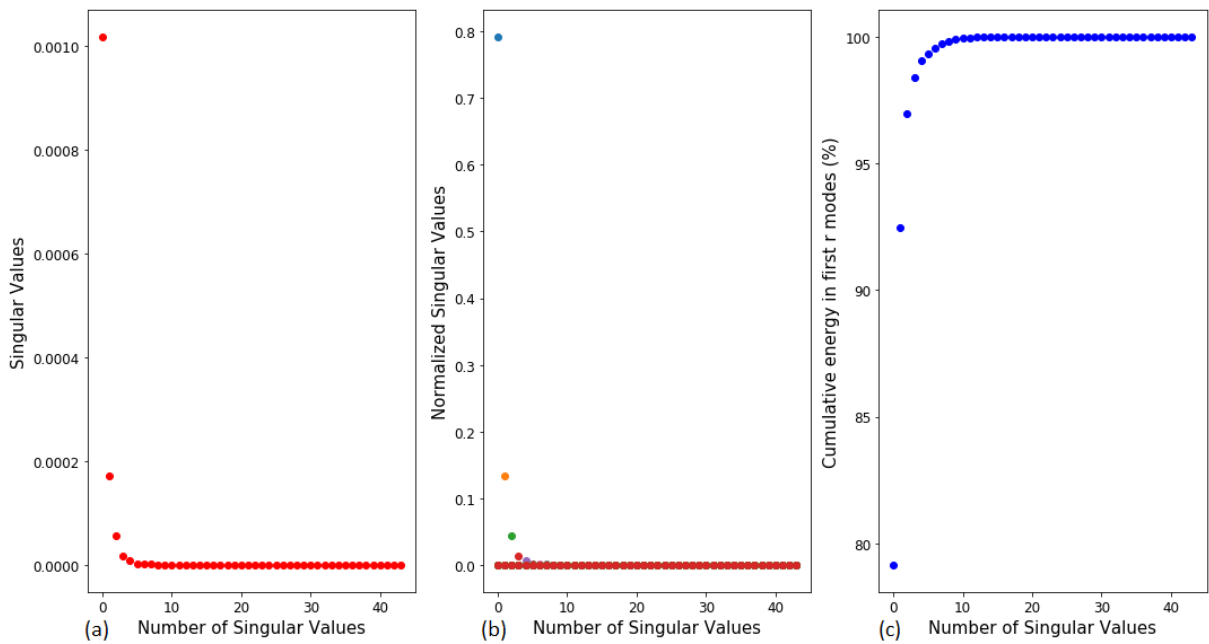


Figure 3.9 Singular values of interval No. 1 (0 - 43)

As shown in Figure 3.8b and Figure 3.9b, the maximum energy of the system is captured by the first singular value, which means, it is the dominant mode through measurements (see Figure 3.8a) for 79.13% of the reconstruction. Now the DMD algorithm is implemented to catch these modes and their dynamics and finally reconstruct the interval. Figure 3.10 shows spatial modes and their dynamics used for reconstruction.

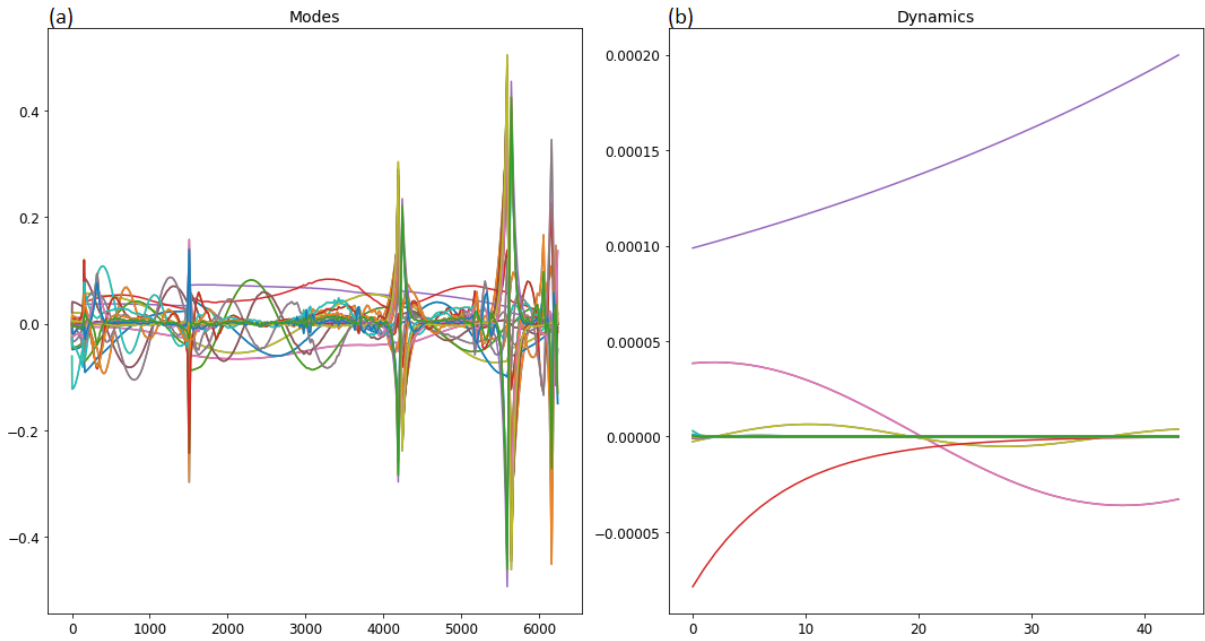


Figure 3.10 *The Spatial modes and their dynamics in time, interval 1 (0 - 43)*

As mentioned earlier, modes and their dynamics are components of the DMD approximate solution (2.24). Dynamics of modes are achieved by $\exp(\omega_k) b_k$, which ω_k and b_k denote eigenvalue and initial amplitude, respectively. These two parameters are critical for DMD performance as eigenvalues determine stability of modes and initial amplitude, which is a scale for each mode reconstruction, determines the amplitude of a mode. Eigenvalues of interval 0 – 43 are shown in Figure 3.11.

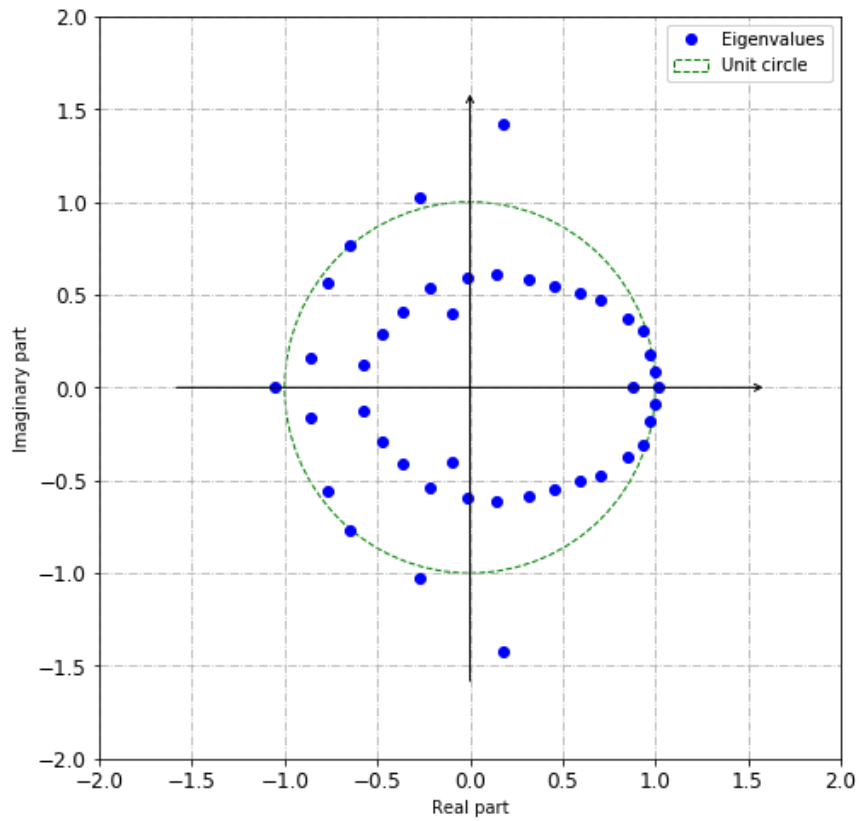


Figure 3.11 *Eigenvalues of interval 1 (0 - 43)*

The temporal modes which represent the behavior of the system in time, are dependent on the position of eigenvalues. When a dynamic change abruptly and with various periods in time, some of eigenvalues will be located outside of the unit circle and produce fast mode or slow mode based on the position of each eigenvalue and finally limit the performance of reconstruction and prediction. 3D plots of DMD reconstruction and original measurements 0 – 43 are shown in Figure 3.12 and Figure 3.13.

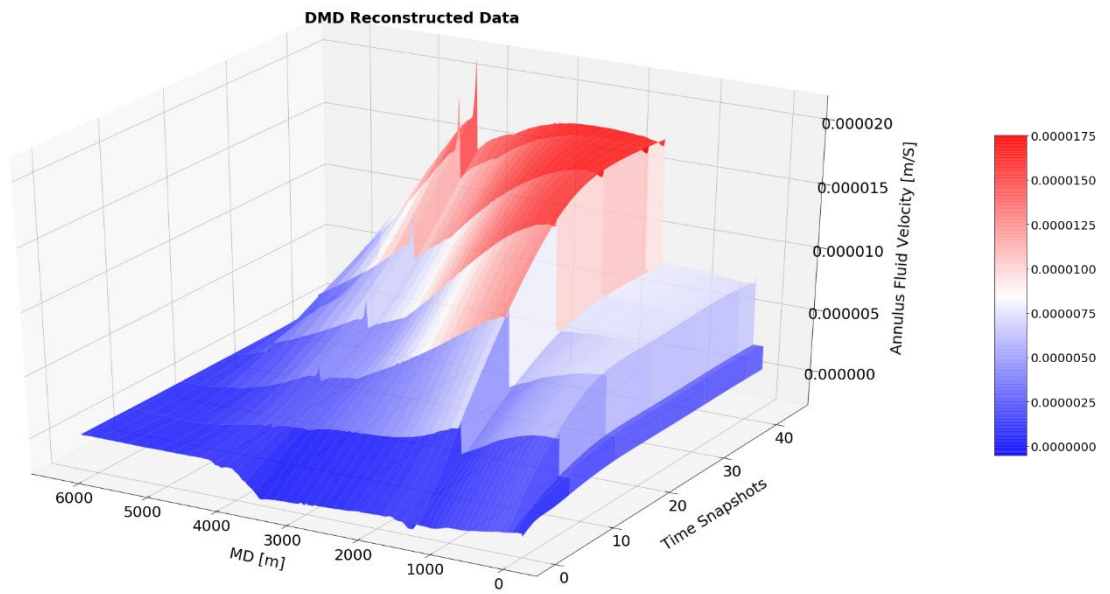


Figure 3.12 DMD reconstruction of the interval No. 1 (0 - 43)

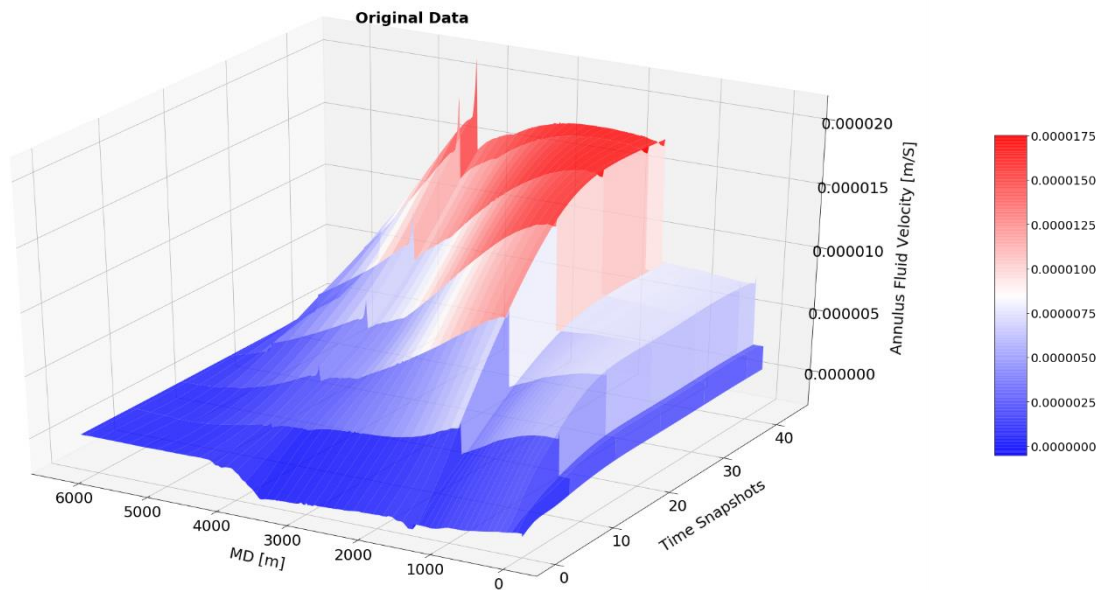


Figure 3.13 Original data interval No. 1 (0 - 43)

The reconstructed and original data plots look similar. To find out the accuracy of DMD reconstruction, it is better to have a look at the figures which represent error.

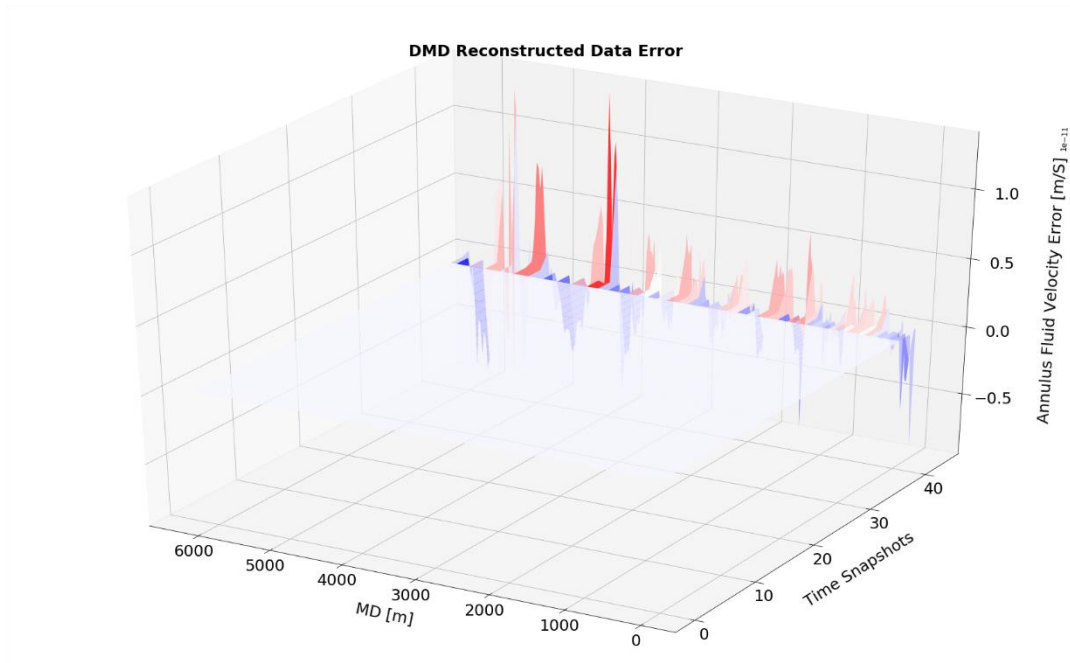


Figure 3.14 DMD reconstructed data error interval No. 1 (0 - 43)

As shown in Figure 3.14, performance of DMD reconstruction over the interval is great. In last time-snapshots, errors are still low, but compare to the rest of timesteps are higher which could be as a result of fast modes due to transition to the next dynamic.

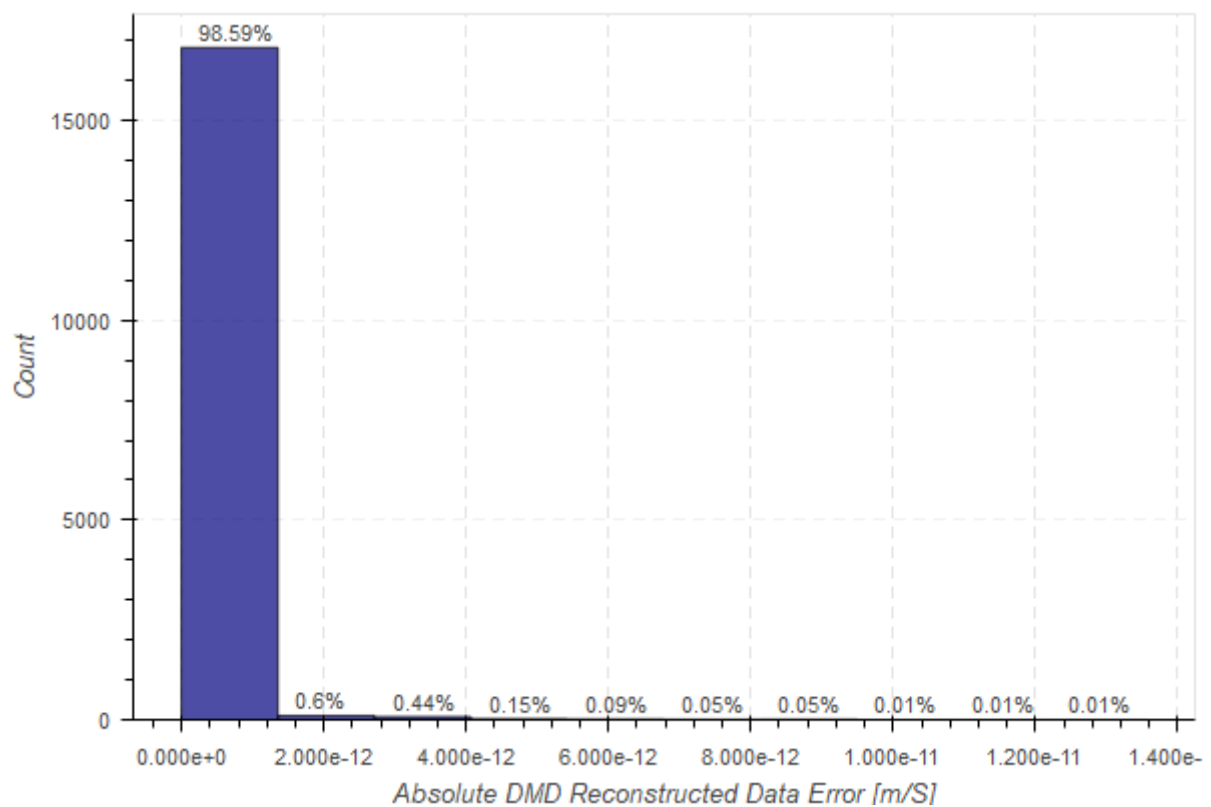


Figure 3.15 The frequency distribution of the absolute DMD reconstructed data error, interval No. 1 (0 – 43)

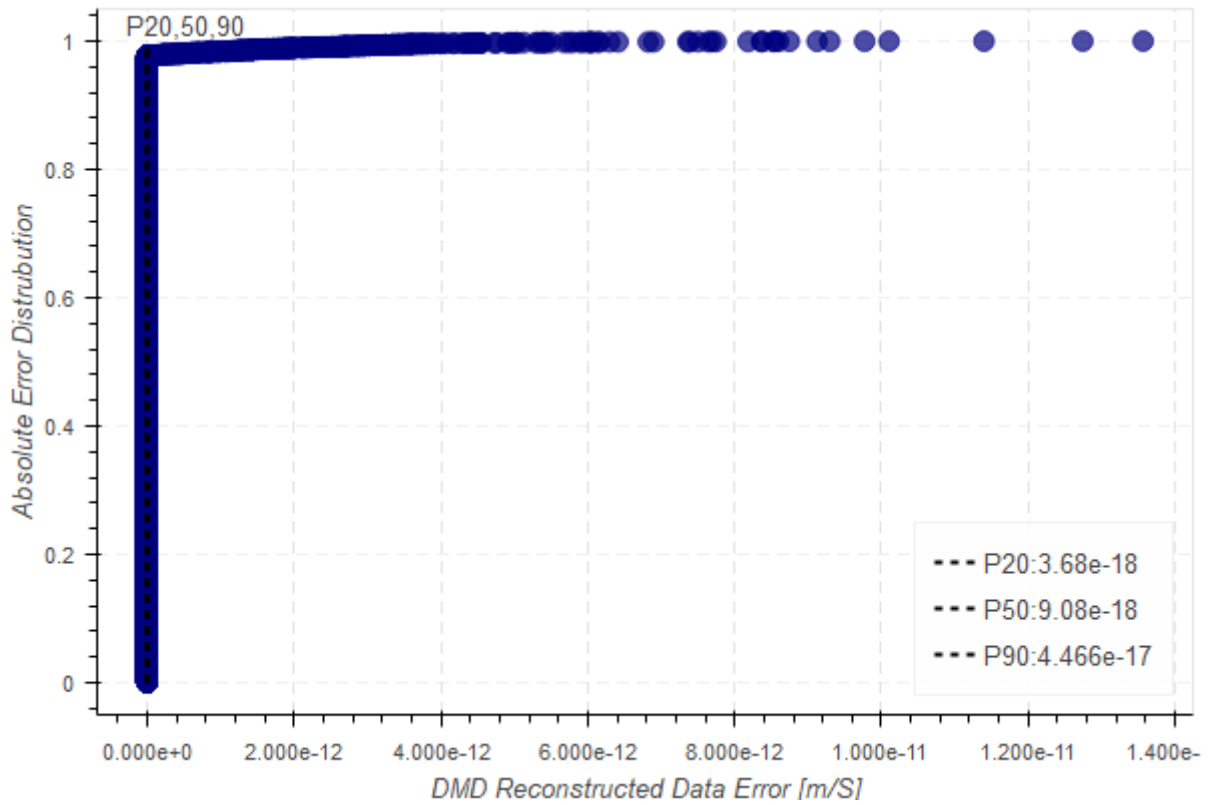


Figure 3.16 The distribution of the absolute DMD reconstructed data error, interval No. 1 (0 - 43)

As can be seen in Figure 3.15, the highest error distribution belongs to first bar (98.59%) and this is the reason for P90 to become 4.5×10^{-17} for 90 percent of the reconstruction in Figure 3.16. The analysis of DMD reconstruction for the rest of sub datasets show high accuracy and acceptable result (see Table 2). DMD reconstruction figures for interval No. 2-5 are attached in Appendix D.

Table 2. Error distribution of DMD reconstruction sub datasets

Interval No.	Interval range	P20	P50	P90
1	0 – 43	3.7×10^{-18}	9.1×10^{-18}	4.5×10^{-17}
2	44 – 112	1.2×10^{-14}	4.1×10^{-14}	1.4×10^{-13}
3	112 – 140	1.3×10^{-16}	3.5×10^{-16}	1.2×10^{-15}
4	141 – 161	4.2×10^{-17}	1.2×10^{-16}	3.6×10^{-16}
5	162 – 185	2.3×10^{-07}	1.0×10^{-06}	8.8×10^{-06}
6	186 – 211	1.0×10^{-16}	2.4×10^{-16}	5.6×10^{-16}
7	212-230	5.9×10^{-17}	1.7×10^{-16}	8.9×10^{-16}
8	231 – 253	4.7×10^{-16}	1.3×10^{-15}	3.8×10^{-15}
9	254 – 278	2.2×10^{-16}	5.6×10^{-16}	2.3×10^{-15}

10	279 – 307	2.5e-16	7.6e-16	2.3e-15
11	308 – 332	5.2e-16	1.4e-15	3.5e-15
12	333 – 347	3.9e-16	9.9e-15	2.9e-15
13	348 – 372	9.7e-17	3.1e-16	1.1e-15
14	373 - 394	4.3e-16	1.5e-115	5.6e-15

As dynamic changes more gently and with a predictable trends or no abrupt oscillation, eigenvalues corresponding to more stable modes (stable eigenvalues)⁴ will be achieved and the energy of system is captured by few modes and as a result optimal amplitude extension works fine for all reconstruction in the interval.

As mentioned earlier, measured data were classified to 14 intervals based on similar patterns. It was difficult to find out the end of dynamics, especially for transition from one dynamic to another and finally DMD algorithm managed to reconstruct all the intervals.

Recall that it is our desire to reconstruct whole dataset, no matter what dynamics exist in measurements in time. In following section, multiple intervals together are used to check the behavior of DMD algorithm and the performance of reconstruction.

3.2.1.2 DMD reconstruction of multiple sub datasets

In this section, some of AFV sub datasets in Table 1 are merged together to evaluate DMD reconstruction capabilities in the following.

- Merged interval No. 1: 0 – 111

This interval contains two dynamics 0 – 43 and 44 – 111 according to Table 1.

⁴ For simplicity, eigenvalues corresponding to stable modes are considered stable eigenvalues in the following

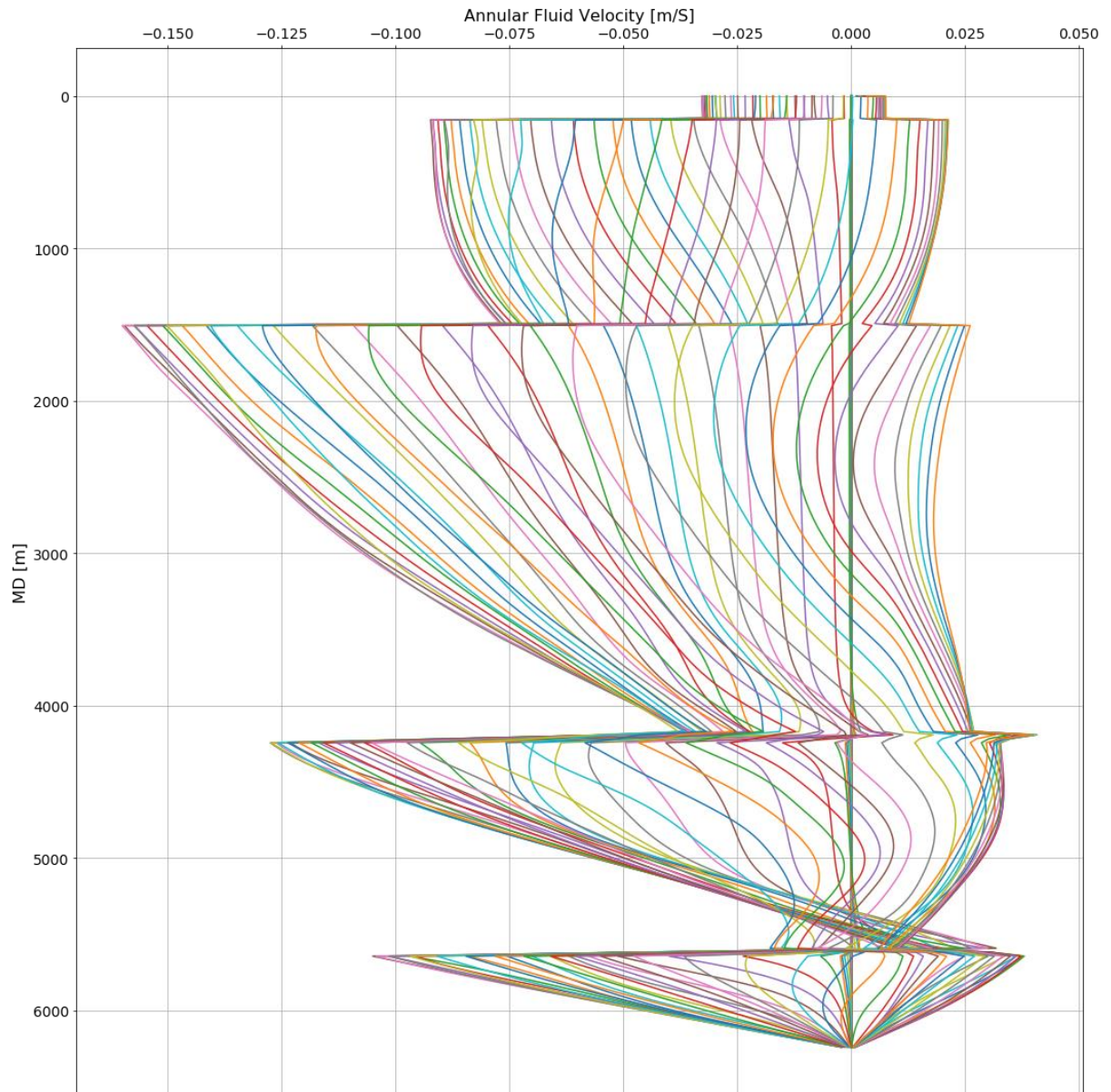


Figure 3.17 *Annular fluid velocity, merged interval No. 1 (0 - 111)*

As shown in Figure 3.17, the first dynamic (0 – 43) has lower amplitudes compare to the second dynamic (44 – 111). These two dynamics are illustrated separately in Figure 3.7 and Append. D.1. Singular value decomposition (SVD) and Singular values of the merged interval No. 1 are shown in Figure 3.18 and Figure 3.19.

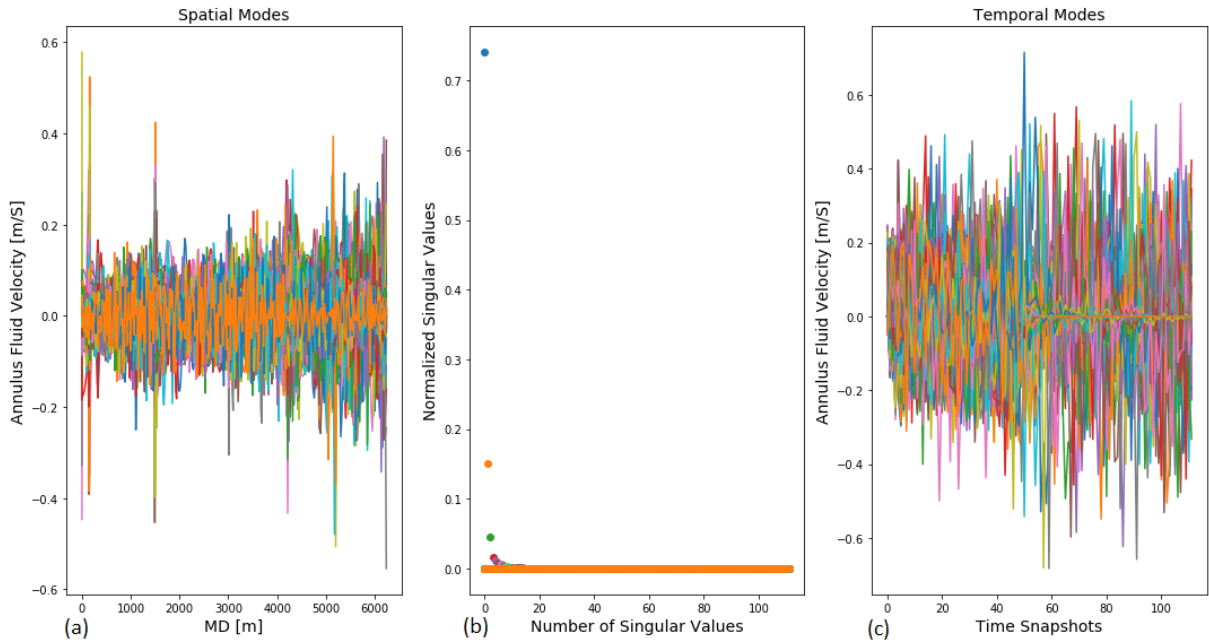


Figure 3.18 Singular value decomposition (SVD) of merged interval 1 (0 - 111)

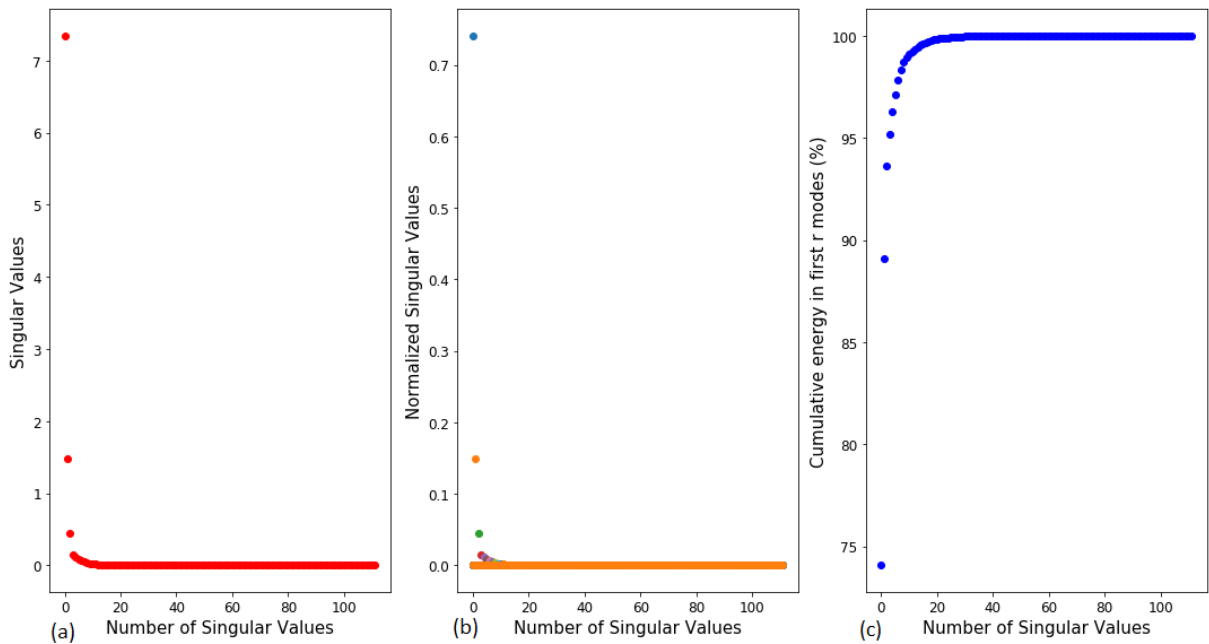


Figure 3.19 Singular values of merged interval No. 1 (0 - 111)

According to previous analyses, the maximum energy of interval No. 1 and 2 (corresponding to the largest singular value) are 79.13% and 74.07%, respectively, which indicate similarity of the modes in each interval. As shown in Figure 3.18b and Figure 3.19b, the energy of whole merged dataset is 74.07%, which indicates that great number of modes over interval No. 1 and 2 are similar. Figure 3.20 shows spatial modes and their dynamics used for reconstruction.

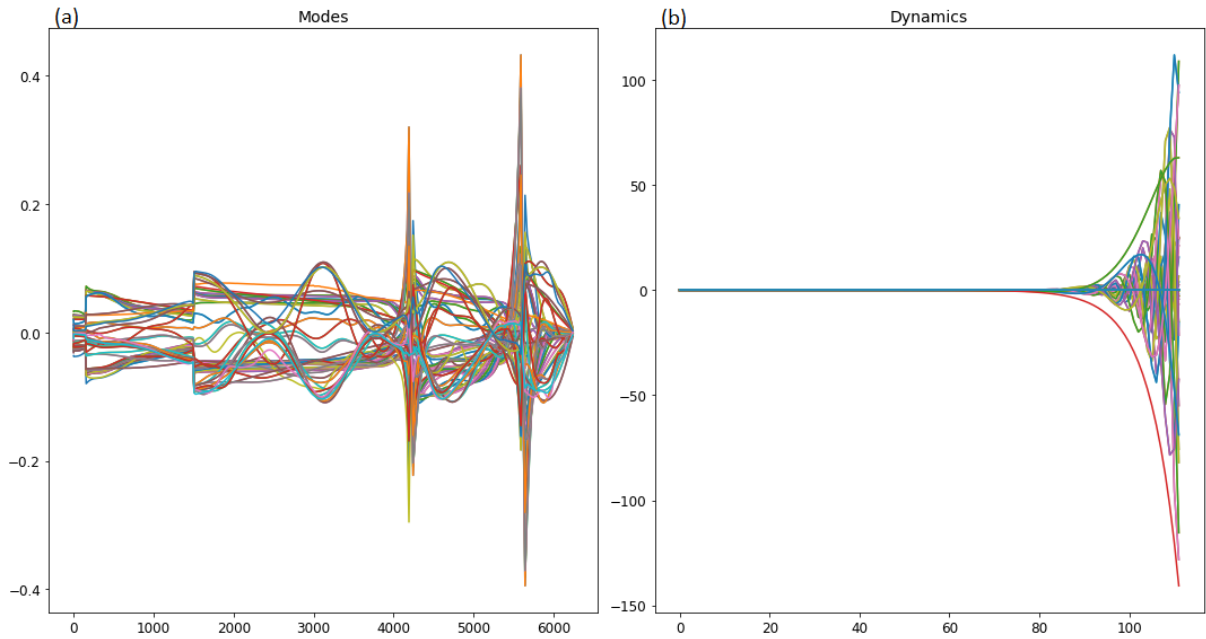


Figure 3.20 *The Spatial modes and their dynamics in time, merged interval 1 (0 - 100)*

As can be seen, by comparing captured modes in Figure 3.20a with modes in constituent intervals in Figure 3.10a and Append. D.4a, it seems that modes also merged together by merging two intervals. According to Figure 3.20b, dynamics of some modes are diverging very fast, which influence on reconstruction and limit prediction capability as a result of fast eigenvalues. Eigenvalues of merged interval 0 – 111 are shown in Figure 3.21.

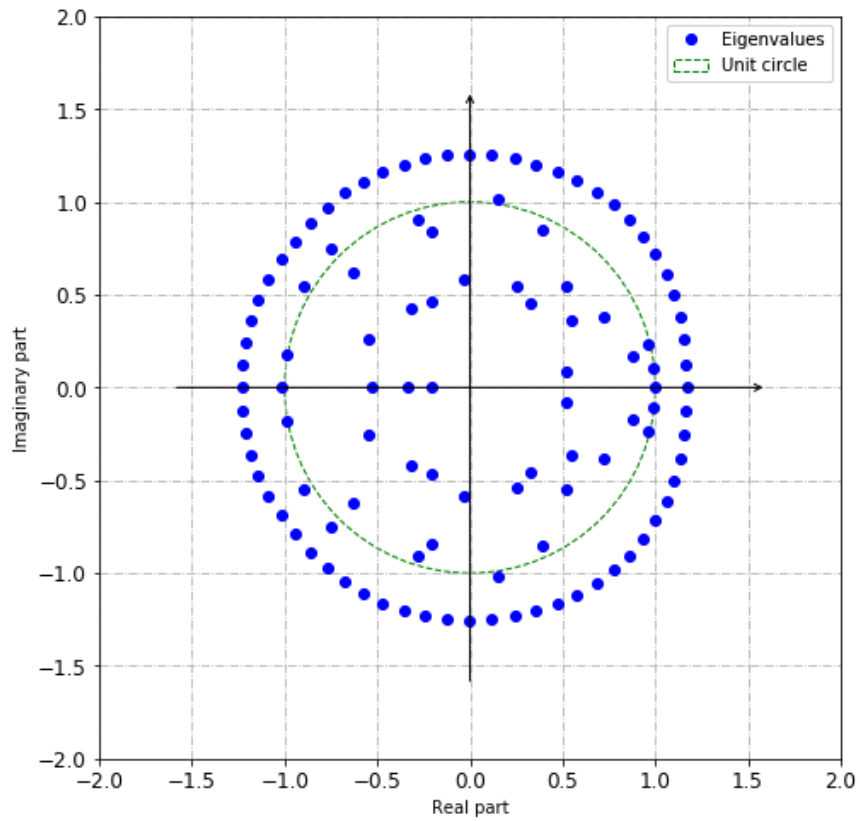


Figure 3.21 *Eigenvalues of merged interval 1 (0 - 111)*

By merging intervals, it seems their eigenvalues are merged as well, and this can be seen by looking at Figure 3.21 and constituent intervals eigenvalues Figure 3.11 and Append. D.5. Since now modes and eigenvalues of the merged interval and constituent intervals are almost identical and as a result good reconstruction is expected. 3D plots of DMD reconstruction and original measurements (merged interval 0 – 111) are shown in Figure 3.22 and Figure 3.23.

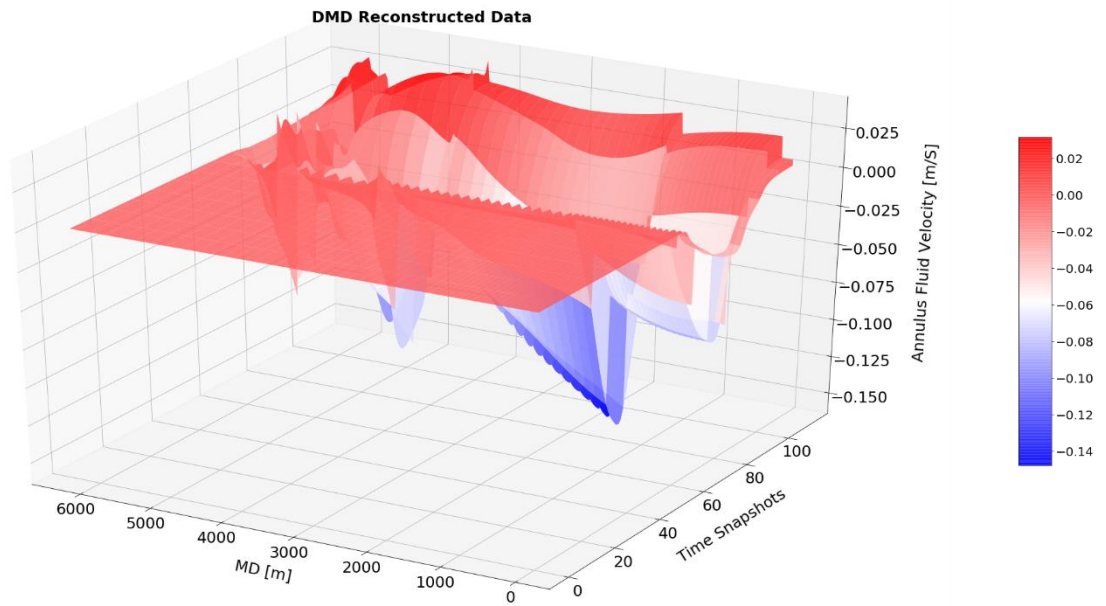


Figure 3.22 DMD reconstruction of merged interval No. 1 (0 - 111)

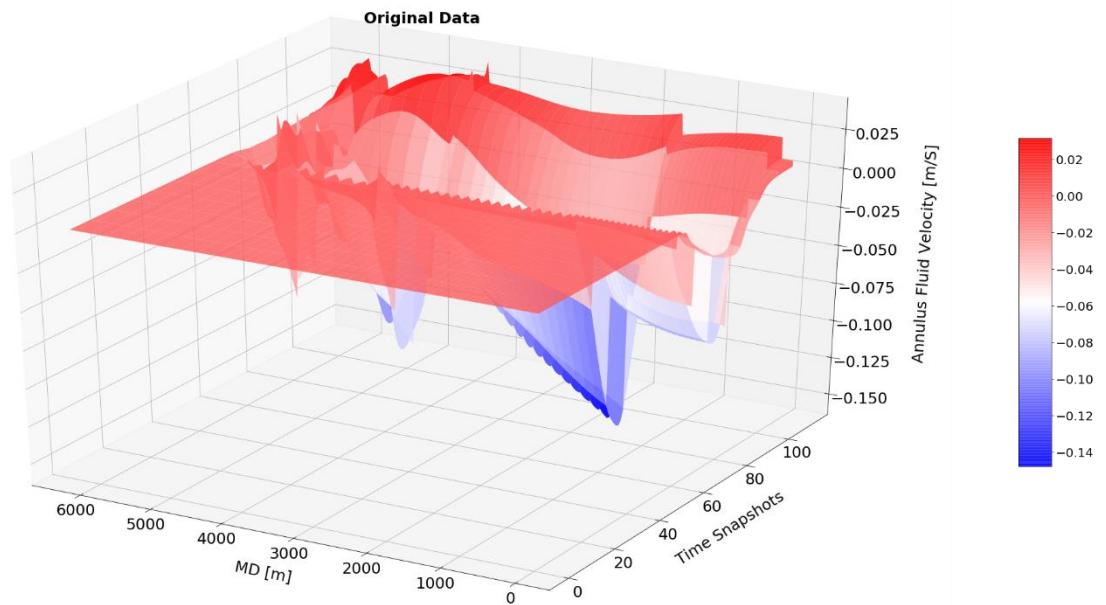


Figure 3.23 Original data, merged interval No. 1 (0 - 111)

Figure 3.22 and Figure 3.23 look similar with positive low oscillating velocities at acceleration interval and negative and high oscillating velocities at middle points. To find out the accuracy of DMD reconstruction, it is better to have a look at the figures which represent error.

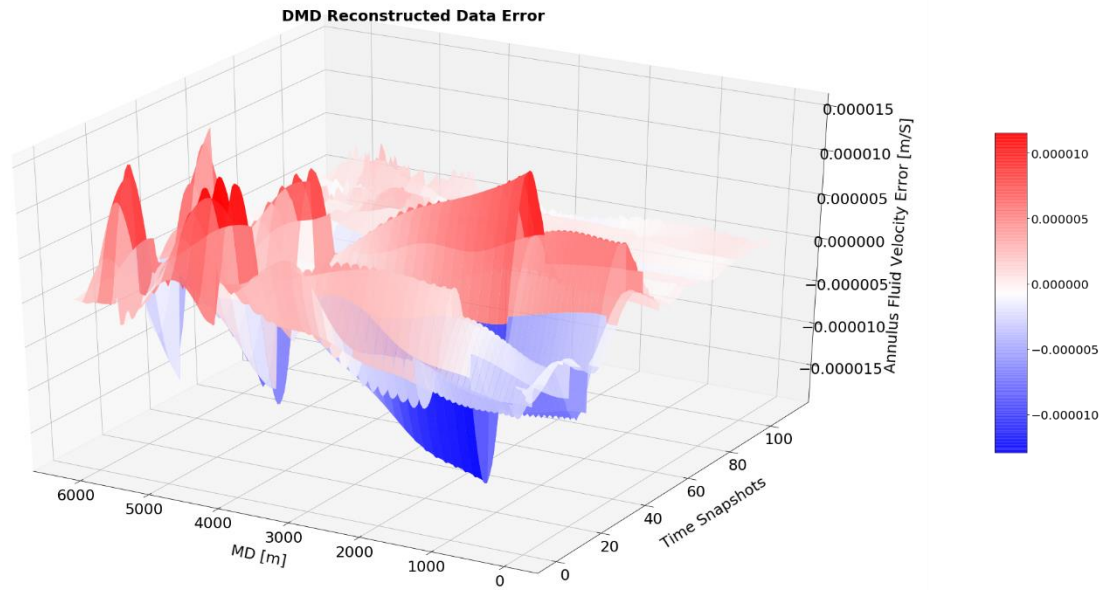


Figure 3.24 DMD reconstructed data error, merged interval No. 1 (0 – 111)

As can be seen in Figure 3.24, there are errors (no matter how small) in DMD reconstruction in spite of what was expected based on the eigenvalues and the spatial modes. The reason for errors could be initial amplitudes selected by algorithm using optimal amplitude extension. When there are dynamics with different ranges of amplitudes, optimal amplitudes measured by algorithm could be overestimated, underestimated or equal, compare to real values. In merged interval 1 (0 – 111), optimal amplitudes are overestimated for starting modes as they have small amplitudes compare to rest of measurements and DMD reconstructed values become larger than real data. Consequently, errors become positive which are shown by red color at the beginning timesteps. This is also true for the rest of dataset. Error distributions are shown in Figure 3.25 and Figure 3.26.

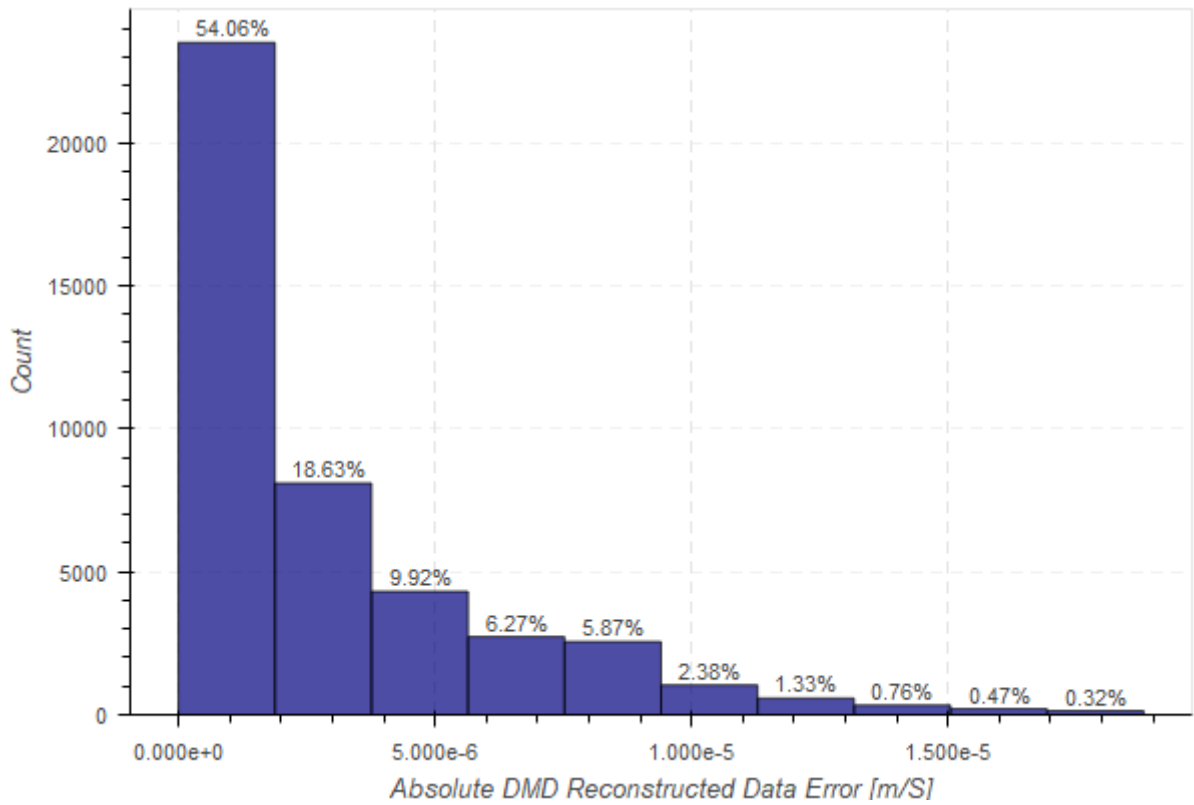


Figure 3.25 The frequency distribution of the absolute DMD reconstructed data error, merged interval No. 1 (0 - 111)

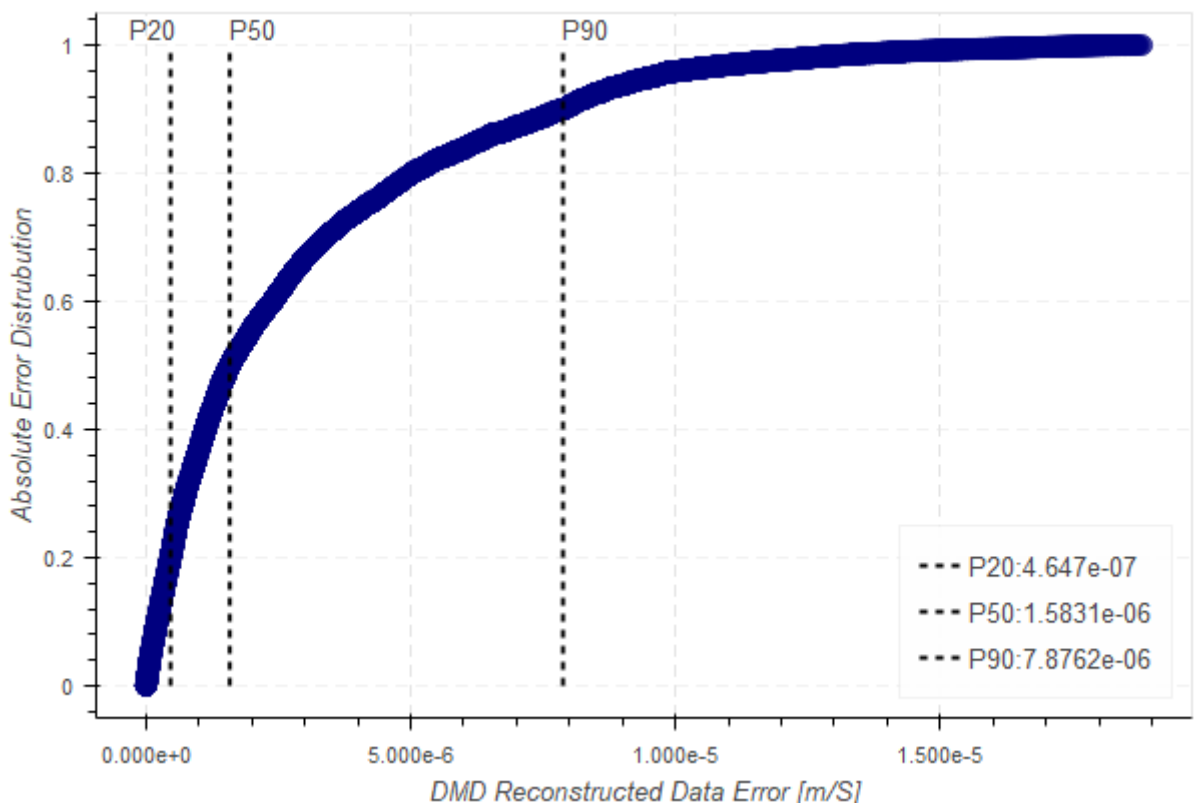


Figure 3.26 The distribution of the absolute DMD reconstructed data error, merged interval No. 1 (0 - 111)

According to Figure 3.25 and Figure 3.26, different ranges of errors exist, however they are still small and this is also indicated by P90 which is $7.9e-06$.

- Merged interval No. 2: 0 – 140

This interval contains three dynamics 0 – 43, 44 – 111 and 112 – 140 according to Table 1.

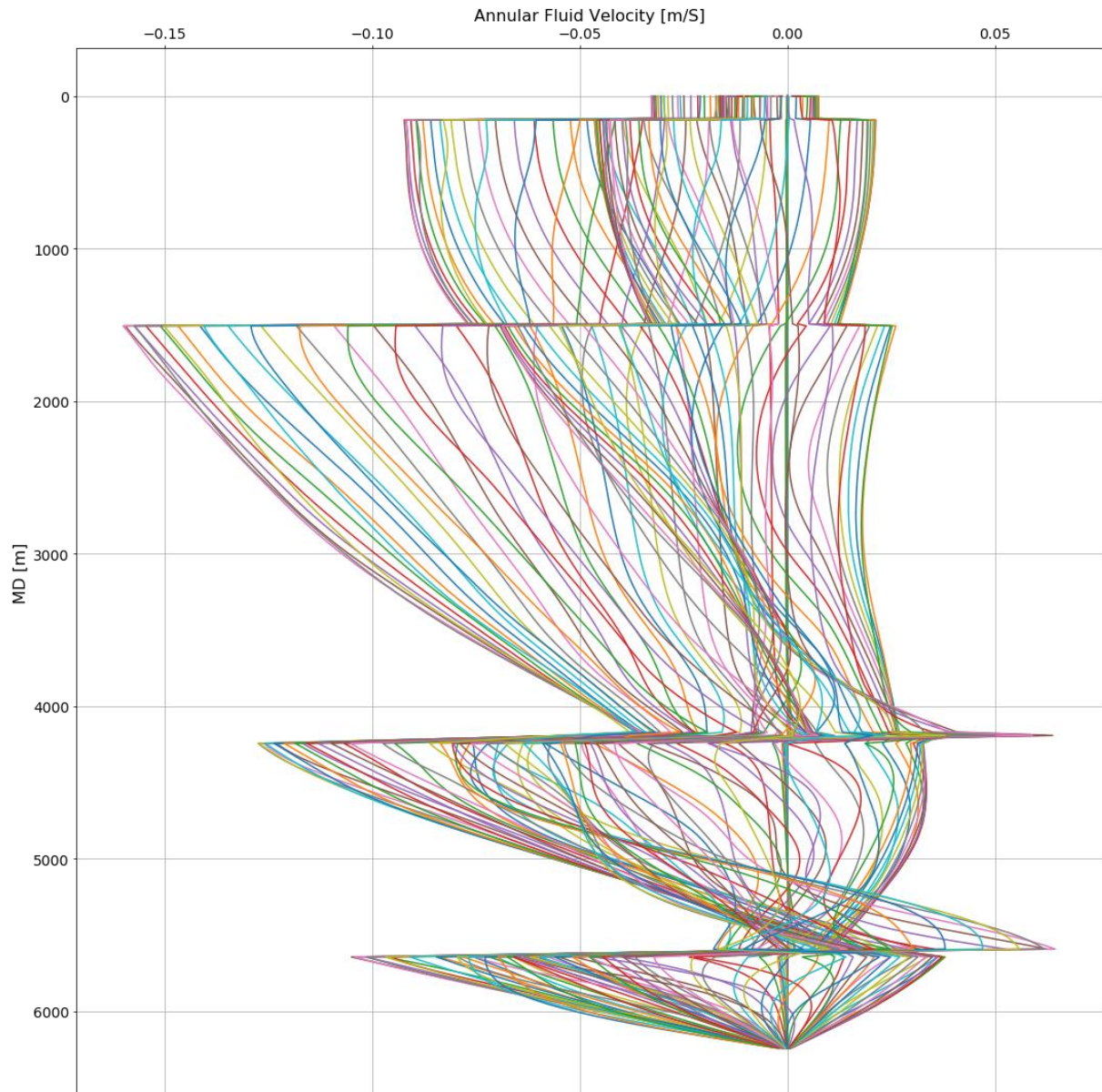


Figure 3.27 Annular fluid velocity, merged interval No. 2 (0 - 140)

By comparing merged interval No. 1 with merged interval No. 2, it can be seen that some oscillating measurements with high amplitudes are added (see Figure 3.27). Singular value decomposition (SVD) and Singular values of the merged interval No. 2 are shown in the following.

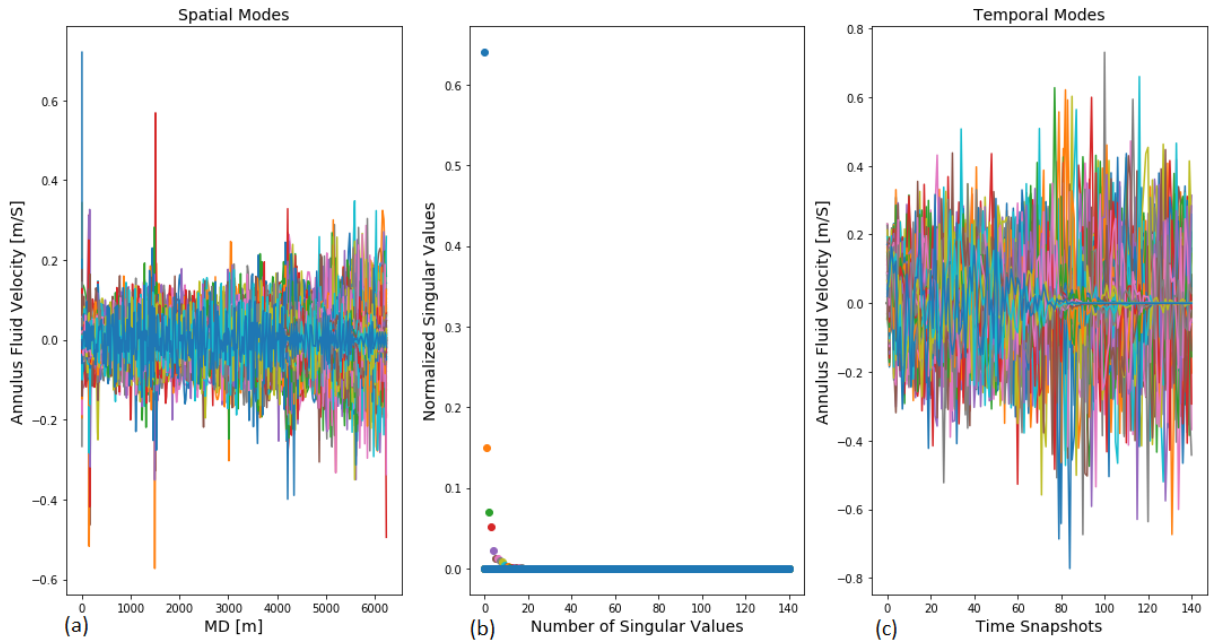


Figure 3.28 Singular value decomposition (SVD) of merged interval 2 (0 - 140)

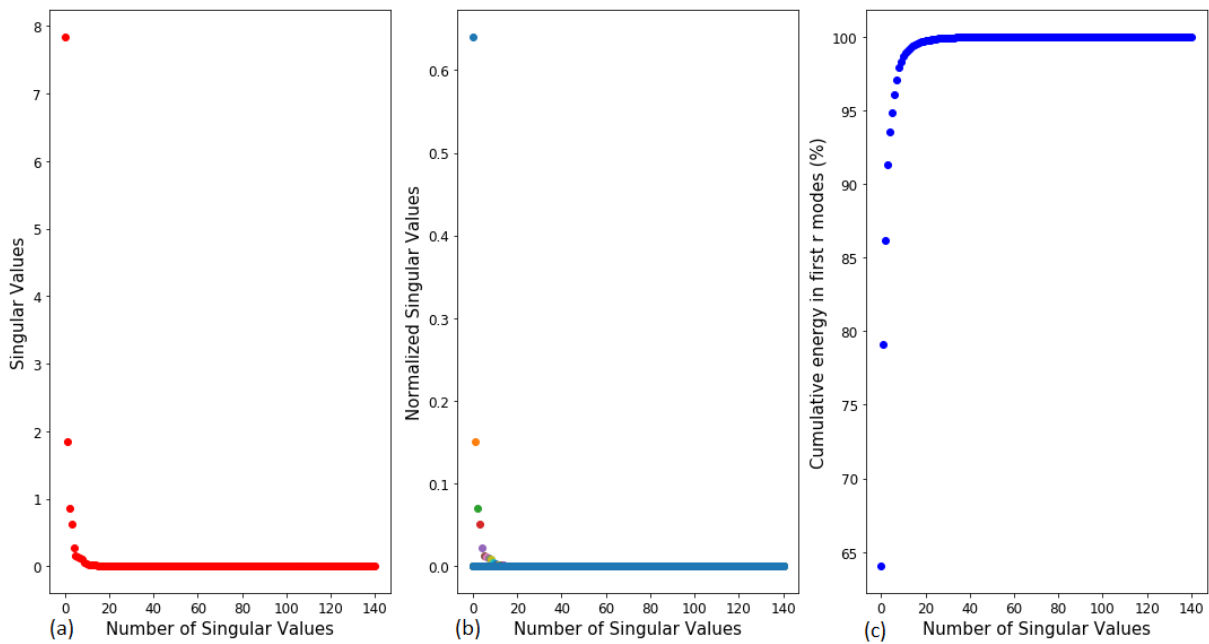


Figure 3.29 Singular values of merged interval No. 2 (0 - 140)

As shown in Figure 3.28 and Figure 3.29, the maximum energy of the system corresponding to the largest singular value is 64.03 %, which compare to previous merged interval is reduced. This means that the number of similar modes reduced by merging another interval. Figure 3.30 shows spatial modes and their dynamics used for reconstruction.

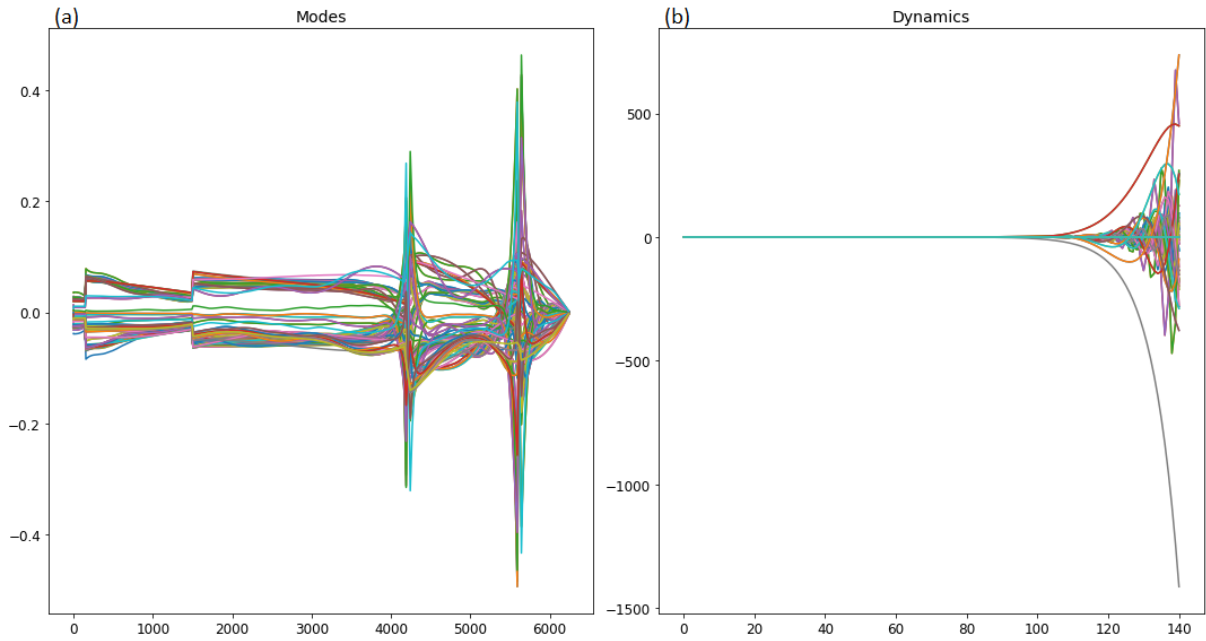


Figure 3.30 *The Spatial modes and their dynamics in time, merged interval 2 (0 - 140)*

As can be seen in Figure 3.30, some of the modes are diverging very fast as a result of the position of eigenvalues which are shown in Figure 3.31.

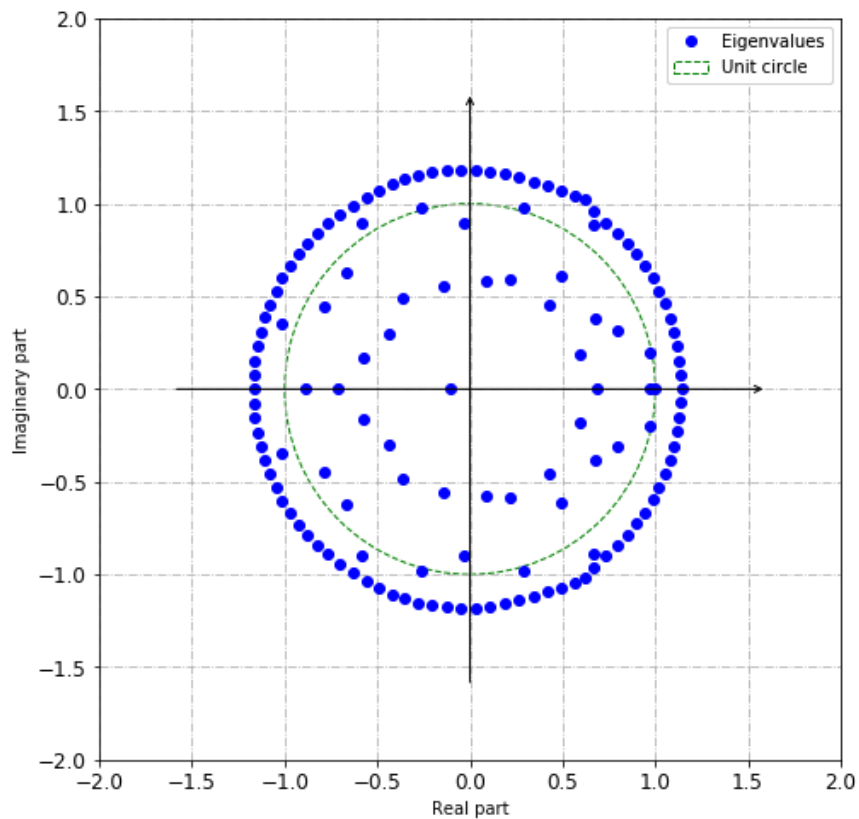


Figure 3.31 *Eigenvalues of merged interval 2 (0 - 140)*

By comparing Figure 3.31 with previous merged interval eigenvalues in Figure 3.21, it can be seen that most of the eigenvalues are located at short distance of unit circle except

acceleration interval which are located inside. 3D plots of DMD reconstruction and original measurements (merged interval 0 – 140) are shown in Figure 3.32 and Figure 3.33.

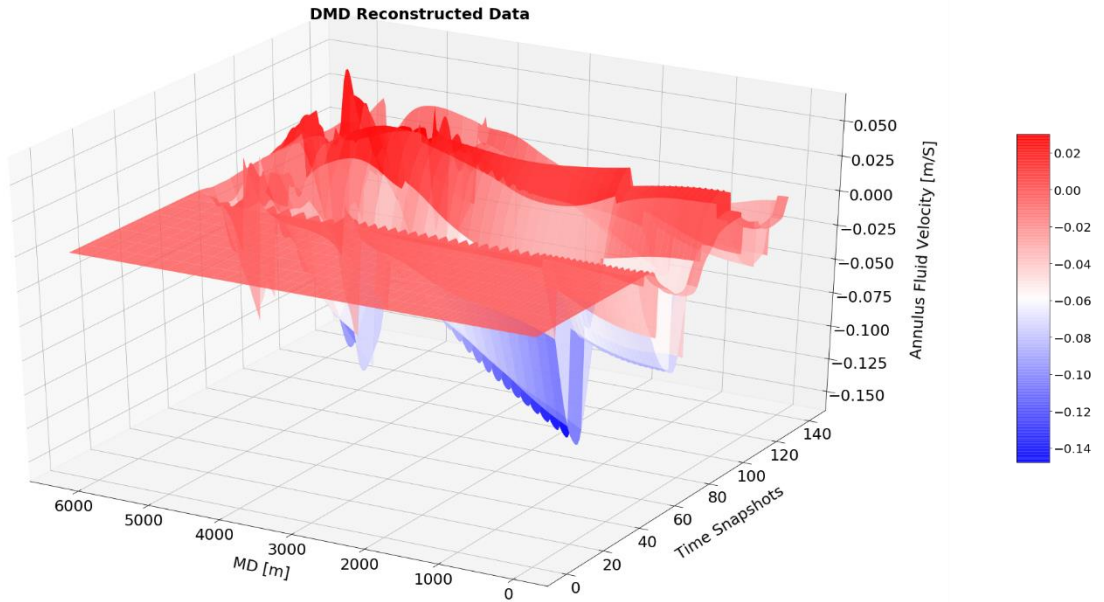


Figure 3.32 DMD reconstruction of merged interval No. 2 (0 - 140)

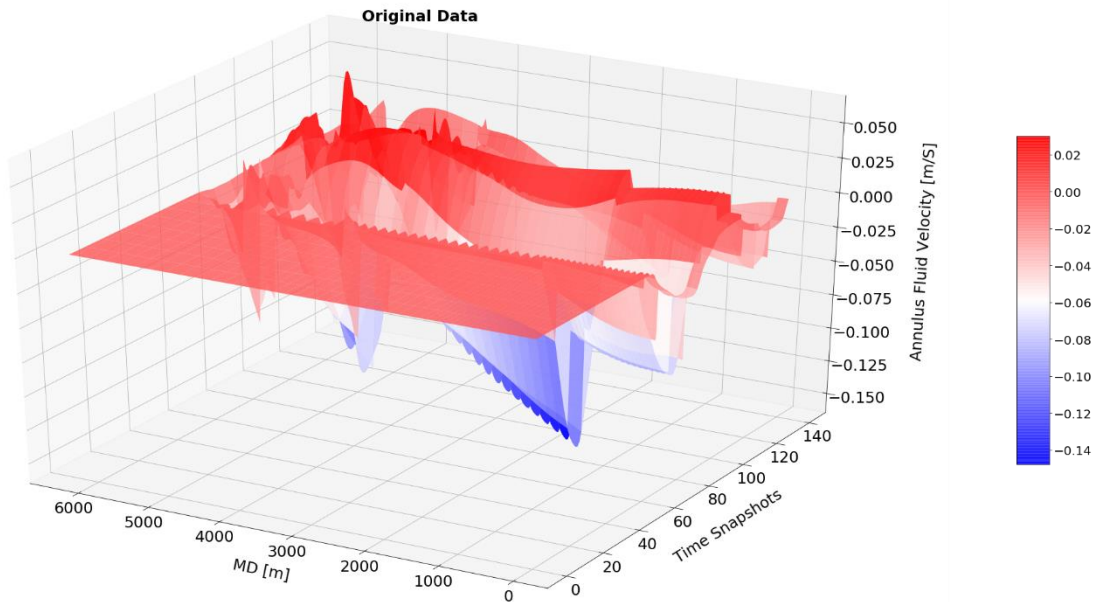


Figure 3.33 Original data, merged interval No. 2 (0 - 140)

As shown in Figure 3.32 and Figure 3.33, DMD reconstruction and original measurements look similar. To find out the accuracy of DMD reconstruction, it is better to have a look at the figures which represent error.

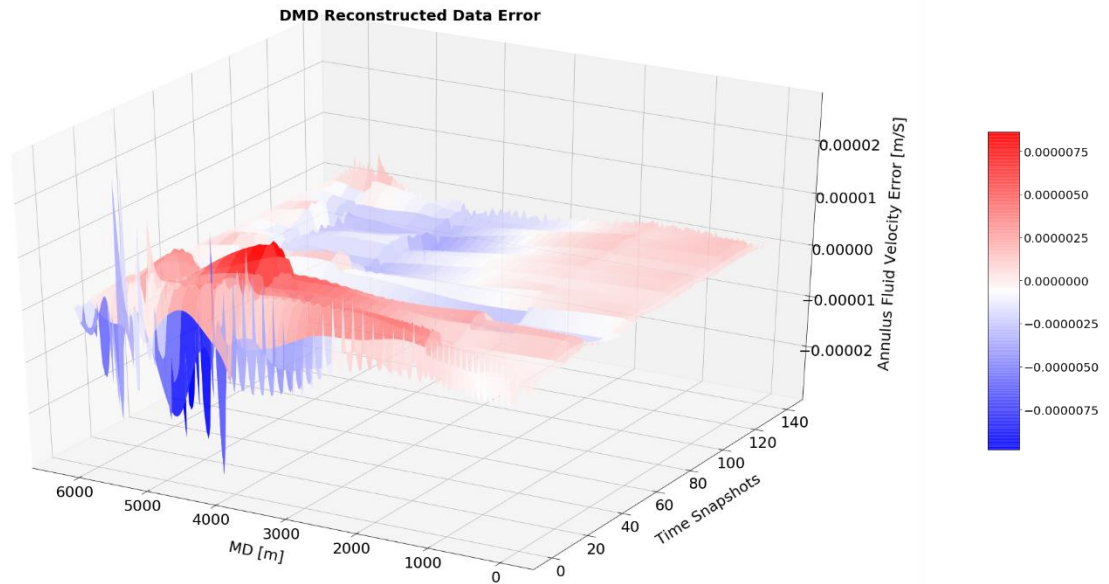


Figure 3.34 DMD reconstructed data error, merged interval No. 2 (0 – 140)

As clearly shown in Figure 3.34, acceleration interval is overestimated as expected, for the reason that the calculated initial amplitudes of DMD are higher than real measurements. Compare to previous merged interval reconstruction, lower errors are observed for the rest of interval and this is occurred as more high-amplitude measurements added to the interval and as a result, initial amplitudes of DMD are close to real measurements. Error distributions are shown in Figure 3.35 and Figure 3.36.

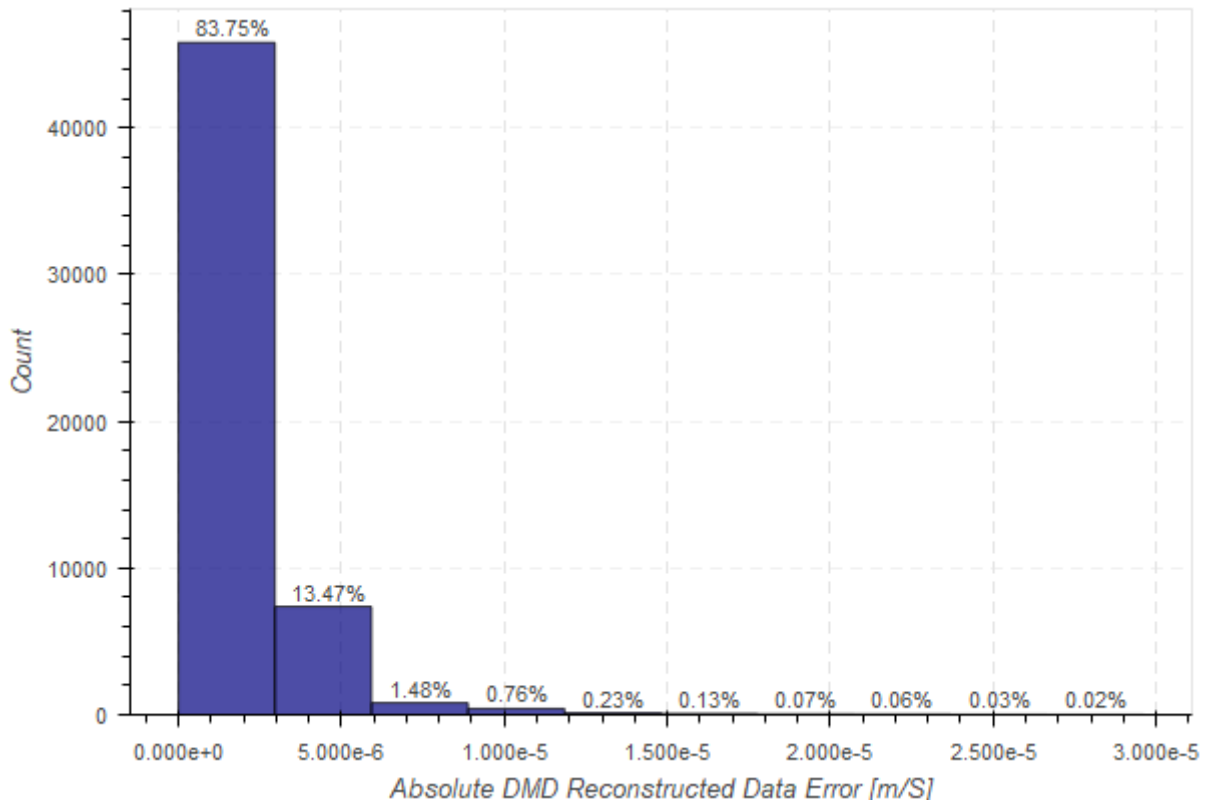


Figure 3.35 The frequency distribution of the absolute DMD reconstructed data error, merged interval No. 2 (0 - 140)

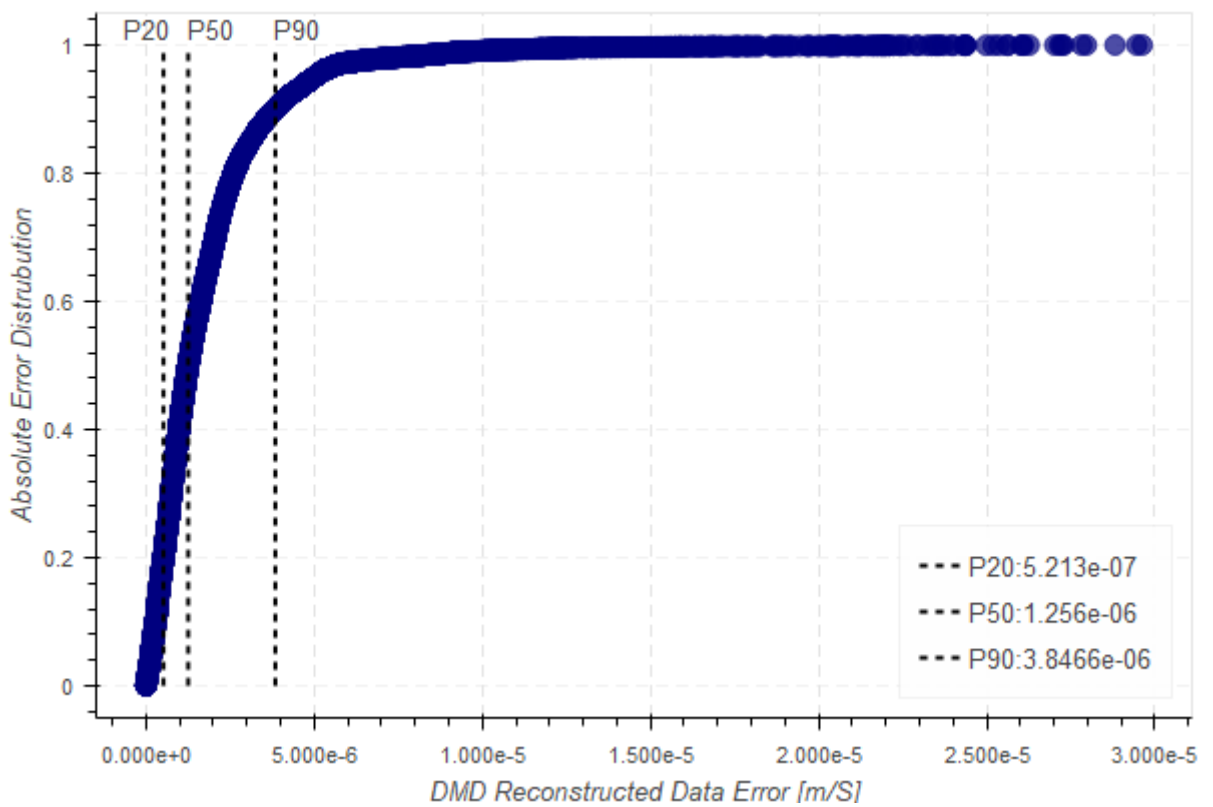


Figure 3.36 The distribution of the absolute DMD reconstructed data error, merged interval No. 2 (0 - 140)

When there are similar dynamics with the same range of amplitudes in an interval, better reconstruction accuracy will be achieved and this is the reason for improved performance of algorithm over the interval 0-140, which are shown in Figure 3.35 and Figure 3.36.

- Merged interval No. 3: 0 – 161

This interval contains four dynamics, including 0 – 43, 44 – 111, 112 – 140, 141– 161 according to Table 1.

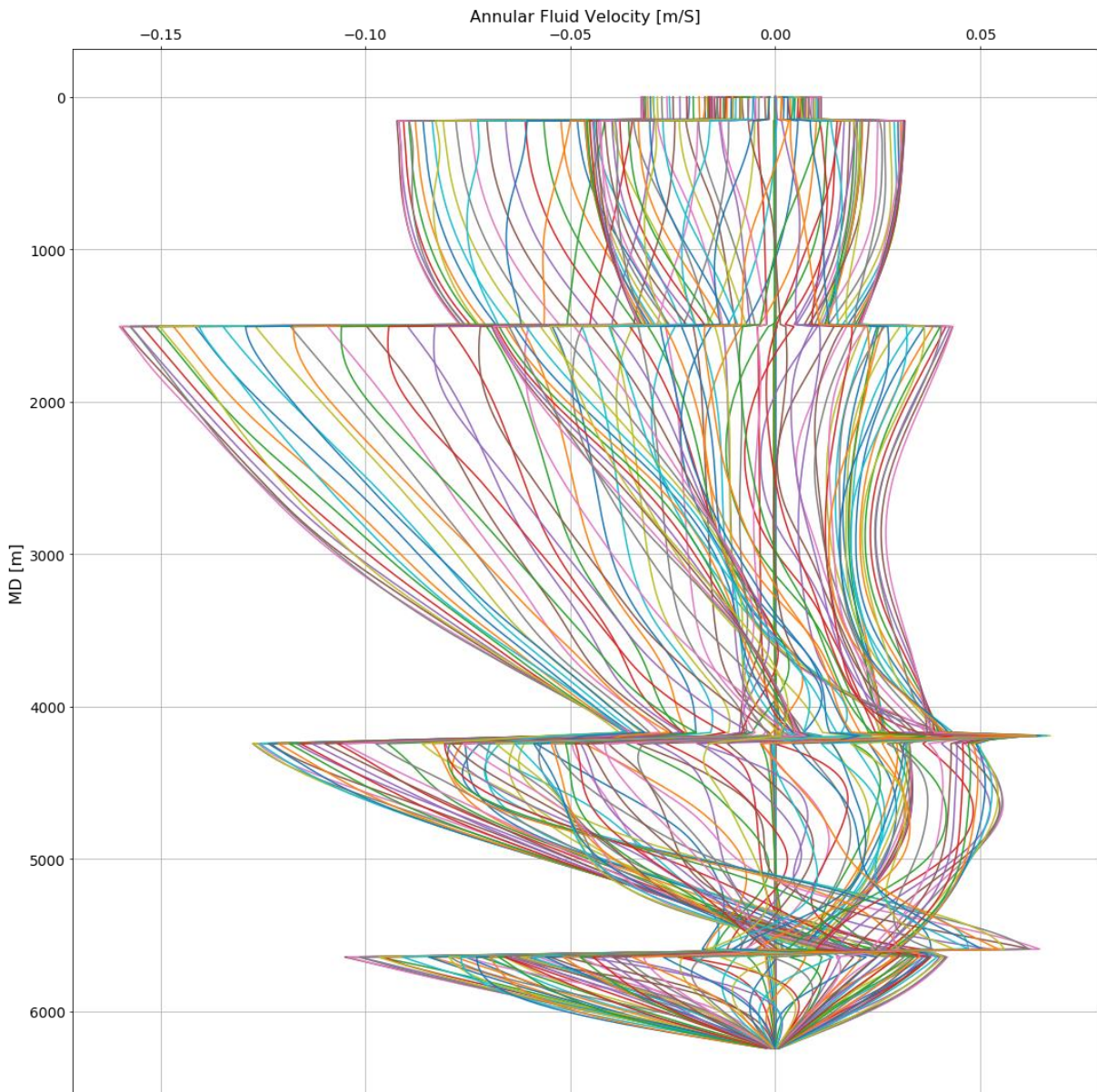


Figure 3.37 Annular fluid velocity, merged interval No. 3 (0 - 161)

To find out added measurements, it is better to compare Figure 3.37 with Figure 3.27. The added measurements are located at the right side of the figure, which oscillate with higher amplitudes at lower section of the well compare to the previous measurements. Singular value

decomposition (SVD) and Singular values of the merged interval No. 3 are shown in Figure 3.38 and Figure 3.39.

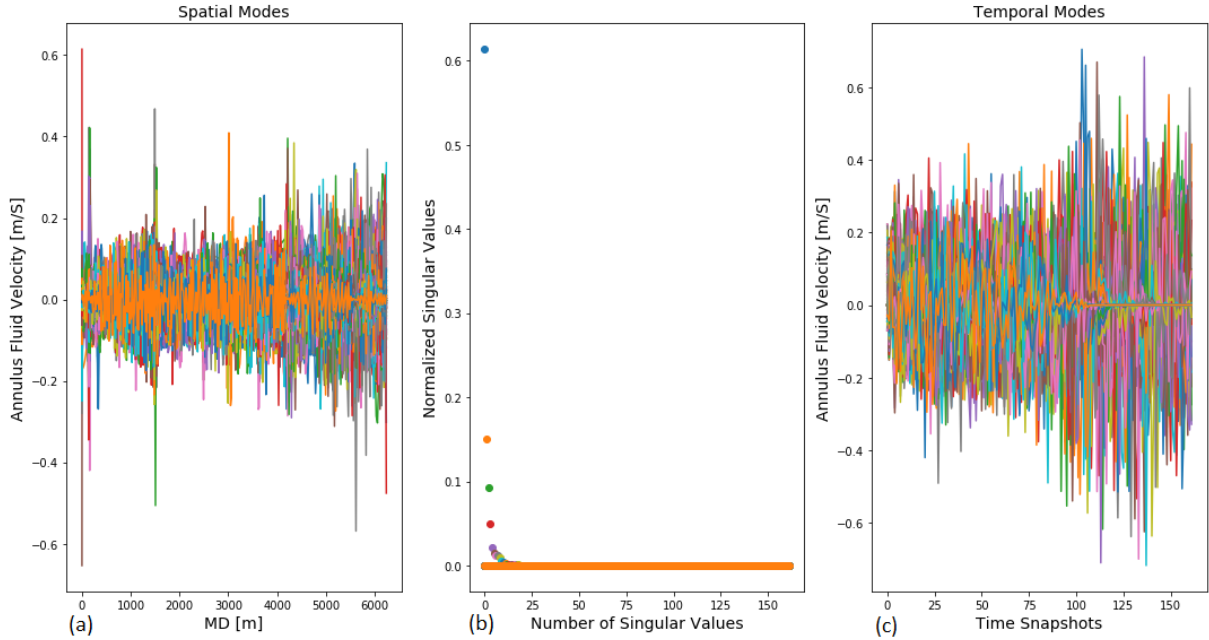


Figure 3.38 Singular value decomposition (SVD) of merged interval 3 (0 - 161)

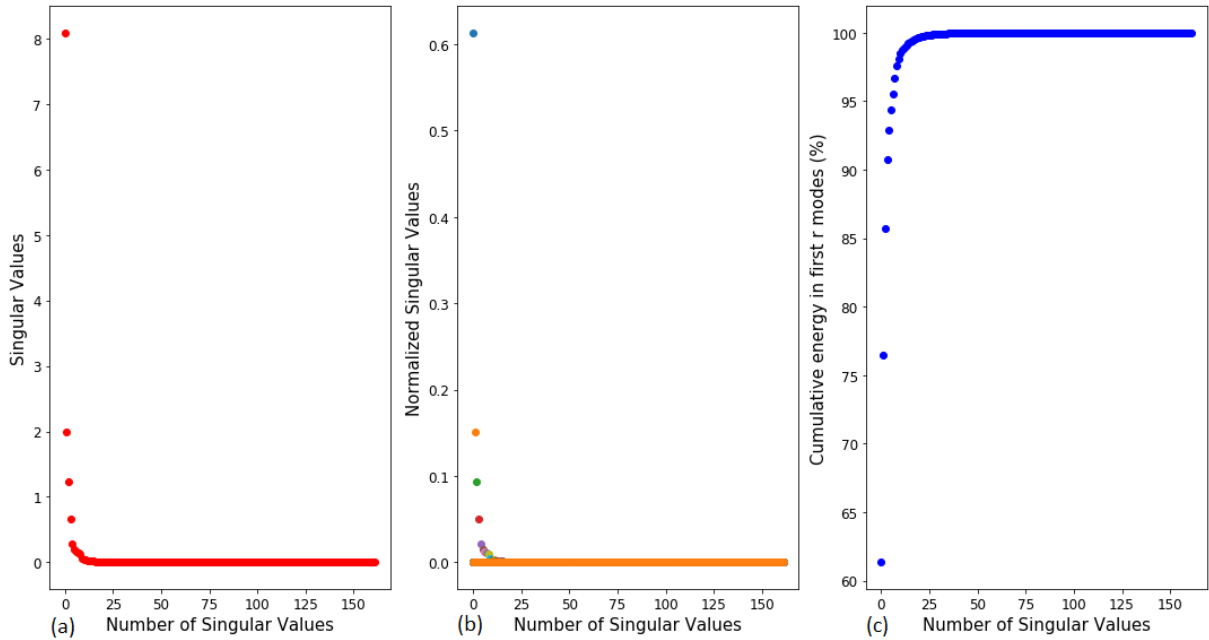


Figure 3.39 Singular values of merged interval No. 3 (0 - 161)

As can be seen in Figure 3.38 and Figure 3.39, by adding new measurements to previous merged interval, the maximum energy of the system corresponding to the largest singular value

is reduced to 61.35%, meaning that the number of similar modes are reduced. Figure 3.40 shows spatial modes and their dynamics used for reconstruction.

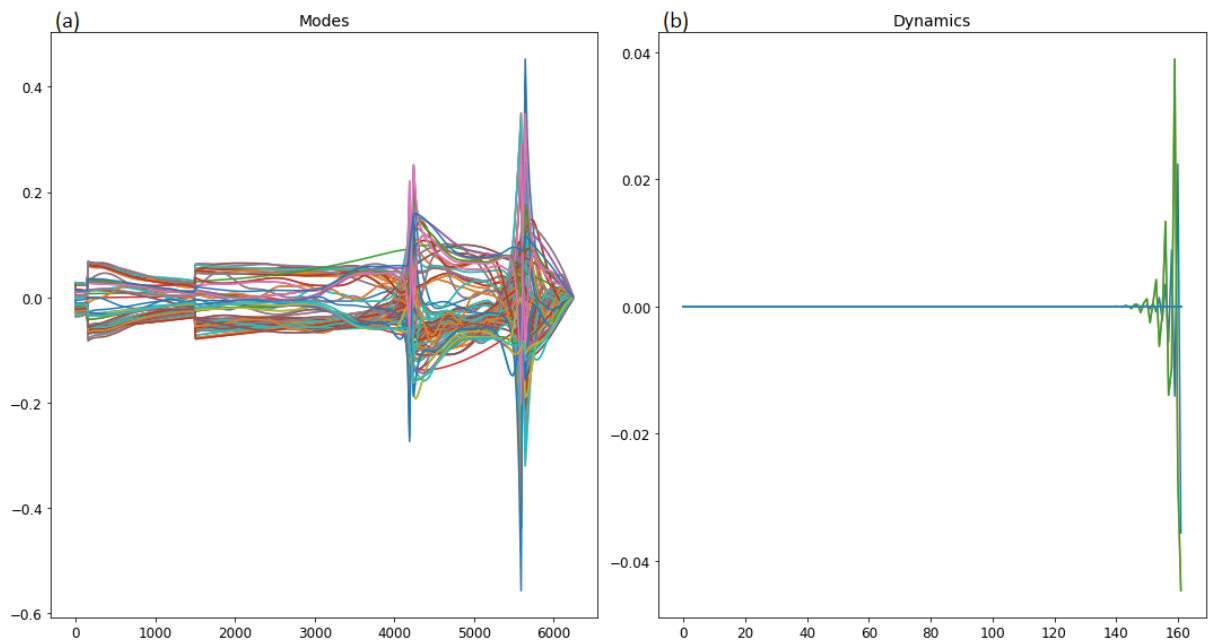


Figure 3.40 *The Spatial modes and their dynamics in time, merged interval 3 (0 - 161)*

As shown in Figure 3.40b, fast modes cause that some dynamics converge soon at the beginning and some dynamics diverge at the end. For better understanding, it is better to have look at eigenvalues in Figure 3.41.

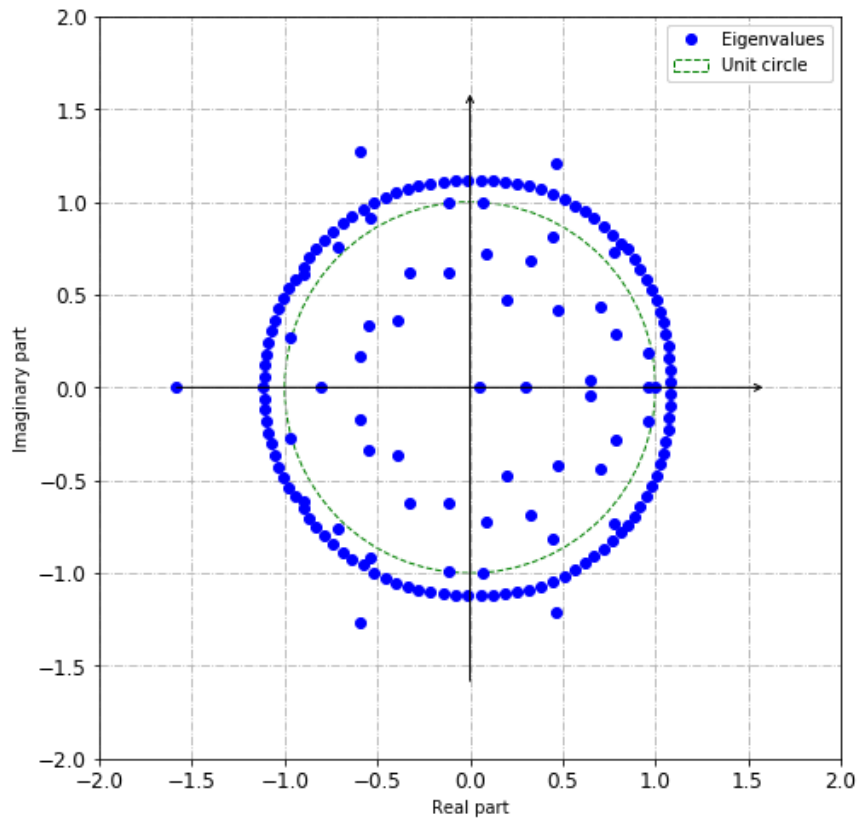


Figure 3.41 *Eigenvalues of merged interval 3 (0 - 161)*

According to Figure 3.41 and previous analyses, it is perceived that the circle made by eigenvalues is the stability of the modes, thus five eigenvalues outside the eigenvalues circle may fail the reconstruction. 3D plots of DMD reconstruction and original measurements (merged interval 0 – 161) are shown in Figure 3.42 and Figure 3.43.

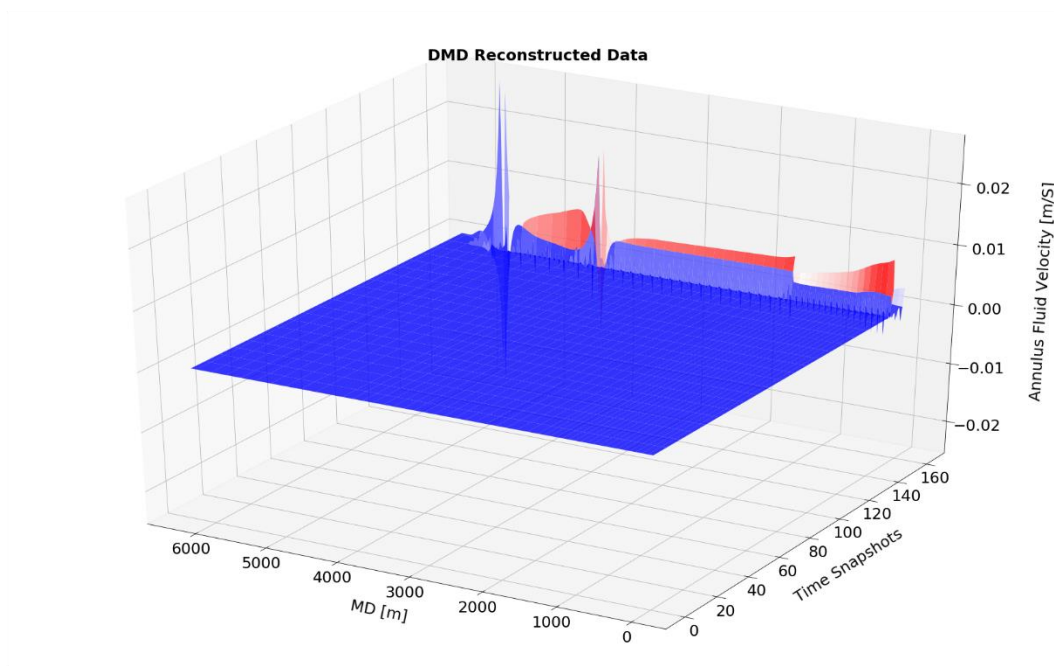


Figure 3.42 DMD reconstruction of merged interval No. 3 (0 - 161)

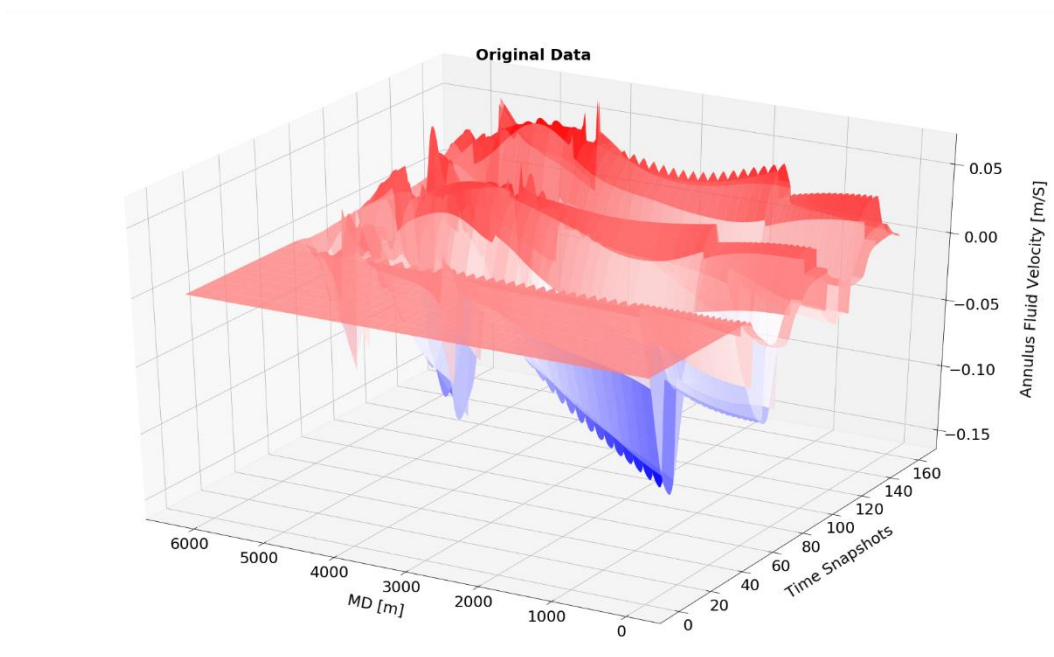


Figure 3.43 Original data, merged interval No. 3 (0 - 161)

By comparing Figure 3.42 and Figure 3.43, it is obvious that DMD reconstruction is not preferable as expected. To find out the accuracy of DMD reconstruction, it is better to have a look at the figures which represent error.

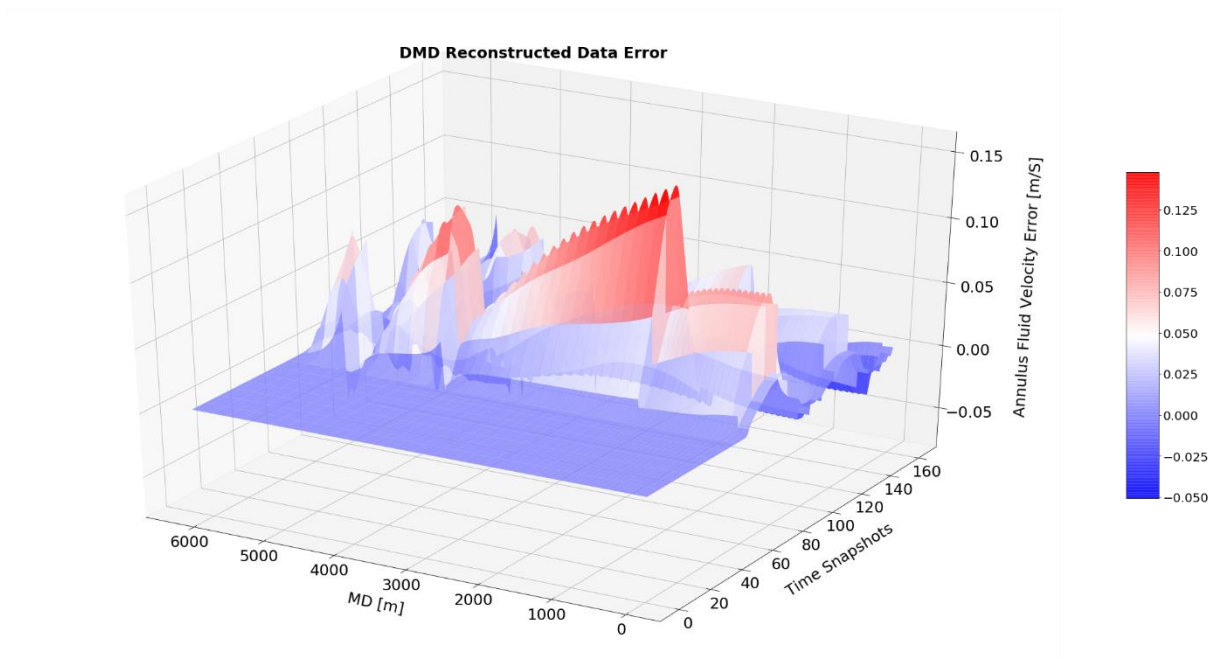


Figure 3.44 DMD reconstructed data error, merged interval No. 3 (0 – 161)

As shown in Figure 3.44, error values are high and, in some points, errors are even equal to measured data. Error distributions are shown in Figure 3.45 and Figure 3.46.

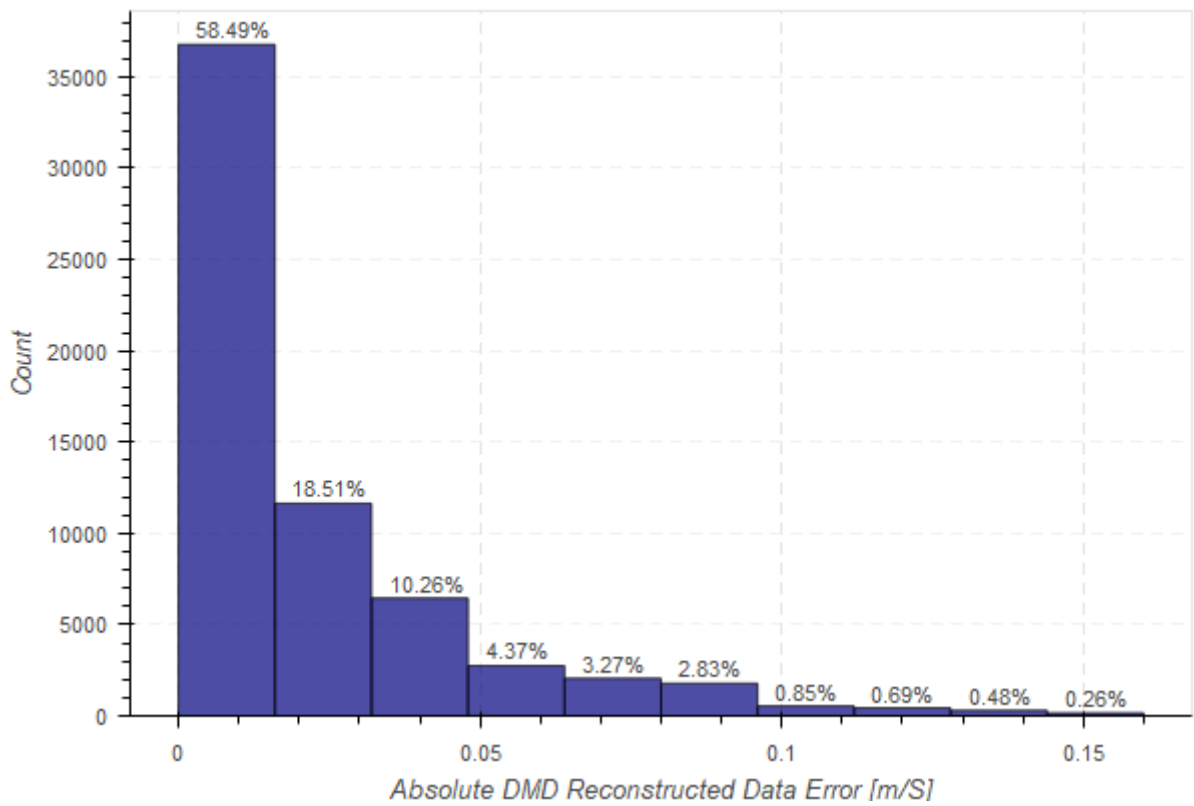


Figure 3.45 The frequency distribution of the absolute DMD reconstructed data error, merged interval No. 3 (0 – 161)

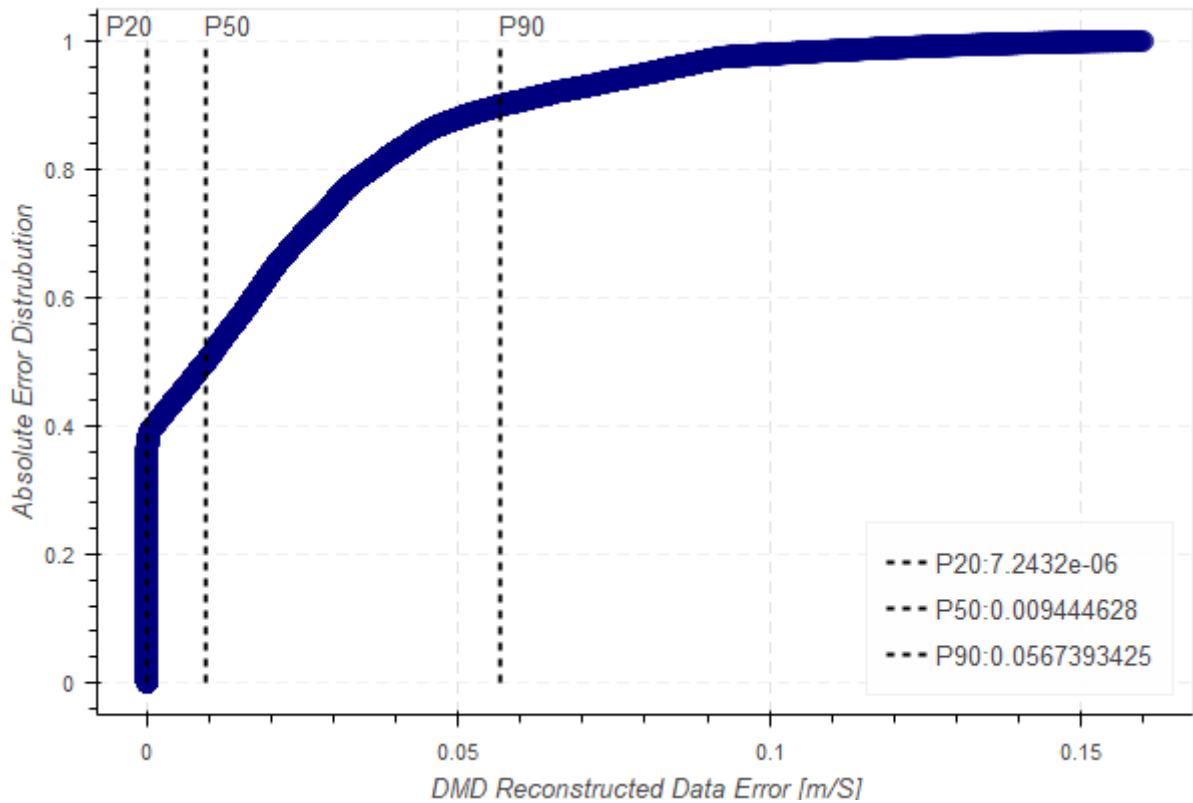


Figure 3.46 The distribution of the absolute DMD reconstructed data error, merged interval No. 3 (0 - 161)

As can be seen in Figure 3.45 and Figure 3.46, performance of DMD reconstruction is unsatisfactory, as there are high error ranges (as an example 0.1 to 0.15) in error distribution and also P90 is 0.056, which is unfavorable. As mentioned before, similar patterns were used to classify different dynamics in measured data and it worked successfully for single sub datasets, even though it was difficult to separate measurements related to transition to next dynamic. Classifying intervals based on the similar patterns is not working for merged intervals, which may contain transitional data from next dynamic, and this was shown by unstable eigenvalues in the merged interval No. 3. Stability of modes based on the position of eigenvalues on the unit circle could be a recommended solution that needs to be checked to separate different dynamics in a dataset. In this regard, five columns of merged interval No. 3 are removed and DMD algorithm is going to be implemented in the following.

- Corrected merged interval No. 3: 0 – 156

By removing five columns from the merged interval 3, SVD and singular values do not change considerably and to avoid repetition, they are skipped. Figure 3.47 shows spatial modes and their dynamics used for DMD reconstruction.

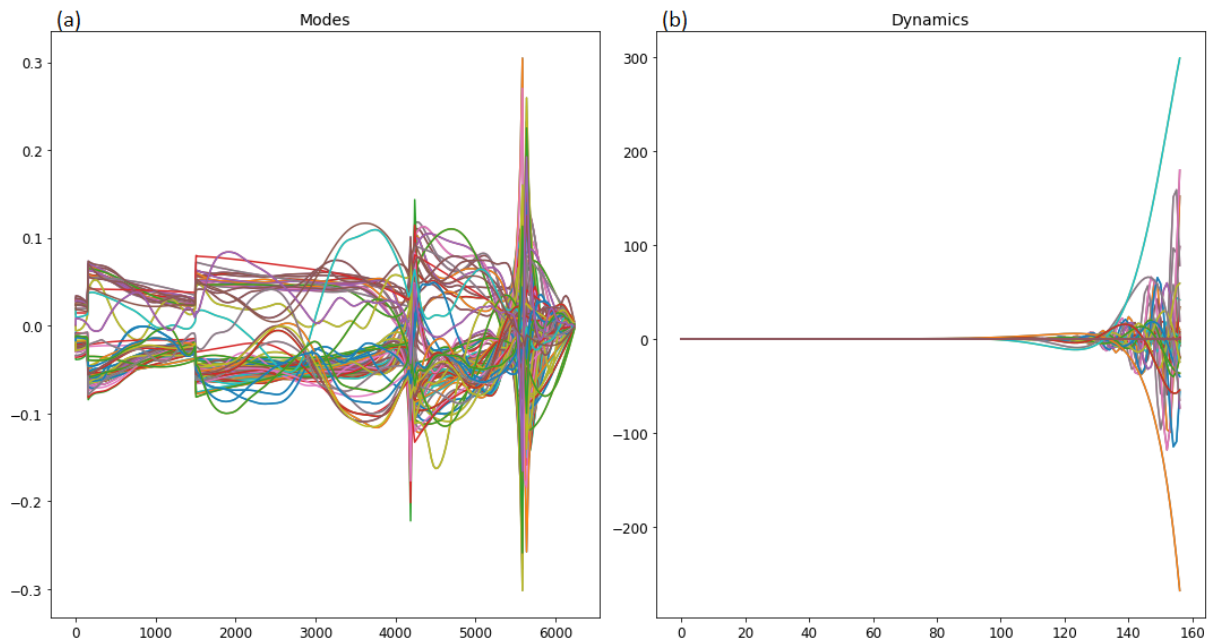


Figure 3.47 *The Spatial modes and their dynamics in time, corrected merged interval 3 (0 - 156)*

By comparing Figure 3.47b and Figure 3.40b, it is observed that in both figures, some dynamics converge at the beginning and some diverge at the end, depending on the positions of eigenvalues. In corrected interval, oscillations of modes are smoother as a result of stable eigenvalues. Eigenvalues of the corrected merged interval are shown in Figure 3.48.

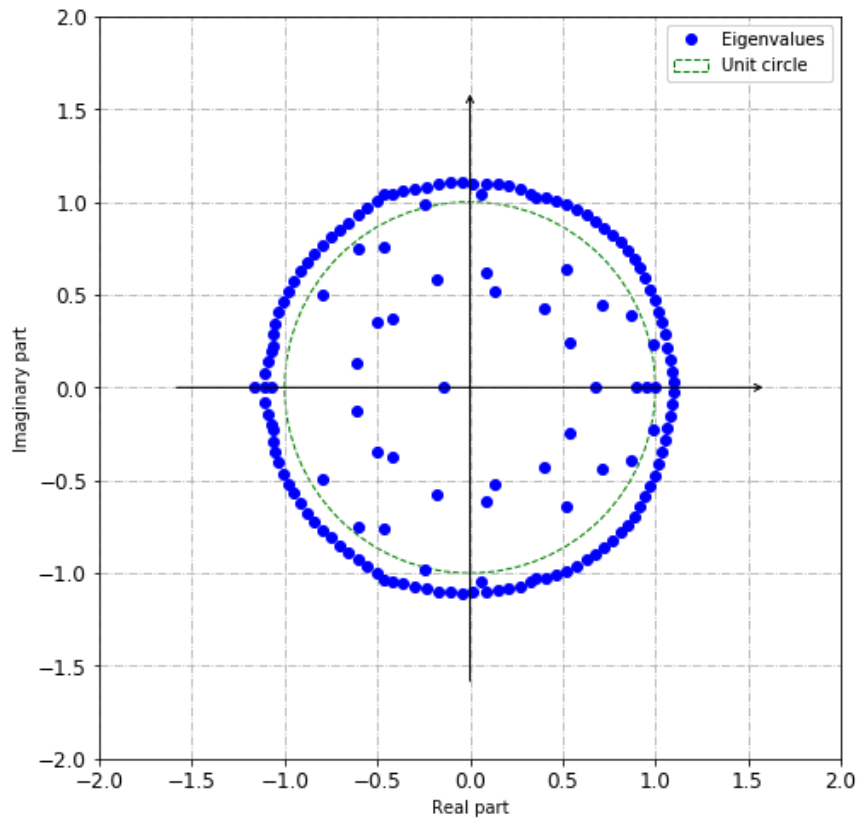


Figure 3.48 Eigenvalues of corrected merged interval 3 (0 - 156)

By removing measurements which do not belong to tested interval, stable eigenvalues achieved (see Figure 3.48). 3D plots of DMD reconstruction and original measurements (corrected merged interval 0 – 156) are shown in Figure 3.49 and Figure 3.50.

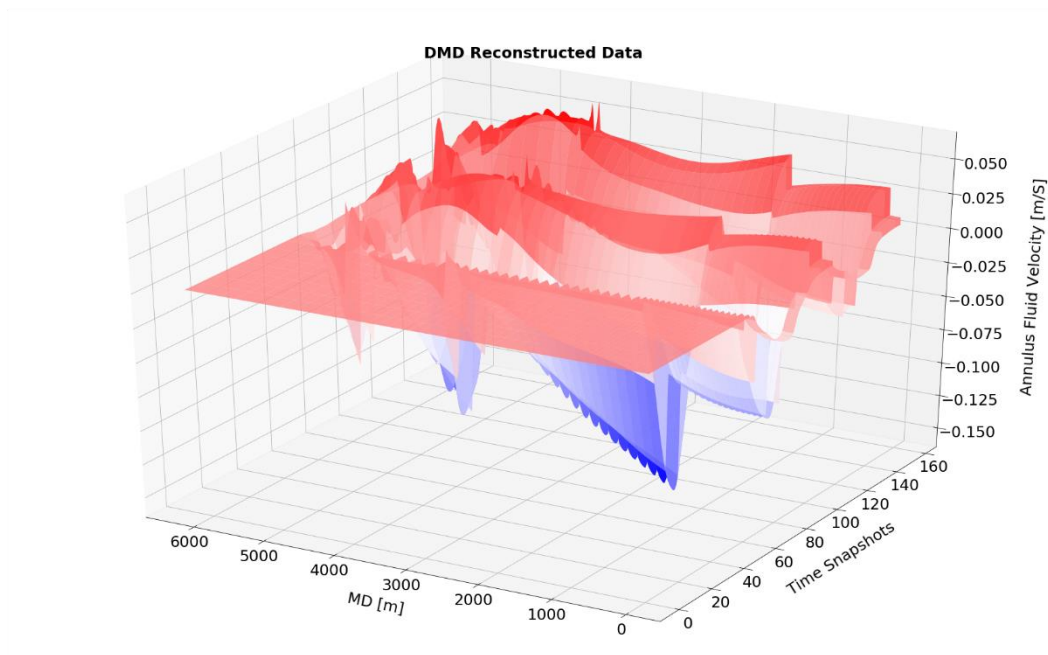


Figure 3.49 DMD reconstruction of corrected merged interval No. 3 (0 - 156)

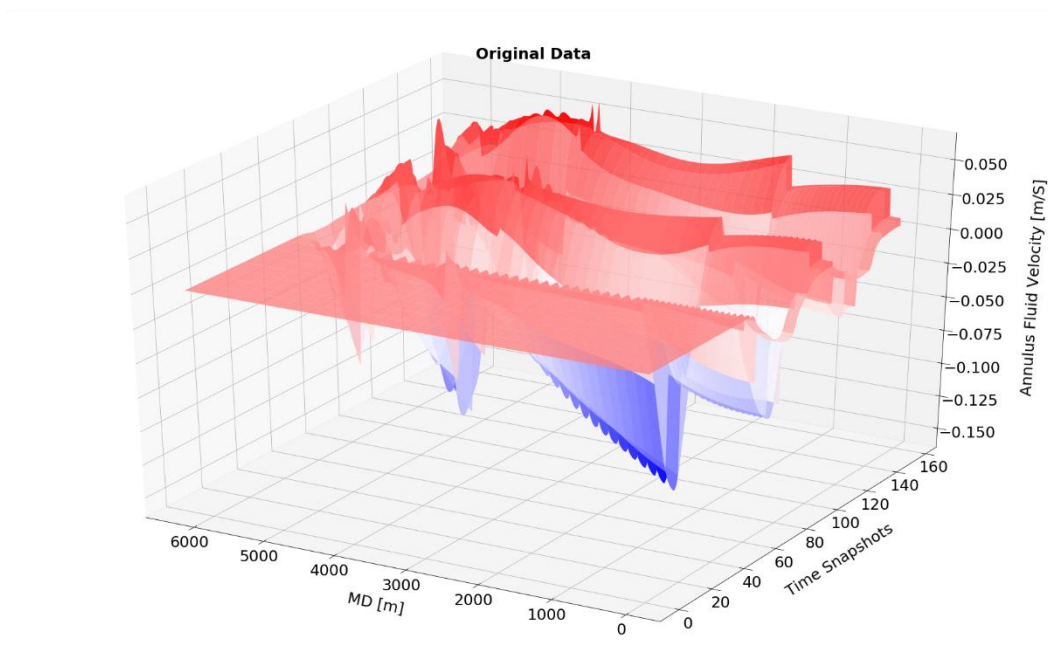


Figure 3.50 Original data, corrected merged interval No. 3 (0 - 156)

As can be seen in Figure 3.49 and Figure 3.50, DMD reconstruction and original measurements look similar, as expected. To find out the accuracy of DMD reconstruction, it is better to have a look at the figures which represent error.

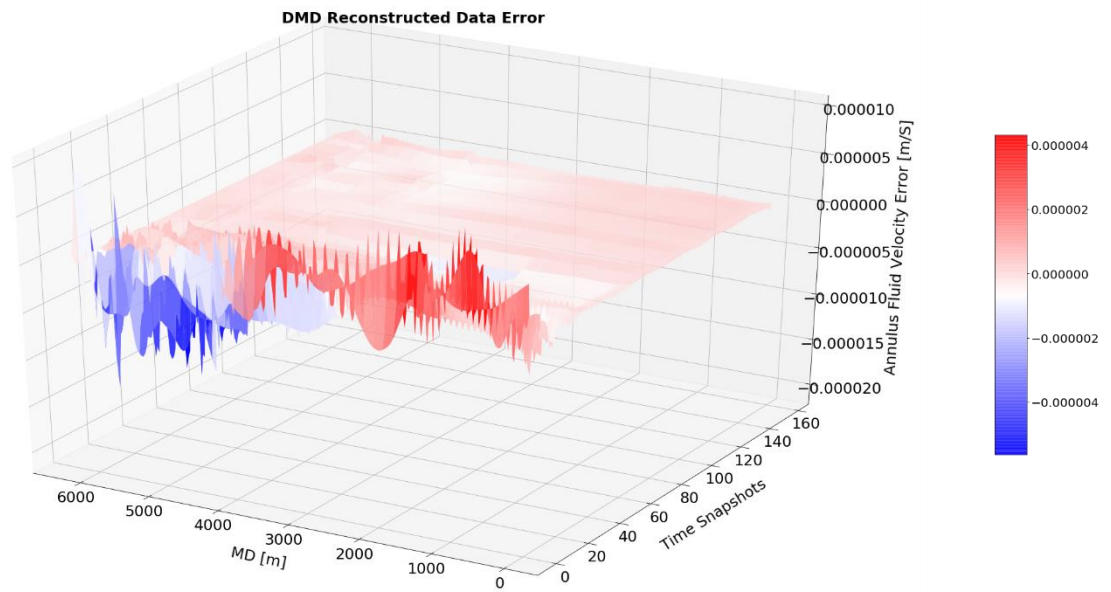


Figure 3.51 *DMD reconstructed data error, corrected merged interval No. 3 (0 – 156)*

As mentioned earlier, acceleration interval is overestimated as expected, for the reason that the amplitudes of the acceleration interval are lower than the rest of measurement amplitudes and as a result, calculated initial amplitudes of DMD for starting timesteps are higher than real measurements. Error distributions are shown in Figure 3.52 and Figure 3.53.

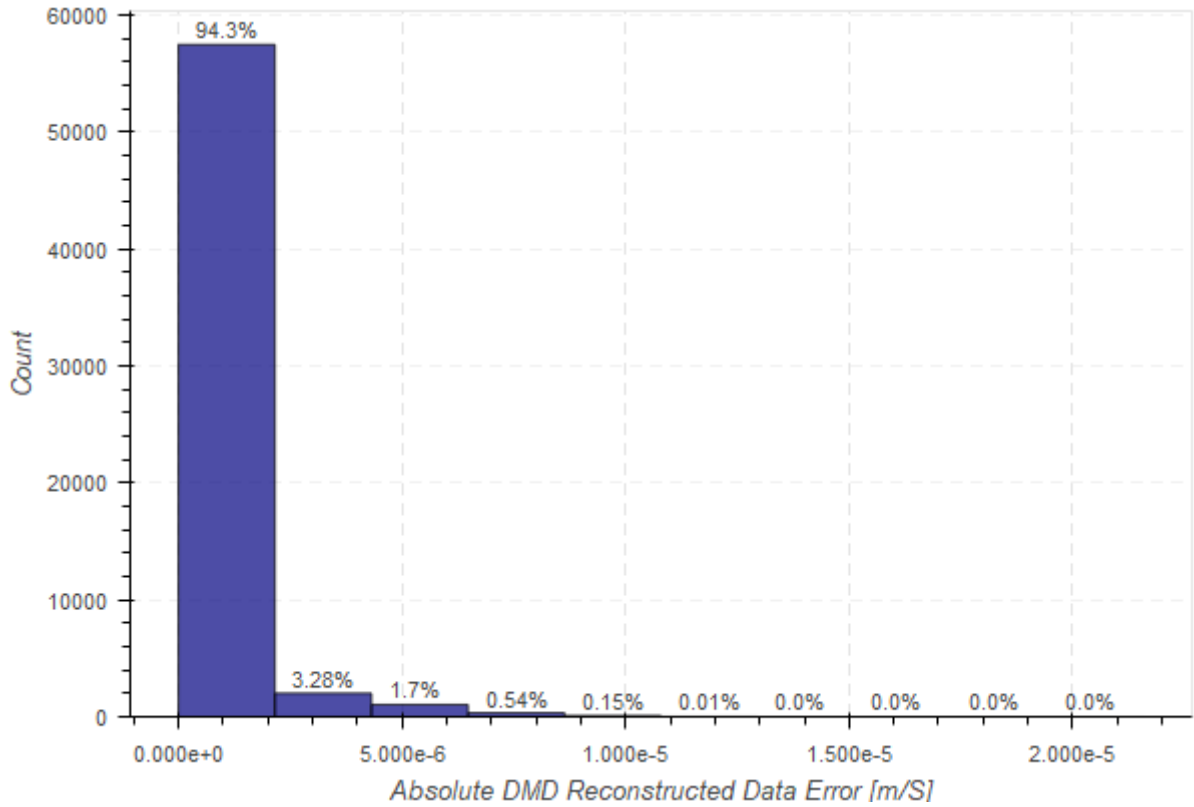


Figure 3.52 The frequency distribution of the absolute DMD reconstructed data error, corrected merged interval No. 3 (0 – 156)

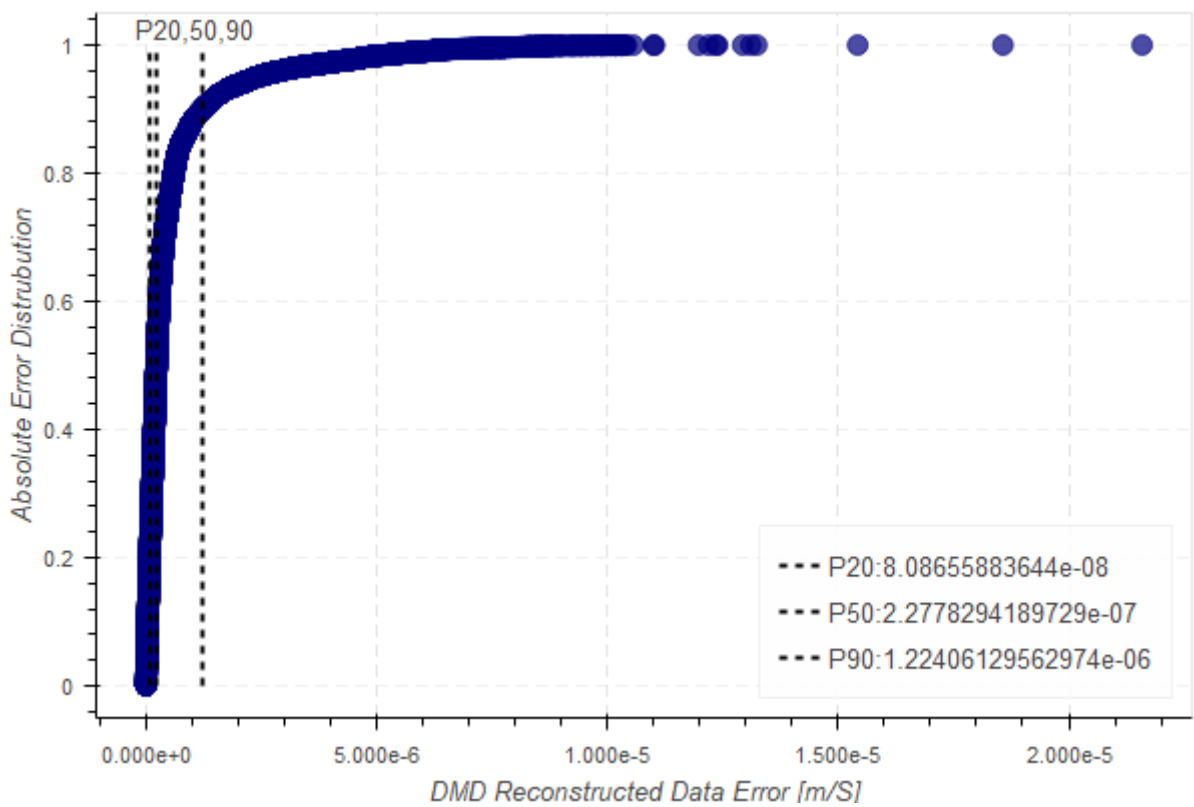


Figure 3.53 The distribution of the absolute DMD reconstructed data error, corrected merged interval No. 3 (0 - 156)

As can be seen in Figure 3.52 and Figure 3.53, the performance of DMD reconstruction is satisfactory and P90 (1.2e-06) is an indication of this performance. The performance of DMD algorithm for reconstruction of the merged intervals are summarized in Table 3.

Table 3. Error distribution of DMD reconstruction data, merged intervals

Merged interval No.	Interval range	P20	P50	P90
1	0 – 111	4.6e-07	1.6e-06	7.9e-06
2	0 – 140	5.2e-07	1.3e-06	3.9e-06
3	0 – 156	8.1e-08	2.3e-07	1.2e-06
4	0 – 185	4.5e-06	1.1e-05	3.5e-05
5	0 – 210	1.5e-07	4.2e-07	3.3e-06
6	0 – 229	3.9e-10	1.2e-09	6.4e-09
7	0 – 248	3.9e-10	1.2e-09	8.5e-09
8	0 – 278	7.7e-07	2.9e-06	3.4e-05
9	0 – 306	7.9e-09	2.4e-08	2.4e-07
10	0 – 334	1.5e-08	4.5e-08	2.5e-08
11	0 – 346	9.1e-08	2.5e-07	7.9e-07
12	0 – 368	0.002	0.02	0.06

By separating different dynamics based on the stability of eigenvalues, DMD algorithm could manage to reconstruct most of the merged intervals successfully. As can be seen in Table 3, P20, P50 and P90 for intervals No. 1 to 11 are satisfactory. As length of the interval increases, it becomes difficult to find stable eigenvalues for all data and as a result reconstruction fails. For interval No. 12, all eigenvalues for captured modes are not stable, as number of dynamics increased and the corresponding result in Table 3 is the best performance of DMD algorithm which is not preferable.

Comparing 3D plots of the reconstruction error in previous section indicate that the most of errors are related to acceleration interval, as they have small amplitudes than the rest of measurements. In the following section, merged intervals without acceleration measurements will be used to check reconstruction performance.

3.2.1.3 DMD reconstruction of multiple sub datasets without acceleration

According to Table 1, interval No. 1 (0 – 43) is the acceleration measurement, but after checking stable eigenvalues 0 – 44 is selected as the acceleration interval. DMD algorithm is implemented for several intervals as follows:

- Merged Interval No. 1: 45 – 140

This interval contains two dynamics, including 45 – 111, 112 – 140 (see *Figure 3.54*).

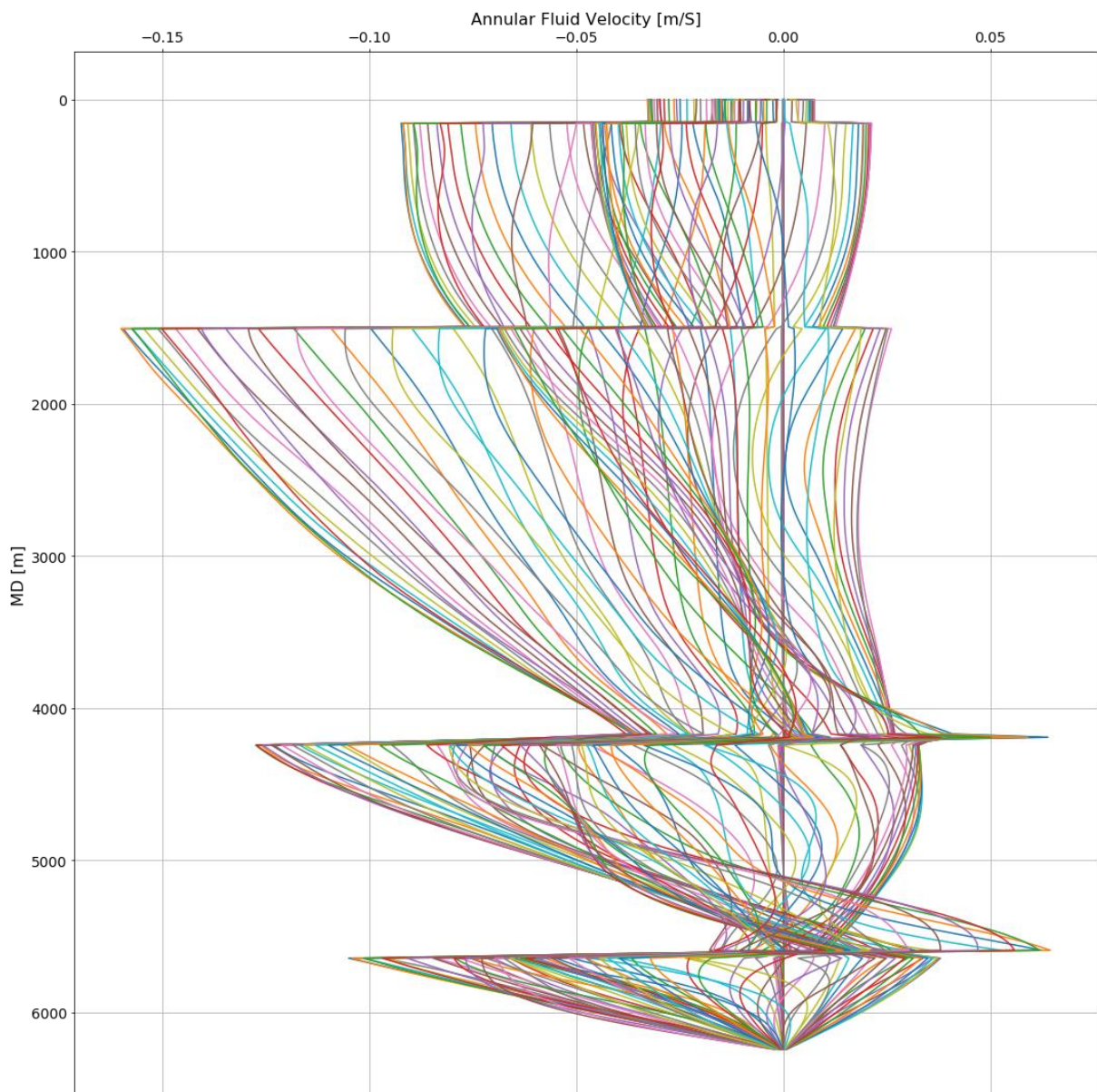


Figure 3.54 Annular fluid velocity, merged interval No. 1 (45 - 140) without acceleration data

Singular value decomposition (SVD) and Singular values of the merged interval No. 1 (without acceleration) are shown in Figure 3.55 and Figure 3.56.

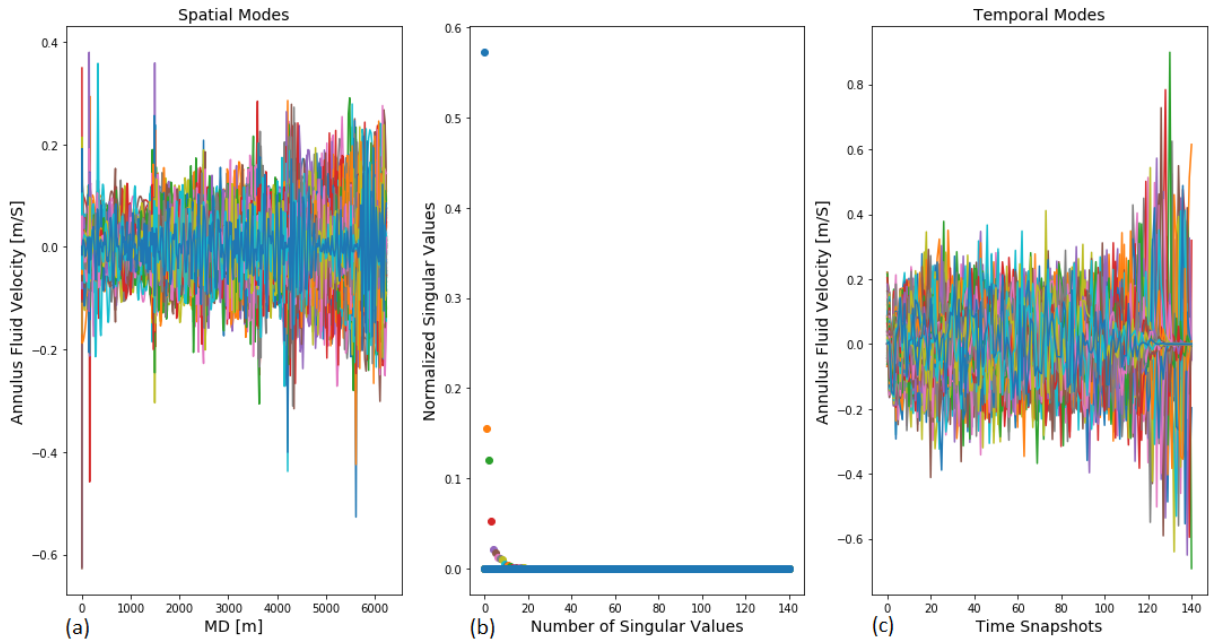


Figure 3.55 Singular value decomposition (SVD) of merged interval 1 (45 - 140) without acceleration

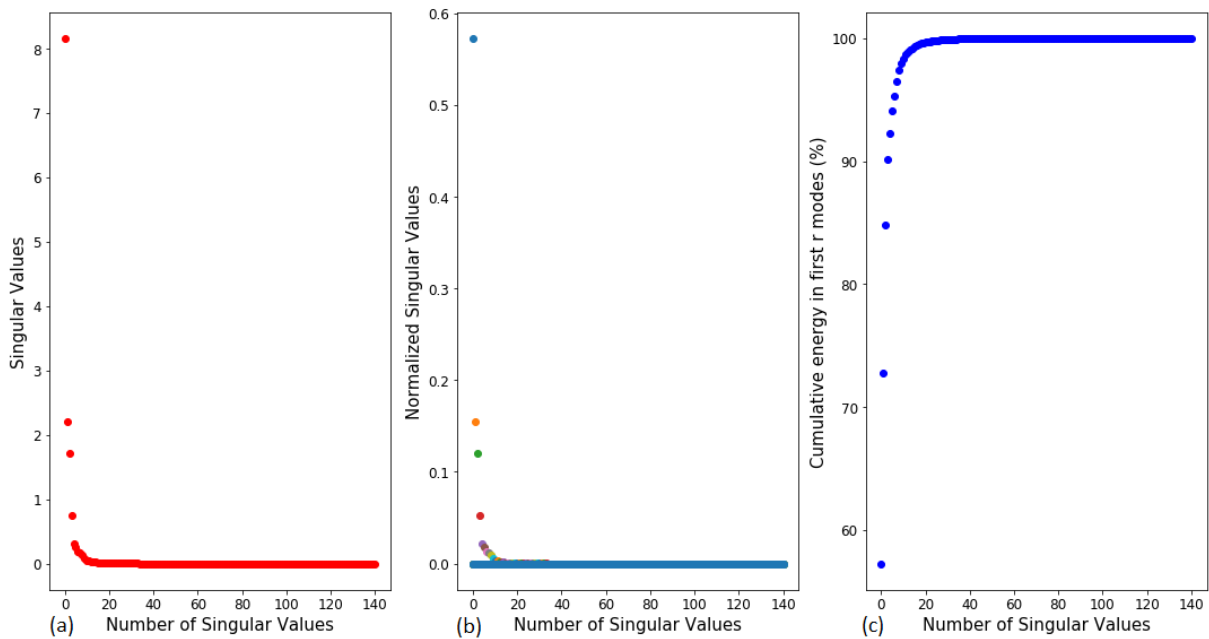


Figure 3.56 Singular values of merged interval No. 1 (45 - 140) without acceleration data

As can be seen in Figure 3.55 and Figure 3.56, the maximum energy of the system corresponding to the largest singular value is 57.26%, which compare to merged interval (0 – 140) in Figure 3.28 is reduced. Figure 3.57 shows spatial modes and their dynamics used for reconstruction.

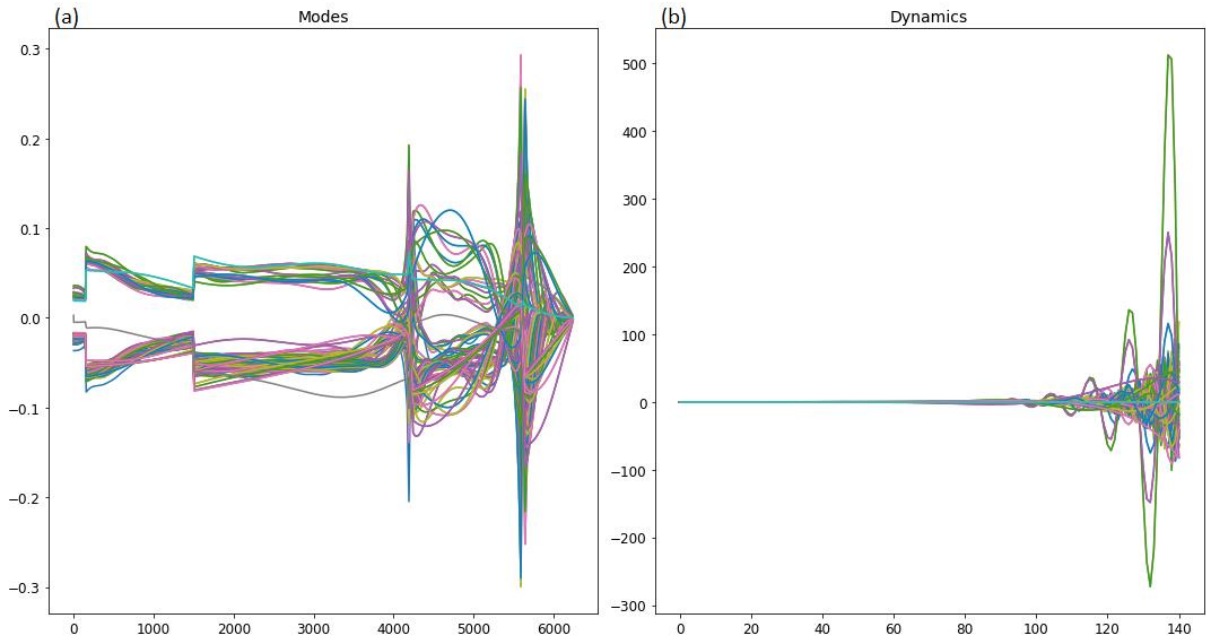


Figure 3.57 *The Spatial modes and their dynamics in time, merged interval 1 (45 - 140) without acceleration data*

As can be seen in Figure 3.57, captured modes and their dynamics are identical to previous analyses as intervals are classified based on stable eigenvalues. Eigenvalues of the merged interval (45 – 140) are shown in Figure 3.58.

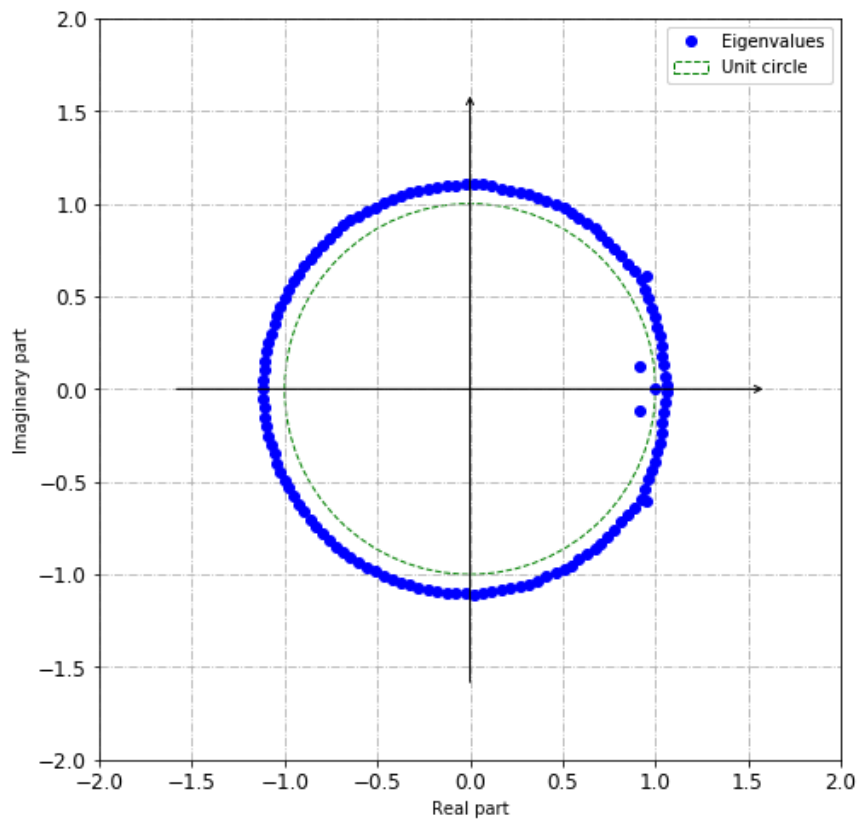


Figure 3.58 *Eigenvalues of merged interval 1 (45 - 140) without acceleration data*

By removing acceleration measurements, the corresponding eigenvalues inside unit circle are removed and stable eigenvalues achieved (see Figure 3.58). 3D plots of DMD reconstruction and original measurements (merged interval 45 – 140) are shown in Figure 3.59 and Figure 3.60.

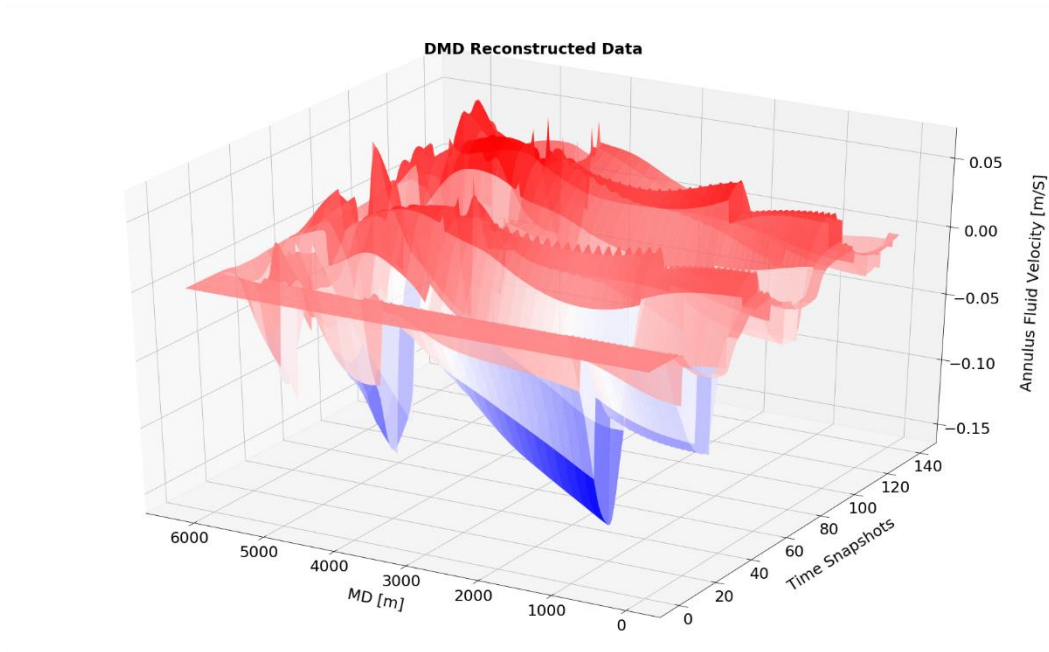


Figure 3.59 DMD reconstruction of merged interval No. 1 (45 - 140) without acceleration data

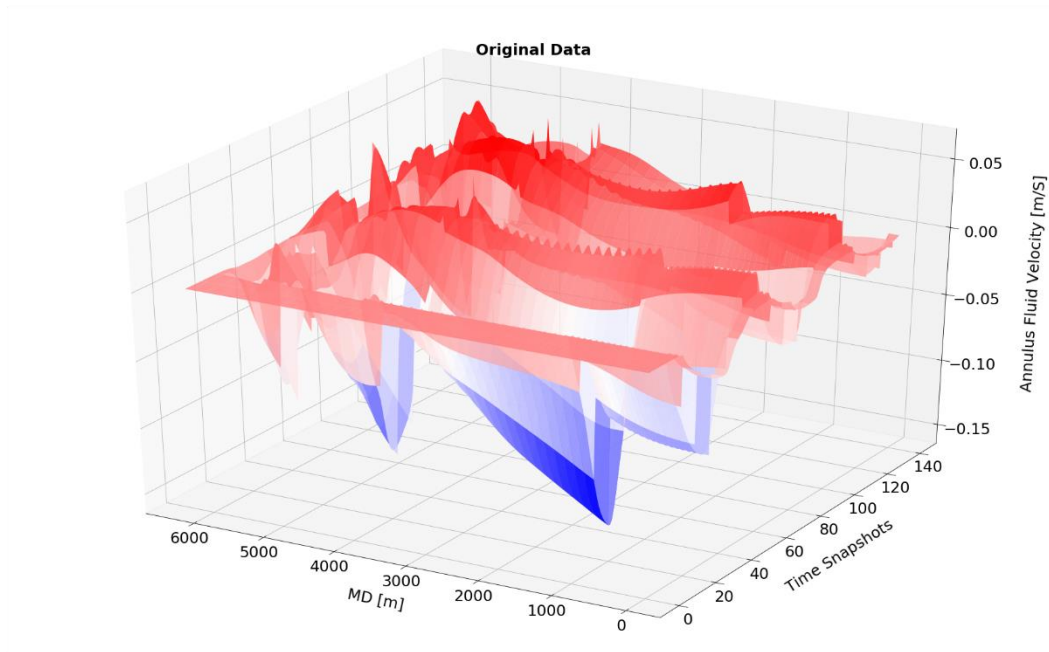


Figure 3.60 Original data, merged interval No. 1 (45 - 140) without acceleration data

As shown in Figure 3.59 and Figure 3.60, the reconstruction and original data plots are similar. To find out the accuracy of DMD reconstruction, it is better to have a look at the figures which represent error.

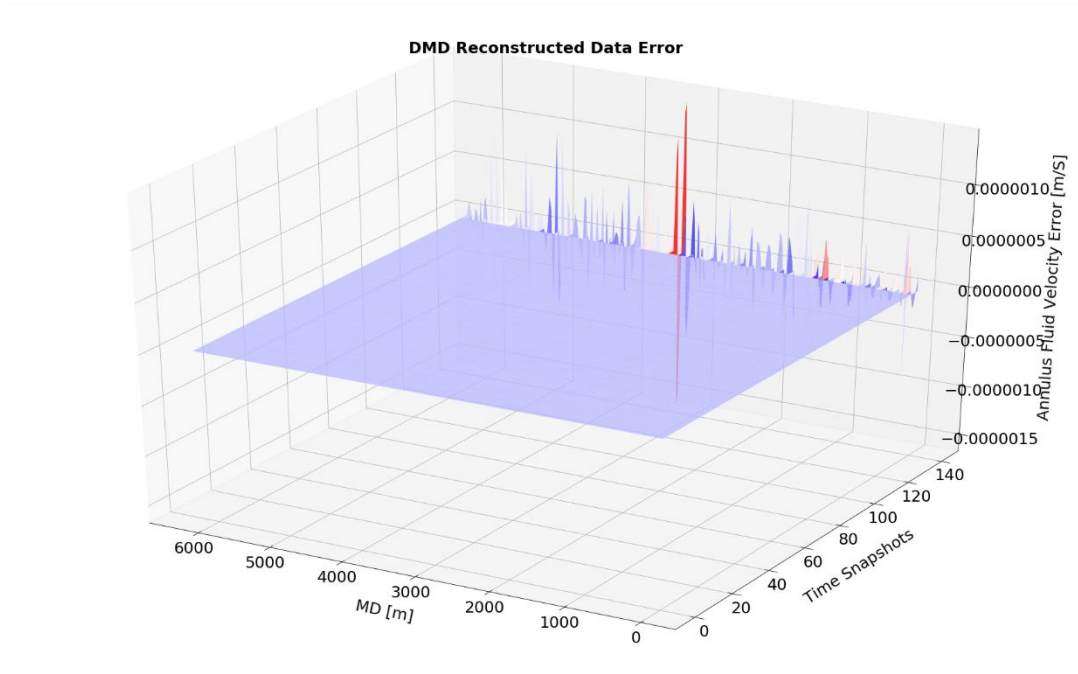


Figure 3.61 DMD reconstructed data error merged interval No. 1 (45 – 140) without acceleration data

As can be seen in Figure 3.61, error plot is mostly light blue which indicates almost zero error in reconstruction. Error distributions are shown in Figure 3.62 and Figure 3.63.

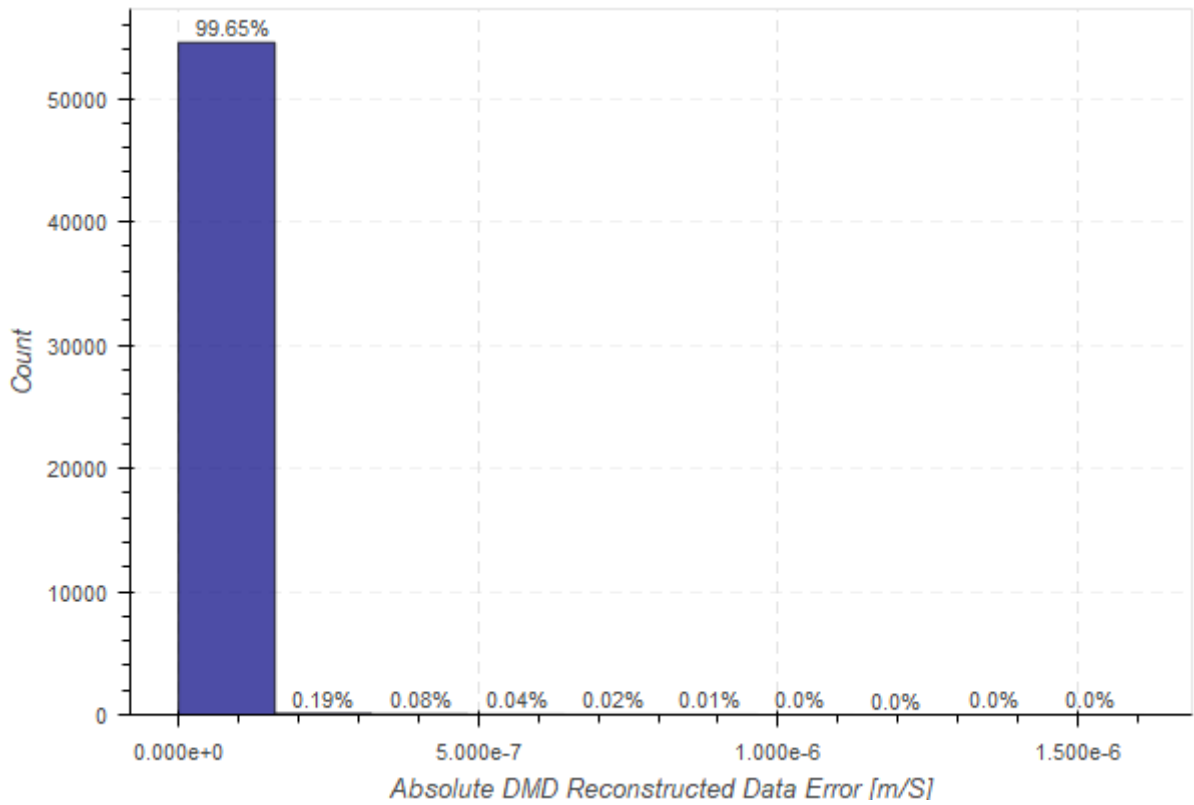


Figure 3.62 The frequency distribution of the absolute DMD reconstructed data error, merged interval No. 1 (45 – 140) without acceleration data

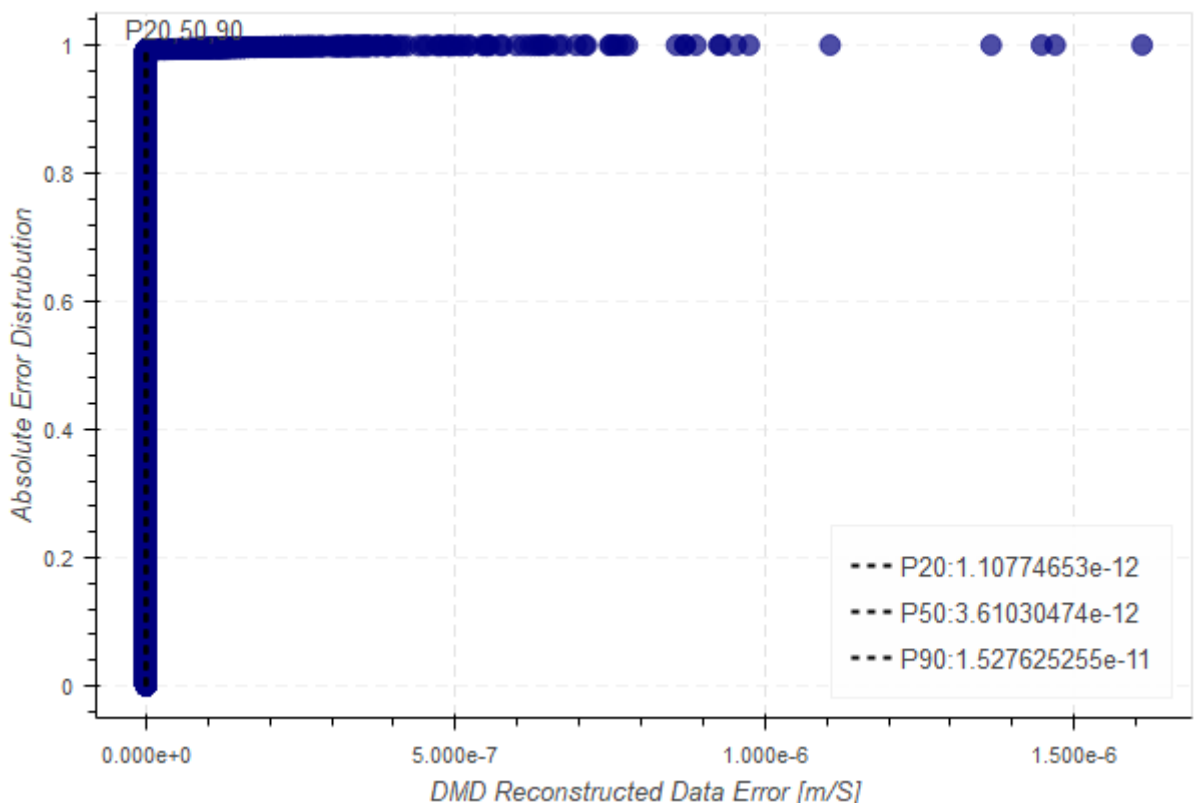


Figure 3.63 The distribution of the absolute DMD reconstructed data error, merged interval No. 1 (45 - 140) without acceleration data

By removing acceleration interval, DMD algorithm could calculate more precise initial amplitudes for modes, as amplitudes are in same ranges. Thus, better results are achieved compare to the reconstruction of merged intervals with acceleration. P90 is $1.5e-11$, which is an indication of great performance. The performance of DMD algorithm for reconstruction of the merged intervals without acceleration data are summarized in Table 4.

Table 4. Error distribution of DMD reconstruction data, merged intervals without acceleration

Merged interval No.	Interval range	P20	P50	P90
1	45 – 140	$1.1e-12$	$3.6e-12$	$1.5e-11$
2	45 – 156	$3.6e-12$	$1.1e-11$	$4.2e-11$
3	45 – 184	$2.9e-12$	$9.6e-12$	$4.0e-11$
4	45 – 203	$2.9e-11$	$8.1e-11$	$2.4e-10$
5	45 – 224	$4.7e-11$	$1.4e-10$	$5.9e-10$
6	45 – 251	$3.6e-10$	$1.1e-09$	$3.8e-09$
7	45 - 288	$1.8e-09$	$5.2e-09$	$1.7e-08$

As can be seen in Table 4, great performance is achieved by removing acceleration data. After classifying data based on similar patterns and separating different dynamics especially transitional measurements between two dynamics using stable eigenvalues, these results achieved. Moreover, after removing acceleration interval, adding or removing columns of data continued to get stable eigenvalues and this is why you may see different interval ranges compare to Table 3. As mentioned earlier, by increasing the number of timesteps and as a result number of dynamics, some eigenvalues will be located out of stability circle and for this reason, it is not possible to achieve satisfactory reconstruction beyond 288 timesteps.

Until now, attempts have been made to reconstruct the original dataset. In spite of all limitations and constraints, DMD algorithm succeeded in reconstructing number of time snapshots under certain conditions. In the following sections, interpolation and extrapolation of DMD reconstructed data are going to be analyzed.

3.2.2 DMD data interpolation

If DMD algorithm was able to reconstruct any dataset with no constraints, then it would be possible to use interpolation capability of DMD unlimitedly. For now, to check the accuracy of

DMD data interpolation, one of merged intervals without acceleration data (45 – 225) is used. One out of two measured data snapshots is used for reconstruction and missed measurements are interpolated to check with real measurements. DMD reconstruction error is shown in Figure 3.64.

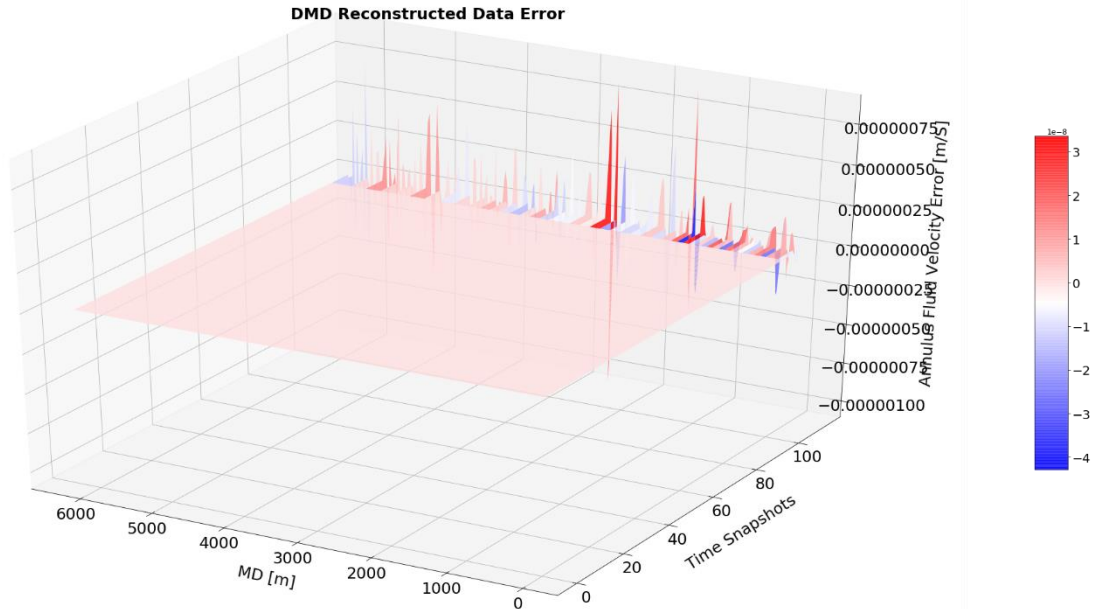


Figure 3.64 DMD reconstructed data error, (45 – 225) odd columns

As shown in Figure 3.64, DMD reconstruction error is low through the dataset except the measurements at the boundary. Error distributions are shown in Figure 3.65 and Figure 3.66.

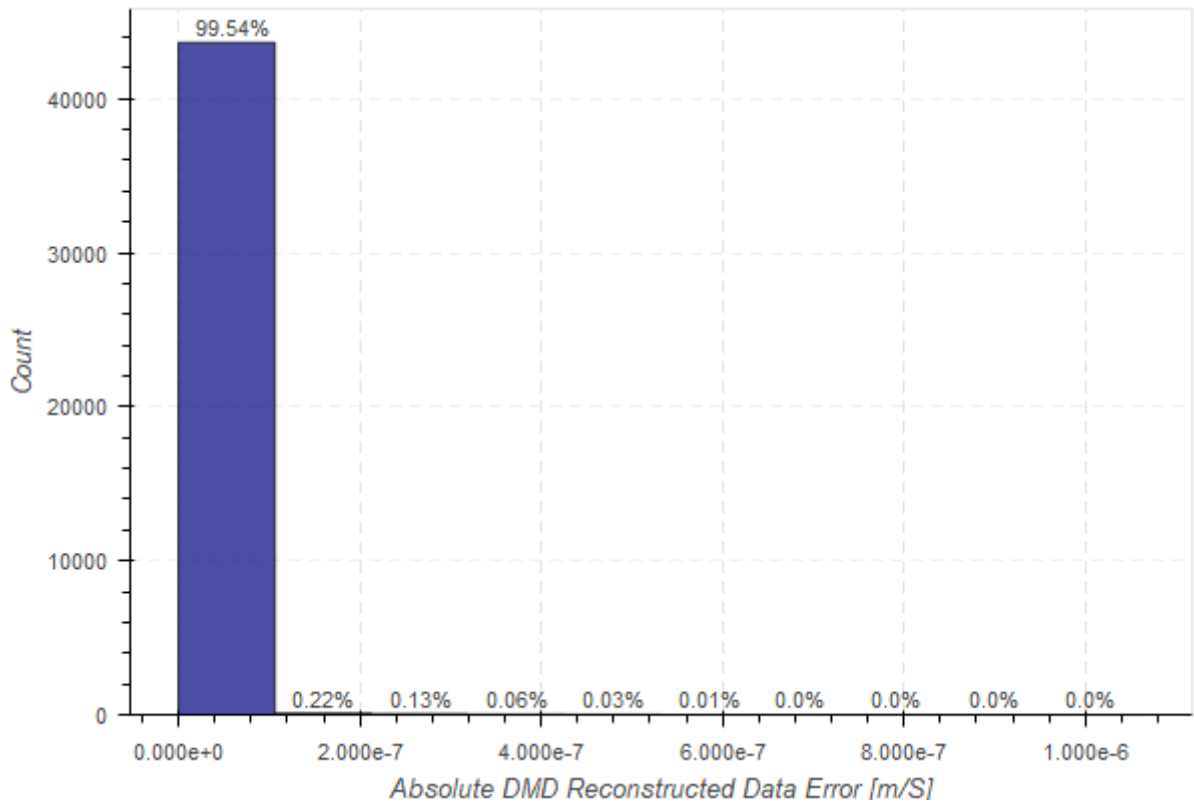


Figure 3.65 The frequency distribution of the absolute DMD reconstructed data error, (45 – 225) odd columns

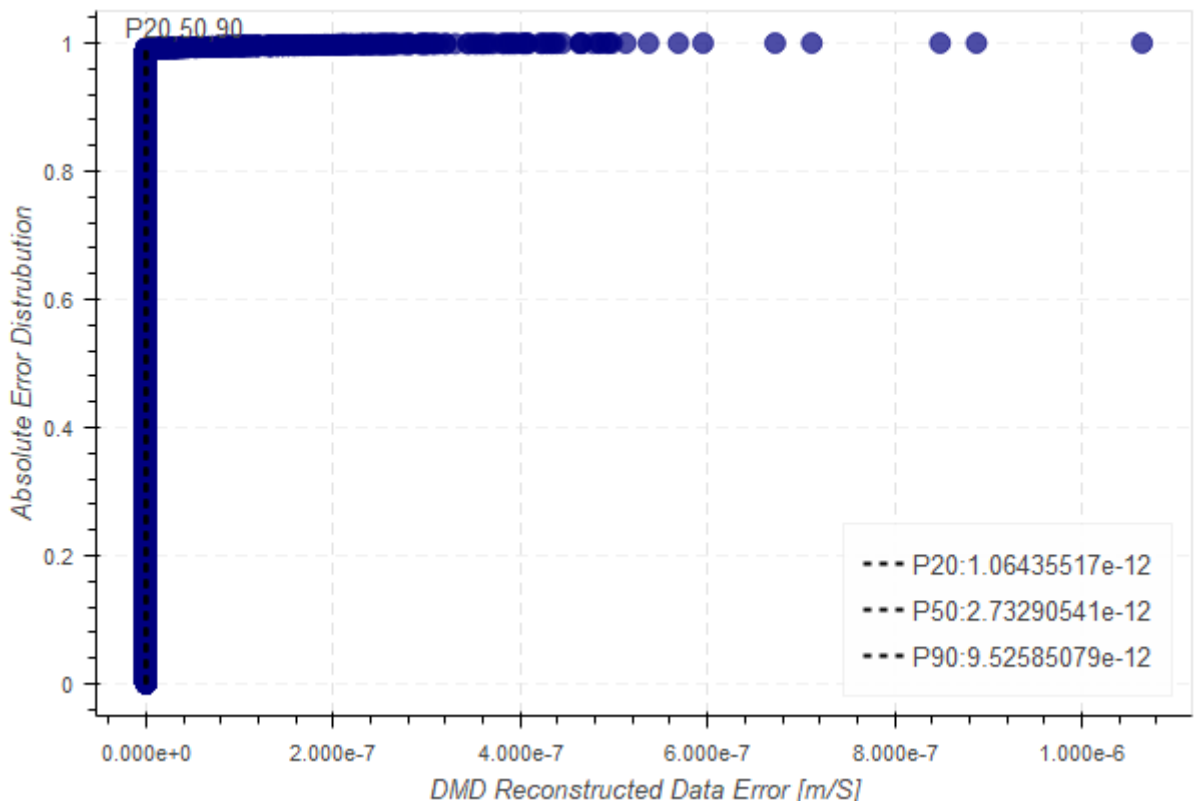


Figure 3.66 The distribution of the absolute DMD reconstructed data error, (45 - 225) odd columns

As can be seen in Figure 3.65 and Figure 3.66, the DMD performance for this interval is satisfactory and this is why it is selected for evaluating DMD data interpolation. 3D plots of DMD data reconstruction and interpolation along with corresponding original data are shown in the following.

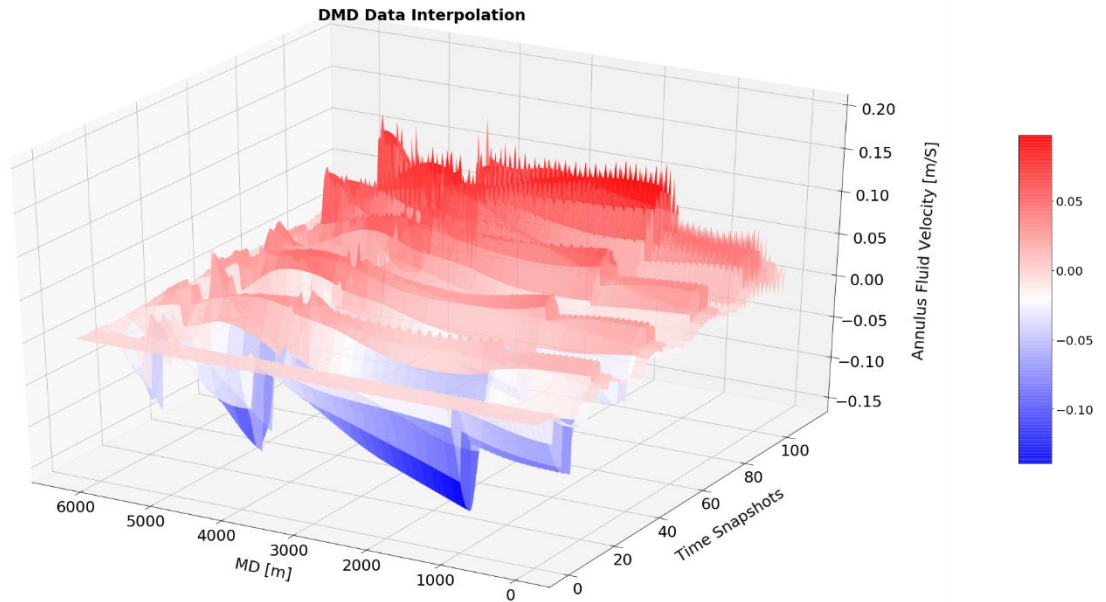


Figure 3.67 DMD reconstruction (odd time snapshots) and interpolation (even time snapshots), (45 - 225)

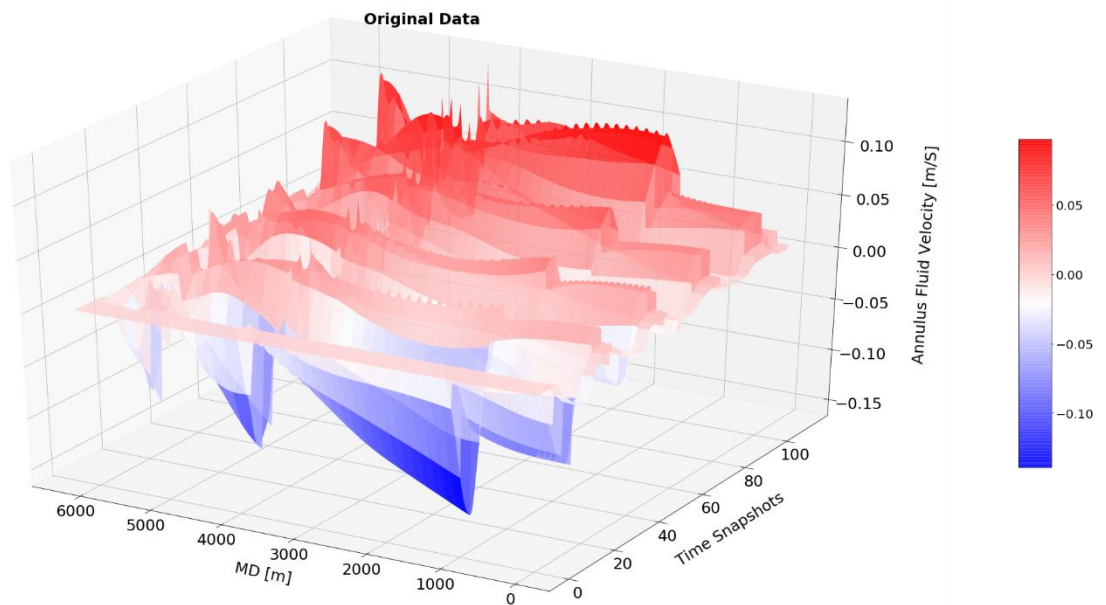


Figure 3.68 Original data, (45 - 225)

Comparing Figure 3.67 and Figure 3.68, it is clear that the interpolated and original data look similar except for last time snapshots. To find out the accuracy of DMD interpolation, it is better to have a look at the figures which represent error.

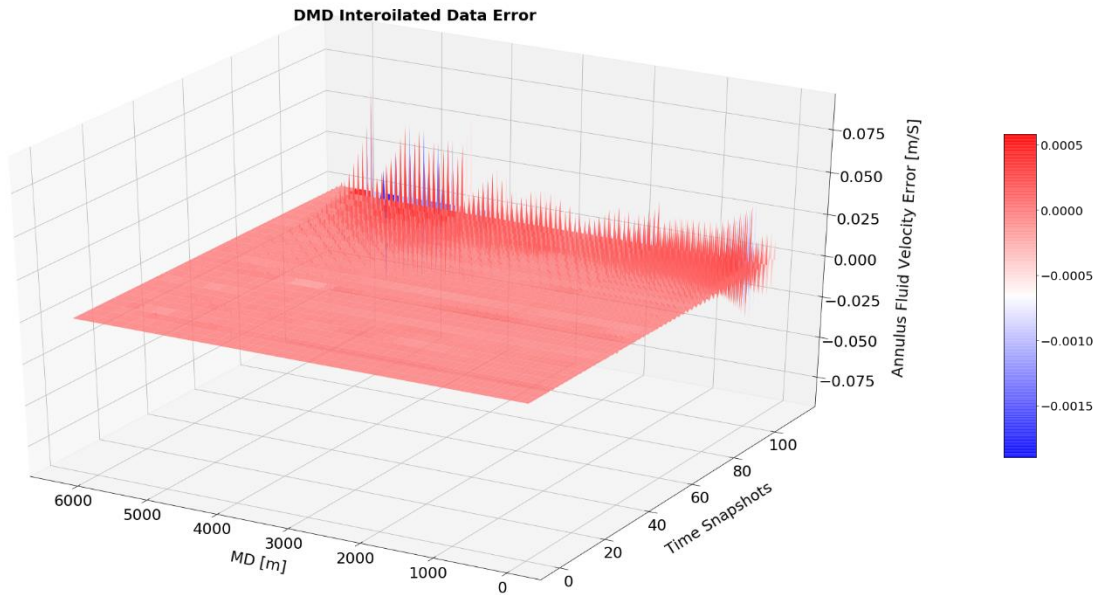


Figure 3.69 DMD interpolated data error, (45 - 225)

As expected, most of DMD interpolated errors are related to last time snapshots (see Figure 3.69). Error distributions are shown in Figure 3.70 and Figure 3.71.

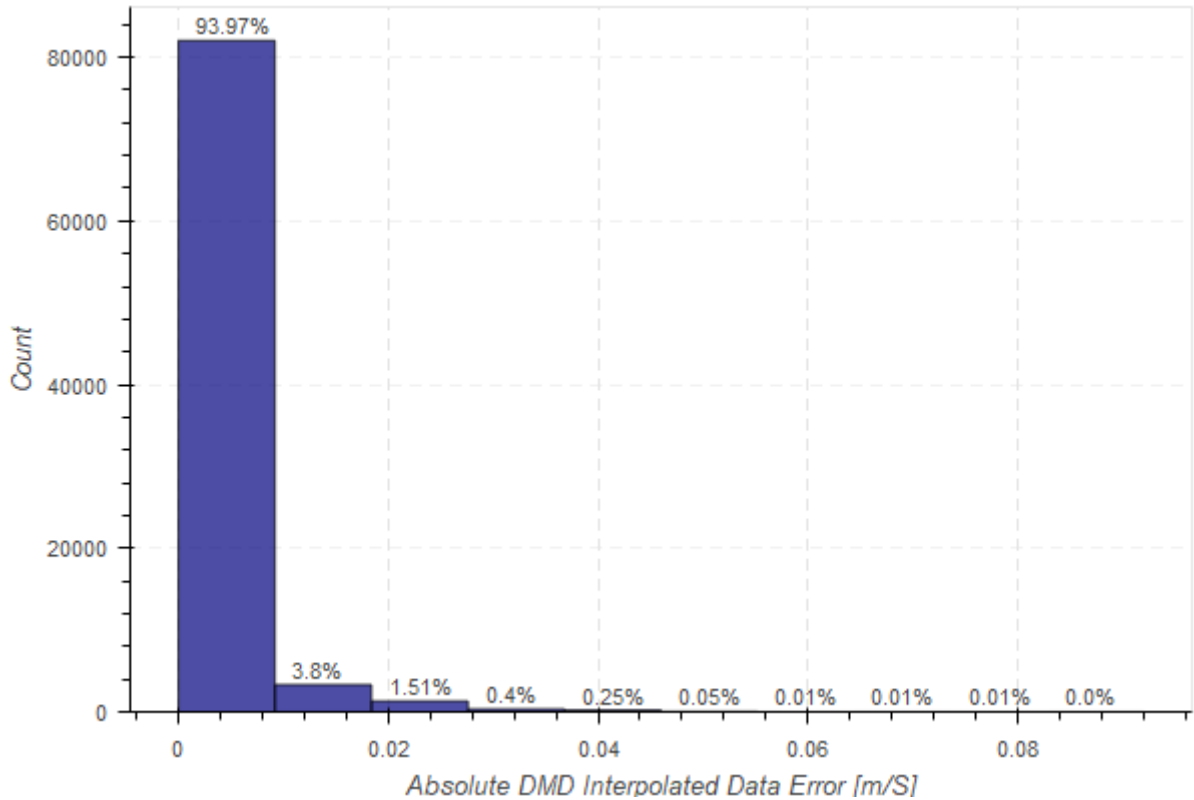


Figure 3.70 The frequency distribution of the absolute DMD interpolated data error, (45 - 225)

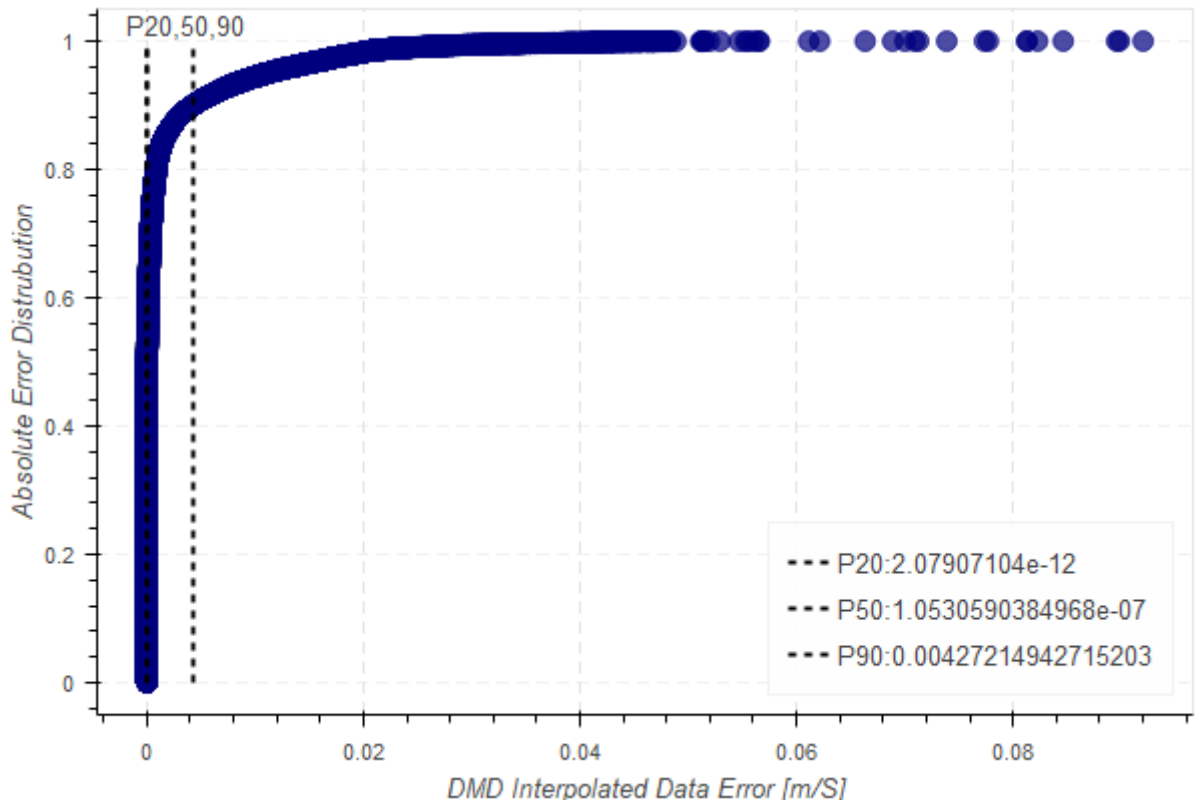


Figure 3.71 The distribution of the absolute DMD interpolated data error, (45 - 225)

As can be seen in Figure 3.70 and Figure 3.71, the performance of DMD interpolation is convincing for some of the measurements. DMD interpolation could be satisfactory as long as the performance of DMD reconstruction is preferable.

3.2.3 DMD data extrapolation or prediction

For evaluating DMD data extrapolation, merged interval (45 – 140) is going to be used as it was successfully reconstructed (see error distributions in Figure 3.62 and Figure 3.63). As mentioned earlier, most of DMD reconstruction error is at the boundary when system transits to another dynamic.

DMD algorithm is used to extrapolate 8 time-snapshots in the following. 3D plots of DMD data reconstruction and extrapolation along with corresponding original data are shown in Figure 3.72 and Figure 3.73.

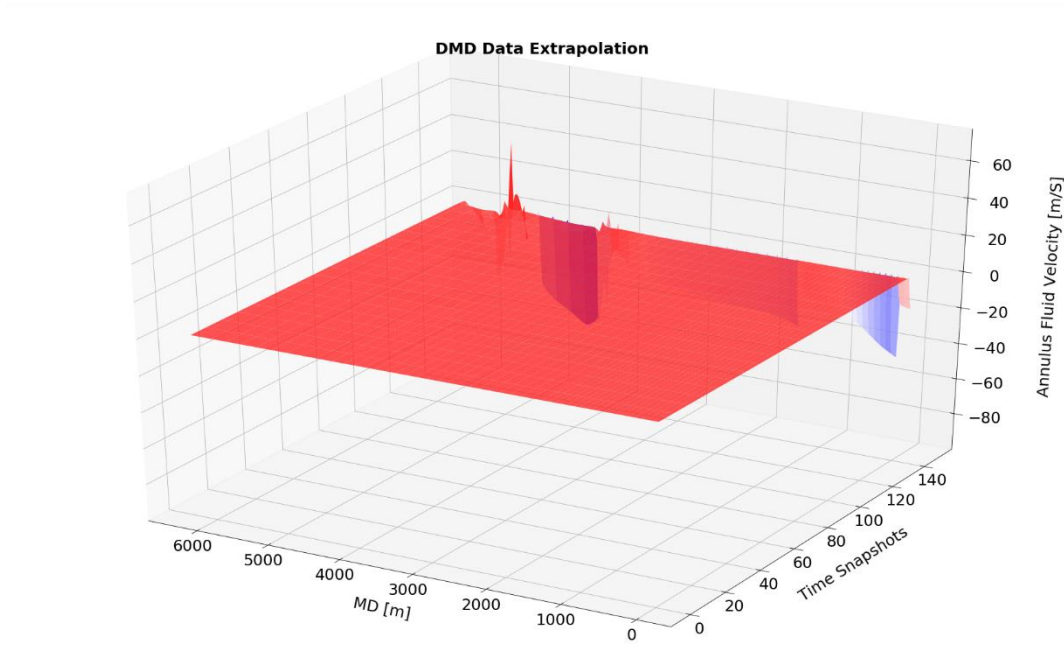


Figure 3.72 DMD reconstructed (0 - 140) and extrapolated (141 - 148) data

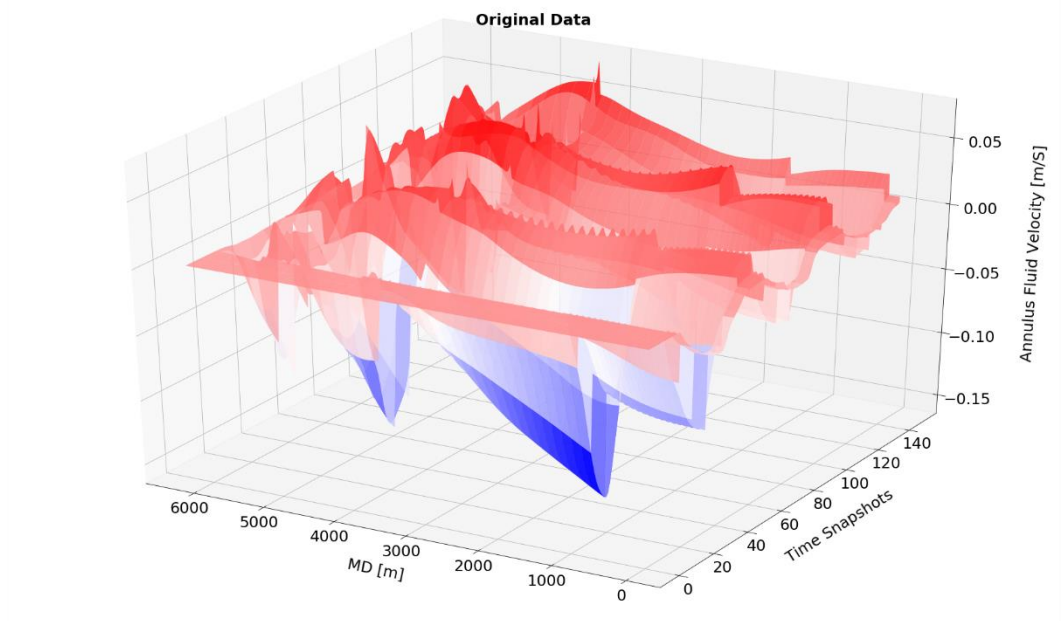


Figure 3.73 Original data interval (0 - 148)

It is obvious that Figure 3.72 and Figure 3.73 are different. Extrapolated values are so much larger than reconstructed values and this is the reason why the reconstructed values are shown like a flat plane with red color. DMD algorithm extrapolates future dynamics based on the features of the previous reconstructed dataset. According to original data, dynamic in interval (140 – 148) is almost similar to previous dynamics, thus initial amplitudes for previous dynamics should work acceptable for the extrapolated interval. Another important factor to be

considered is the stability of eigenvalues. Figure 3.74 shows the location of eigenvalues used for reconstruction of dataset.

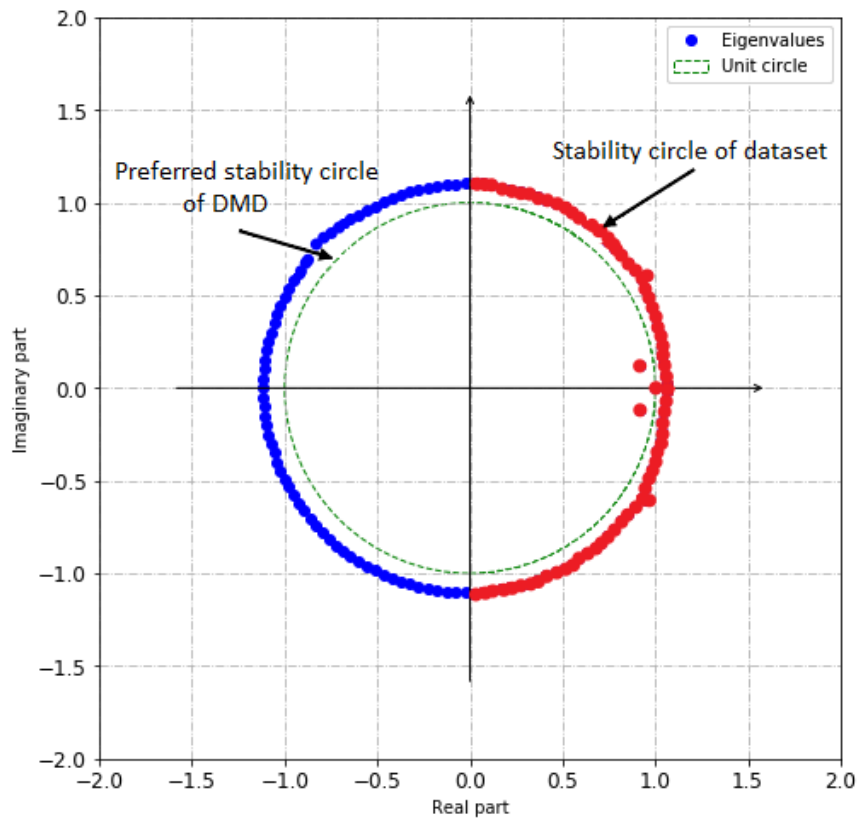


Figure 3.74 Eigenvalues of reconstructed dataset (45 – 140) for extrapolation

Figure 3.74 shows the unit circle and eigenvalues of dataset (45 – 140) with red and blue colors. For successful reconstructing of the dataset, eigenvalues are stabilized as a circle outside of unit circle. Recall that red and blue eigenvalues diverge and converge in time, respectively. As the eigenvalue circle is larger than the unit circle, capture modes change faster and this is critical for extrapolation. By extrapolating more timesteps, diverging eigenvalues oscillate with higher amplitudes and as a result the extrapolated data become far from original data. This is why unit circle is more preferable for DMD prediction which allows for more valid extrapolation in time. Error distributions of extrapolated data are shown in Figure 3.75 and Figure 3.76.

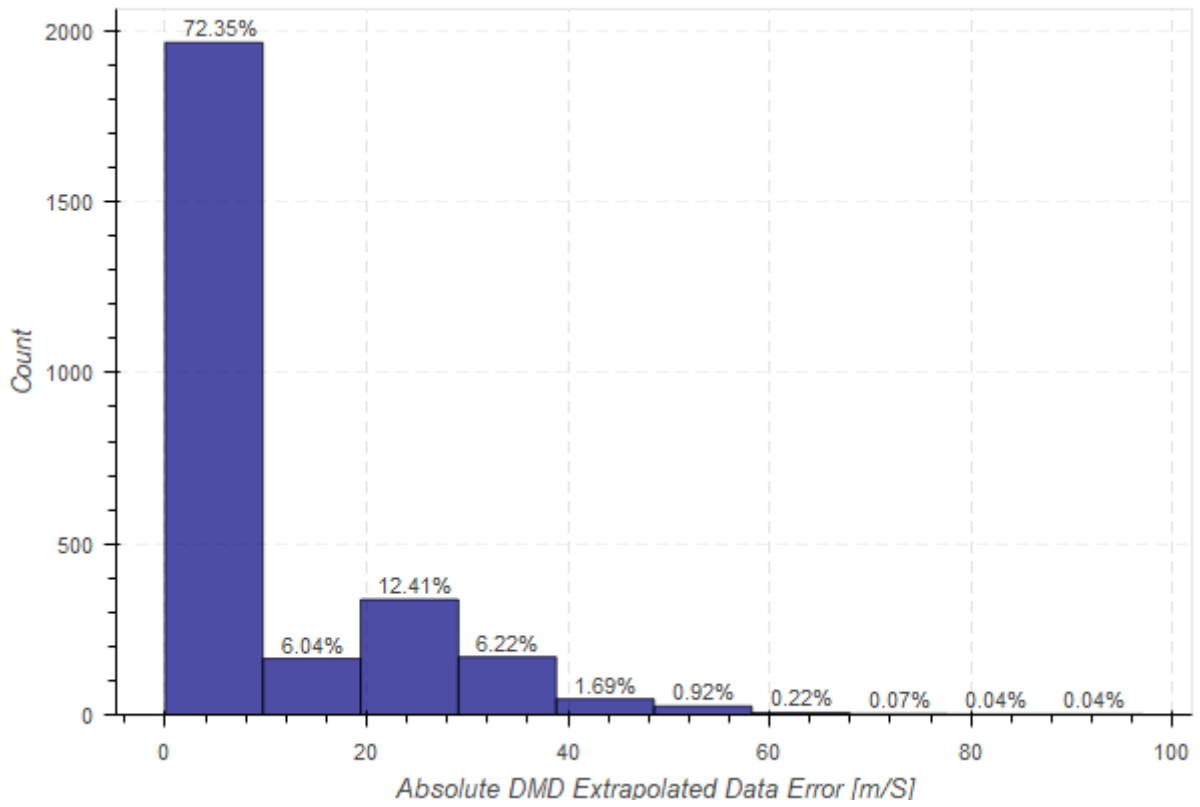


Figure 3.75 The frequency distribution of the absolute DMD extrapolated data error

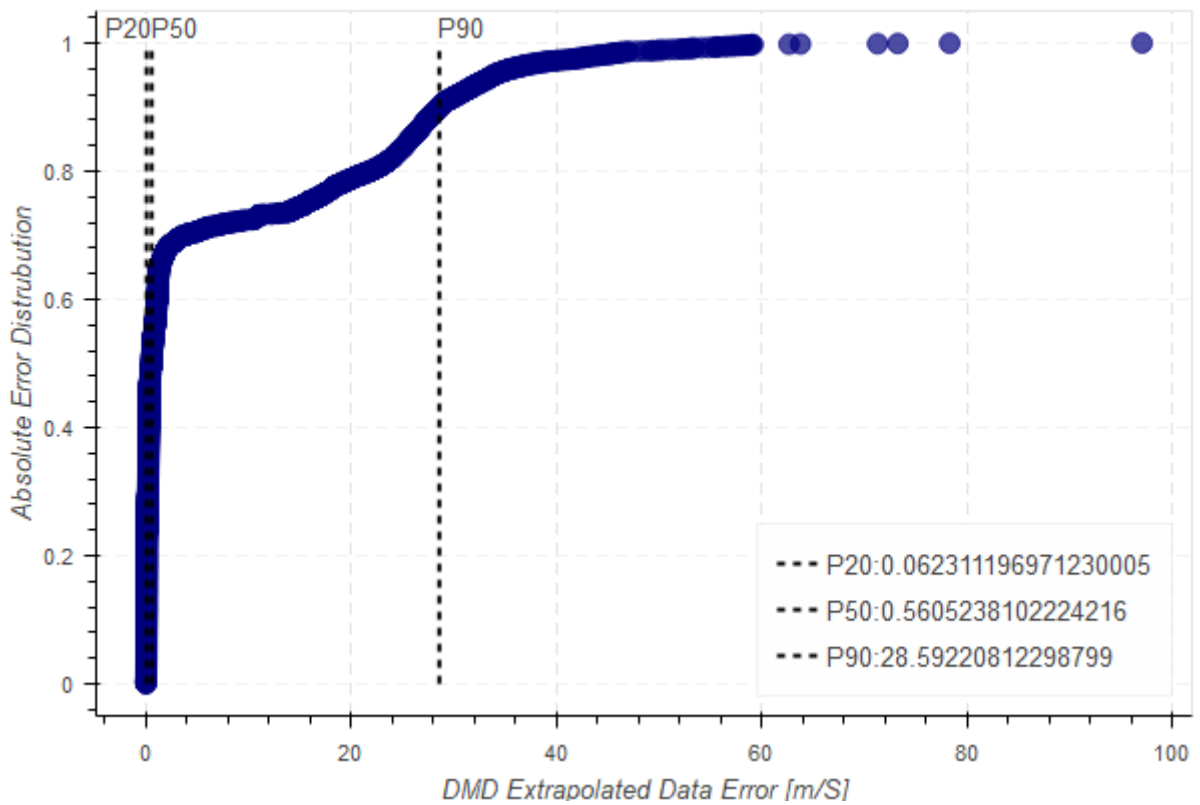


Figure 3.76 The distribution of the absolute DMD extrapolated data error

Figure 3.75 and Figure 3.76 show poor extrapolation of DMD algorithm for this dataset. P90 is 28.592208 which is unacceptable.

3.2.4 Summary of DMD implementation in AFV case study

Annular fluid velocity dataset was used to characterize the basic DMD algorithm capabilities. Among the analyzed DMD applications were DMD data reconstruction, DMD data interpolation and DMD data prediction. The main objective has been the reconstruction of whole dataset because it is the base of all applications of DMD algorithm. Unfortunately, basic DMD algorithm cannot reconstruct different datasets without limitations. Under various conditions, basic DMD algorithm was implemented and its pros and cons were evaluated and good understandings about the method were achieved. Furthermore, based on the successful reconstructed intervals, short studies of DMD data interpolation and extrapolation were accomplished, aiming to learn more about the algorithm capabilities. In the following chapter, the main findings of this master thesis will be summarized.

3.3 Future studies

As explained earlier, this master thesis was just the tip of iceberg and the journey toward more advanced and complicated DMD algorithm started. There are a lot of limitations and unknowns about DMD applications which need to be overcome. Future directions of this work could be extended as implementing other DMD extensions, such as multiresolution DMD, compressed DMD, forward backward DMD, higher order DMD and so forth.

3.4 DMD applicability and current technologies in the industry

One of the utmost desires of the drilling industry is full drilling automation in order to minimize well construction cost, optimize well performance and so many other advantages. DMD is one of data-driven techniques which is developing and could be used in digital drilling technology. The method relies on high quality and high-density data and this turned into reality by introduction of wired drill pipe. Many companies are working on different WDP technologies, as benefits and impacts of WDP has been recognized through field tests, in order to improve the reliability and reduce total cost of ownership which are not the subject of this section (Silvester, Høgset, Torvund, & Saxena, 2020). One of the important factors for applying DMD algorithm is the number of measurements in each timestep which is corresponding to

number of sensors along drill sting. Among the technologies for WDP in the market are IntelliServ wired drill pipe from National Oilwell Varco Company, Powerline drill Sting (PDS) from TDE Group, DualLink drill string from Reelwell AS and Micro-repeater wired pipe or smart pipe from Baker Hughes. According to smart pipe technology, there is one sensor per each pipe joint for real-time measurements (Macpherson, Rodgers, Schoenborn, Mieting, & Lopez, 2019). Regarding IntelliServ, PDS and DualLink, no published document has been found about the maximum number of sensors which can be used along wired drill-sting. However, using a lot of ASMs could be limited by their costs as ASMs are separate tool joints. It seems that smart pipe technology fulfils the DMD algorithm requirements.

4 Conclusions and Recommendations

A key contribution of this master thesis is the introduction of DMD theoretical framework based on linear algebra and applications of DMD algorithm to a drilling dynamic dataset in order to understand the potential benefits and drawbacks of DMD algorithm. The main findings and conclusions of the work are summarized as follows:

- DMD algorithm is easy to formulate and it can be used to explore diagnostic features of data like eigenvalues where their imaginary parts correspond to the underlying frequencies of the system
- DMD algorithm is able to separate independent dynamics of AFV dataset using eigenvalues corresponding to stable modes
- Transition from one dynamic to another is not well captured by the algorithm in such a way that even one measurement from another dynamic can ruin the whole reconstruction.
- DMD algorithm succeeded in reconstructing number of time snapshots under certain constraints and conditions, however our desire is to reconstruct whole dataset, no matter what dynamics exist in it.
- When there are dynamics with different ranges of amplitudes, optimal amplitudes measured by algorithm could be overestimated, underestimated or equal, compare to real values. For this reason, acceleration interval is overestimated. By removing acceleration interval, DMD algorithm could calculate more precise initial amplitudes for modes, as amplitudes are in same ranges. Thus, better results are achieved compare to the reconstruction of merged intervals with acceleration.
- By increasing the number of timesteps in a dataset and as a result number of dynamics, some eigenvalues will be located out of stability circle and for this reason, it is not possible to achieve satisfactory reconstruction beyond 288 timesteps.
- DMD interpolation could be satisfactory as long as the performance of the DMD reconstruction is preferable.
- The AFV during swabbing is so non-linear, that it limits very much the applicability of the DMD for extrapolation.
- In successful reconstruction of AFV datasets, the eigenvalues were stabilized at short distance out of unit circle corresponding to stable modes. As the eigenvalue circle is larger than the unit circle, capture modes change faster and this is critical for extrapolation. By extrapolating more timesteps, diverging eigenvalues oscillate with higher amplitudes and as a result the extrapolated data become far from original data. This is why unit circle is

more preferable for DMD prediction which allows for more valid extrapolation in time.

- Many studies are necessary to develop DMD algorithm and overcome limitations. Moreover, for commercializing DMD algorithm, WDP providers should increase the number of sensors run on WDP to transmit high quality and high-density real-time data.

Appendix A – Eigenvalue and eigenvector

Eigenvalue problems often arise from differential equations which are impossible to solve by elimination. By considering the linear differential equation $\frac{dy}{dt} = Ay$ and attempting a solution of the form $y(t) = \phi \exp(\lambda t)$, the resulting equation for ϕ is:

$$A\phi = \lambda\phi \quad (\text{App. A.1})$$

which is eigenvalue problems. The ϕ and λ denote eigenvector and eigenvalue respectively. Almost all vectors change direction when they are multiplied by A. There are some exceptional vectors ϕ which are in the same direction as $A\phi$ and called eigenvectors. When matrix A is multiplied by an eigenvector ϕ , the resulting vector $A\phi$ is a number λ times the original ϕ . In other words, the eigenvalue λ is an indication which represents whether the vector ϕ is stretched or shrunk or reversed or left unchanged when it is multiplied by A.

Computing eigenvalue and eigenvector:

- Rewriting the *App. A.1* as: $A\phi_i = \lambda_i I \phi_i$ i : number of eigenvalue or eigenvector
- Moving $\lambda I \phi$ to the left side of equation: $A\phi_i - \lambda_i I \phi_i = 0$
- Factoring out the vector ϕ : $(A - \lambda_i I)\phi_i = 0$
- Computing eigenvalues knowing that λ is an eigenvalue of A if and only if $A - \lambda I$ is singular, which means:

$$\det(A - \lambda_i I) = 0$$

- For each eigenvalue λ , solve $(A - \lambda I)\phi = 0$ to find an eigenvector ϕ
- When all eigenvalues and eigenvectors of A are computed, it is possible to collect them in matrix system:

$$A\phi_1 = \lambda_1\phi_1$$

$$A\phi_2 = \lambda_2\phi_2$$

⋮

$$A\phi_n = \lambda_n\phi_n$$

$$A\Phi = \Phi\lambda$$

which columns of matrix Φ are eigenvectors of A and λ is a matrix whose diagonals are eigenvalues of A. (Kutz, 2013; Strang, 2016)⁵

⁵ These references are used to explain mathematical concepts for Appendix A

Appendix B – Singular value decomposition (SVD)

The singular value decomposition (SVD) is one of the most important matrix factorizations for various computational methods. Data that are generally generated from complex systems are in form matrices which are usually low-rank, meaning that there are a few patterns that describe system dynamics. The SVD is an efficient method of extracting dominant patterns purely from high-dimensional data to determine a low-dimensional approximation. The SVD is numerically stable and guaranteed to exist and as a result provide a new coordinate system to represent the data according to the dominant correlations within the data. In addition to dimensionality reduction, the SVD is used to compute the pseudo-inverse of non-square matrices to find solutions of overdetermined or underdetermined system of equations, $Ax=b$.

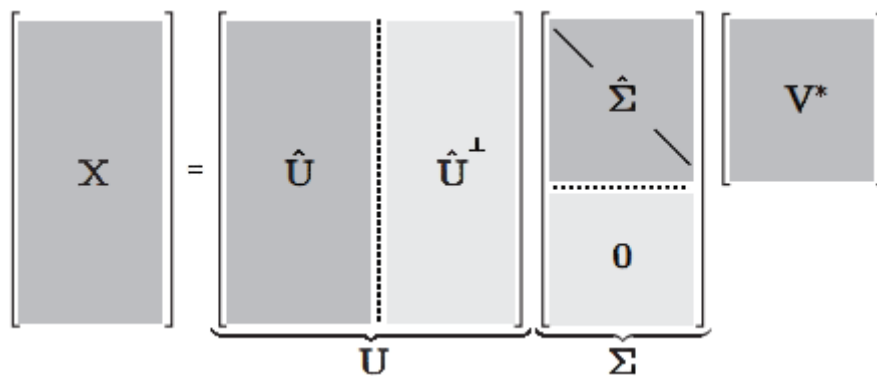
The SVD is a unique matrix decomposition that exists for every matrix $X \in \mathbb{C}^{n \times m}$:

$$X = U\Sigma V^* \quad (\text{App. B.1})$$

where $U \in \mathbb{C}^{n \times m}$ left singular vectors and $V \in \mathbb{C}^{m \times m}$ right singular vectors are unitary matrices (square matrix U is unitary if $U^*U=UU^*=I$), and $\Sigma \in \mathbb{C}^{n \times m}$ singular values of X is a matrix with real, nonnegative and ordered number on diagonal and zeros off the diagonal. $*$ denotes the complex conjugate transpose (for real-valued matrix, it is as regular transpose $X^* = X^T$). For overdetermined system ($n \geq m$), the matrix Σ has the most m nonzero elements on the diagonal, so full SVD (App. B.1) could be written as economy SVD to represent X :

$$X = U\Sigma V^* = [\hat{U} \quad \hat{U}^\perp] \begin{bmatrix} \hat{\Sigma} \\ 0 \end{bmatrix} V^* = \hat{U}\hat{\Sigma}V^* \quad (\text{App. B.2})$$

where columns of \hat{U}^\perp span a vector space the is complementary and orthogonal to that spanned by \hat{U} .



Append. B.1 Full singular value decomposition (SVD)((Brunton & Kutz, 2019))

$$\begin{bmatrix} \mathbf{X} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{U}} \end{bmatrix} \begin{bmatrix} \hat{\Sigma} \end{bmatrix} \begin{bmatrix} \mathbf{V}^* \end{bmatrix}$$

Append. B.2 Economy singular value decomposition (SVD)(Brunton & Kutz, 2019)

For high dimensional data, optimal rank-r approximation of the measured data could be used by truncating rank-r SVD which is obtained by keeping the leading r singular values and vectors and discarding the rest. The truncated SVD provides a coordinate transformation from high dimensional data space into a low dimensional data space. By taking SVD and plotting singular values it is possible to find out the energy of each mode and truncate matrix accordingly which compress the measured data matrix (Brunton & Kutz, 2019).

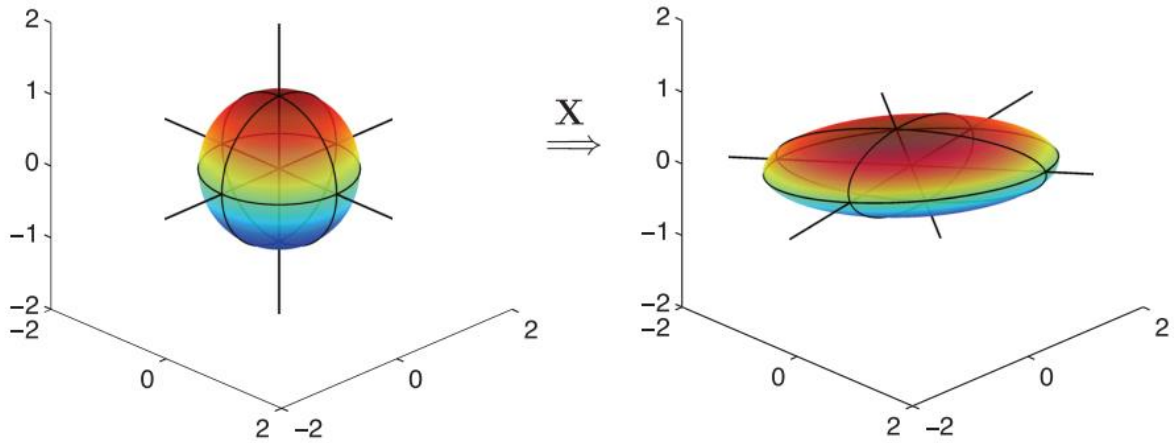
$$\begin{bmatrix} \mathbf{X} \end{bmatrix} = \underbrace{\begin{bmatrix} \tilde{\mathbf{U}} & \hat{\mathbf{U}}_{\text{rem}} & \hat{\mathbf{U}}^{\perp} \end{bmatrix}}_{\mathbf{U}} \begin{bmatrix} \tilde{\Sigma} & & \\ & \hat{\Sigma}_{\text{rem}} & \\ & & \mathbf{0} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{V}}^* \\ \mathbf{V}_{\text{rem}} \end{bmatrix}$$

Truncated SVD

$$\approx \begin{bmatrix} \tilde{\mathbf{U}} \end{bmatrix} \begin{bmatrix} \tilde{\Sigma} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{V}}^* \end{bmatrix}$$

Append. B.3 Truncated Singular Value Decomposition (SVD)(Brunton & Kutz, 2019)

If measured data matrix X contains spatial measurements in time, U encode spatial patterns and V encode temporal patterns. The SVD of X could be interpreted geometrically by linear mapping of hypersphere into an ellipsoid which U is a rotation factor and Σ is stretching factor(Brunton & Kutz, 2019; Kutz, 2013)⁶.



Append. B. 4 *Geometric illustration of the SVD as mapping from a sphere to an ellipsoid (Brunton & Kutz, 2019)*

⁶ These references are used to explain mathematical concepts for Appendix B

Appendix C – Overdetermined and underdetermined system

In mathematics, the number of constraints (independent equations) is counted in order to compare it with the number of variables, parameters, etc. which can be seen as a degree of freedom. If there are fewer equations than unknowns, the system of equations is considered underdetermined. By contrast the overdetermined system has more equations than unknowns. When the number of equations and number of variables are equal, for every variable giving a degree of freedom there is a corresponding constraint. An underdetermined linear system has either no solution or infinitely many solutions. In case a system of equations having no solution is said to be inconsistent. On the other hand, a system of equations is consistent when it has an infinite number of solutions. An overdetermined system may have solutions for some cases for example some equations are linear combinations of the others.

For example:

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \quad (\text{App. C.1})$$

which $A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$ is coefficient matrix, $B = \begin{bmatrix} x \\ y \end{bmatrix}$ is variable matrix, $C = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$ is constant matrix

and $D = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ is augmented matrix by adding constant matrix as a column in coefficient

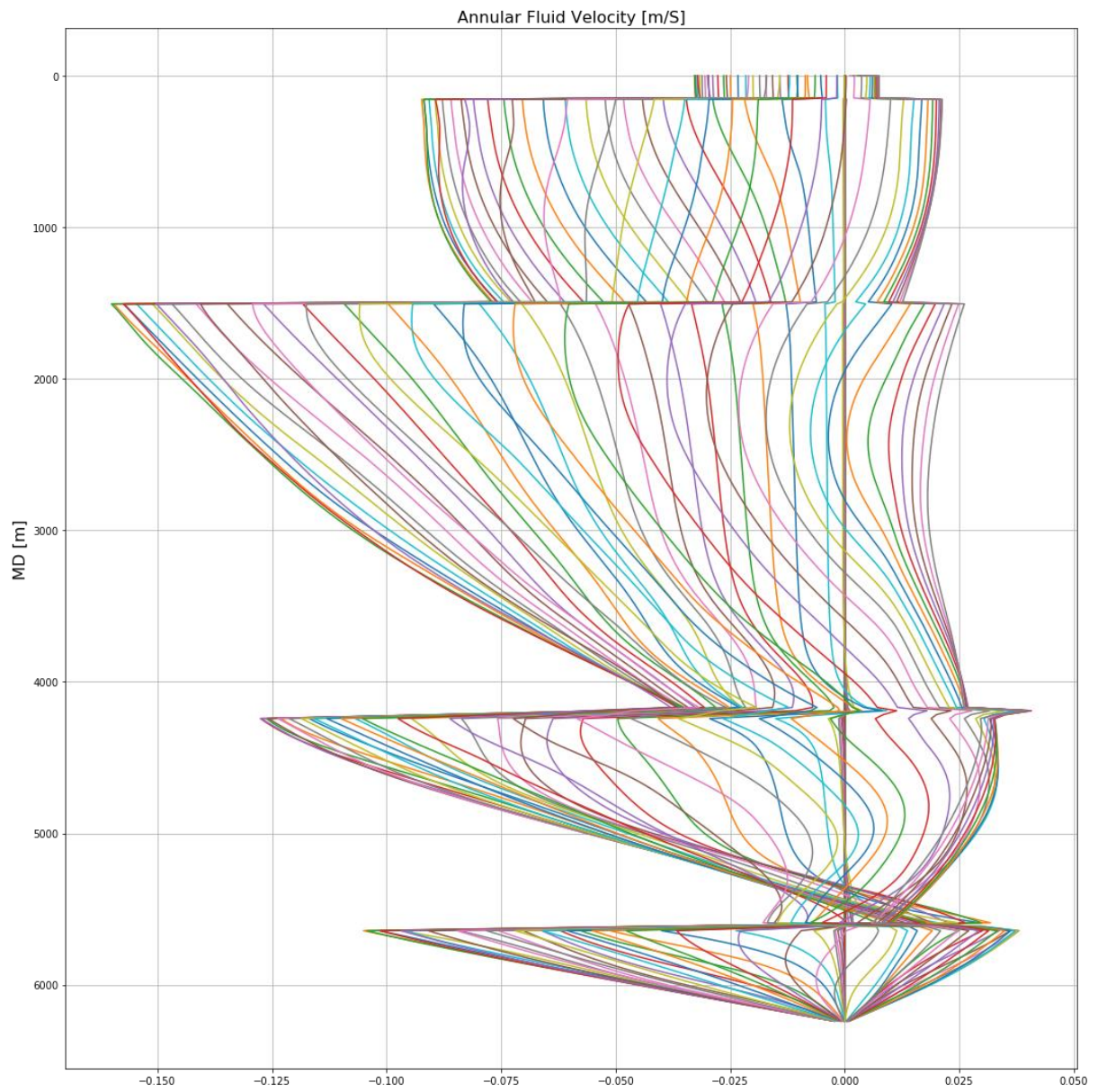
matrix. As the rows of coefficient matrix A (corresponding the number of equations or constraints) is greater than the number of variable in the variable matrix, the system is overdetermined. A linear system is consistent if and only if the coefficient matrix has the same rank as the augmented matrix otherwise it is inconsistent. The rank of matrix corresponds to the maximal number of linearity independent columns of a matrix. In this example matrix A and D both have the ranks equal two which means the system is consistent (Datta, 2010; Gentle, 2012)⁷.

⁷ These references are used to explain mathematical concepts for Appendix C

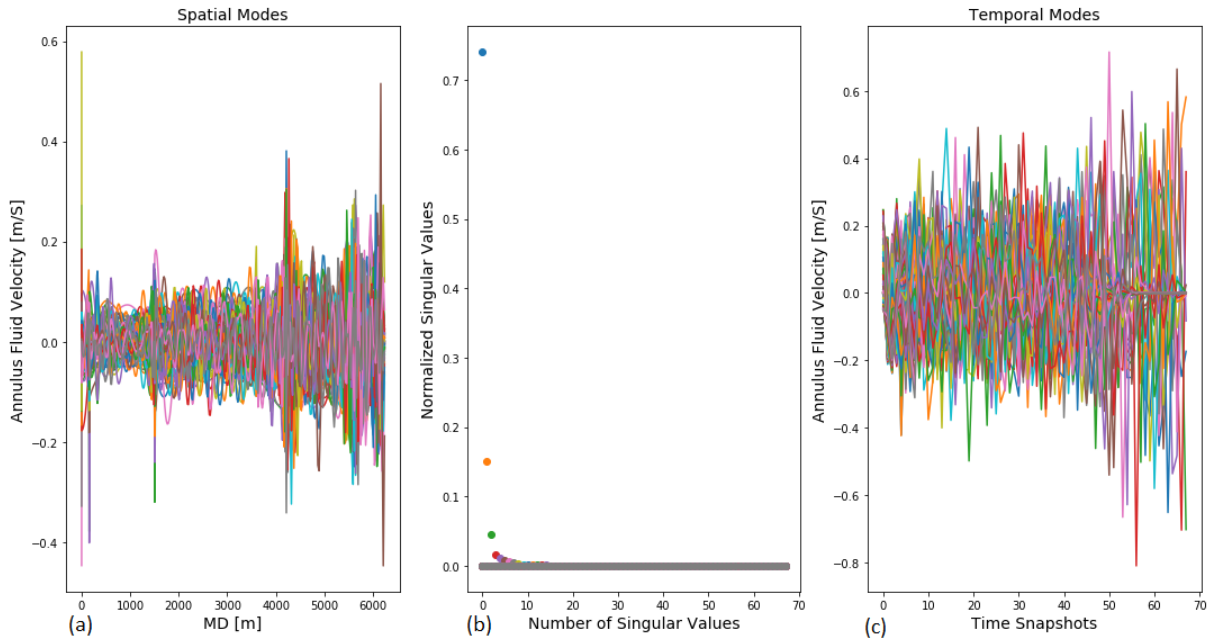
Appendix D – DMD reconstructed result of single sub dataset, interval No. 2

– 5

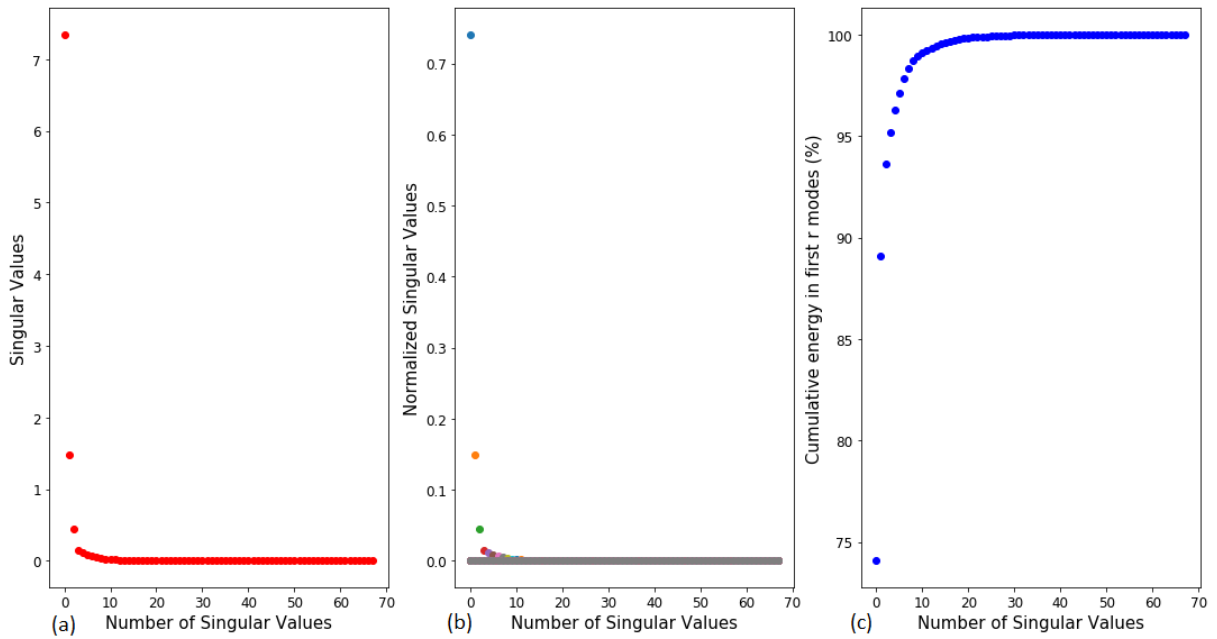
- Interval No. 2: 44 – 111



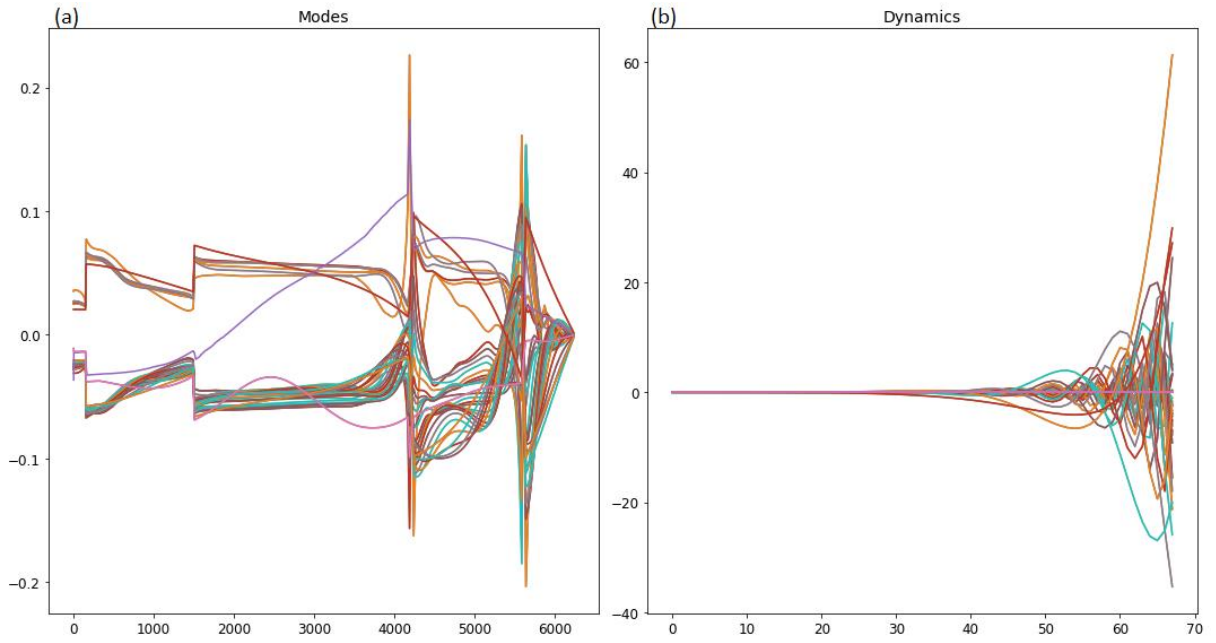
Append. D.1 *Annular fluid velocity, interval No. 2 (44 - 111)*



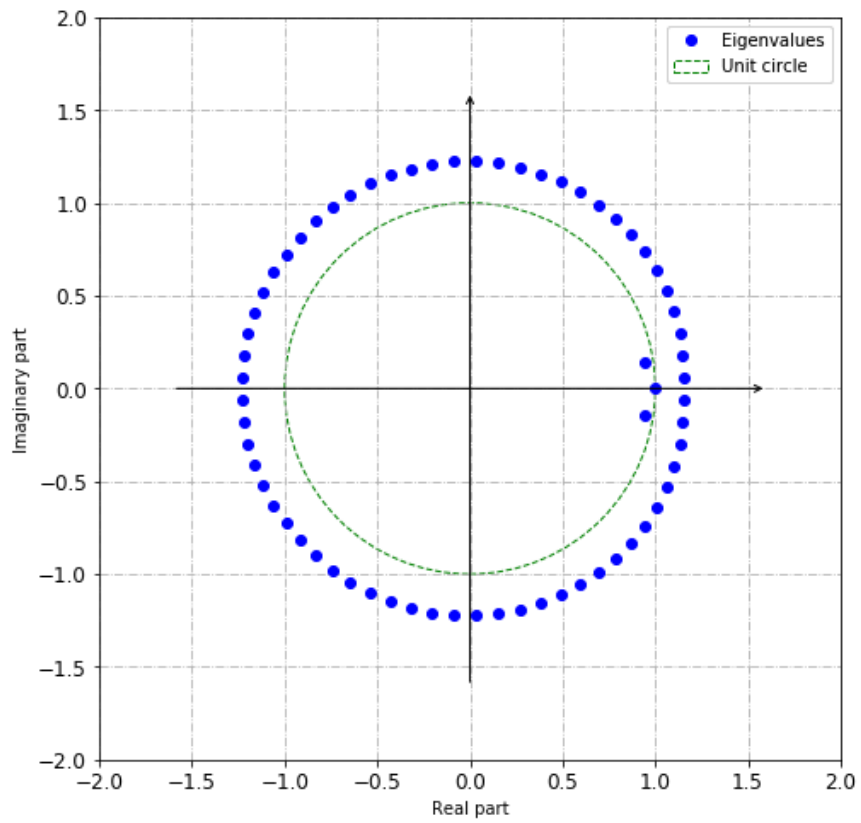
Append. D.2 Singular value decomposition (SVD) of interval 2(44 - 111)



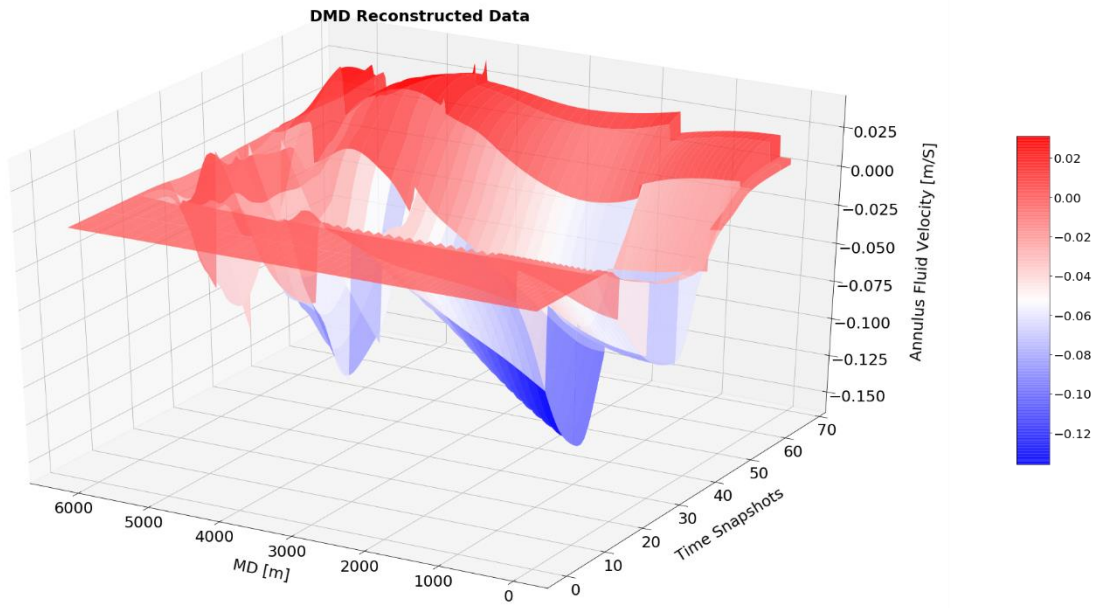
Append. D.3 Singular values of interval No. 2 (44 - 111)



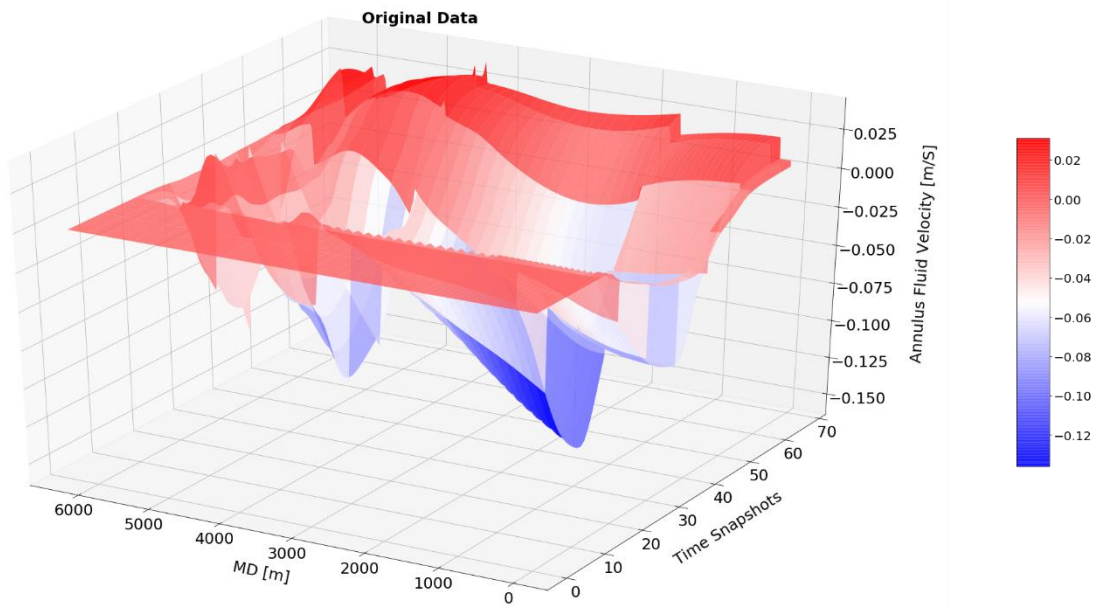
Append. D.4 *The Spatial modes and their dynamics in time, interval 2 (44 - 111)*



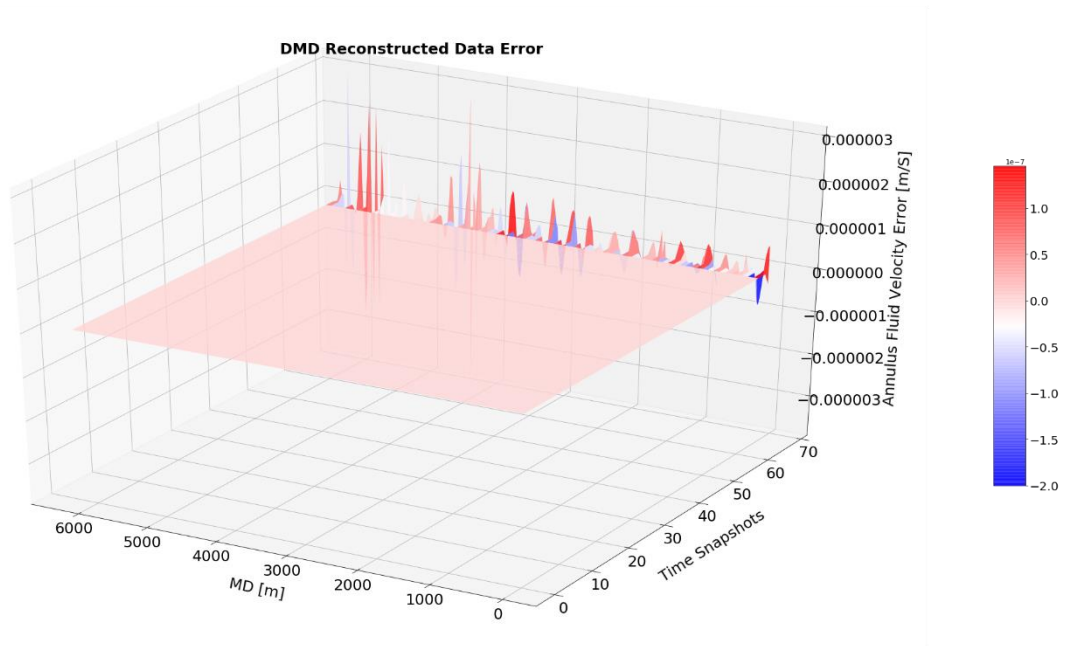
Append. D.5 *Eigenvalues of interval 2 (44 - 111)*



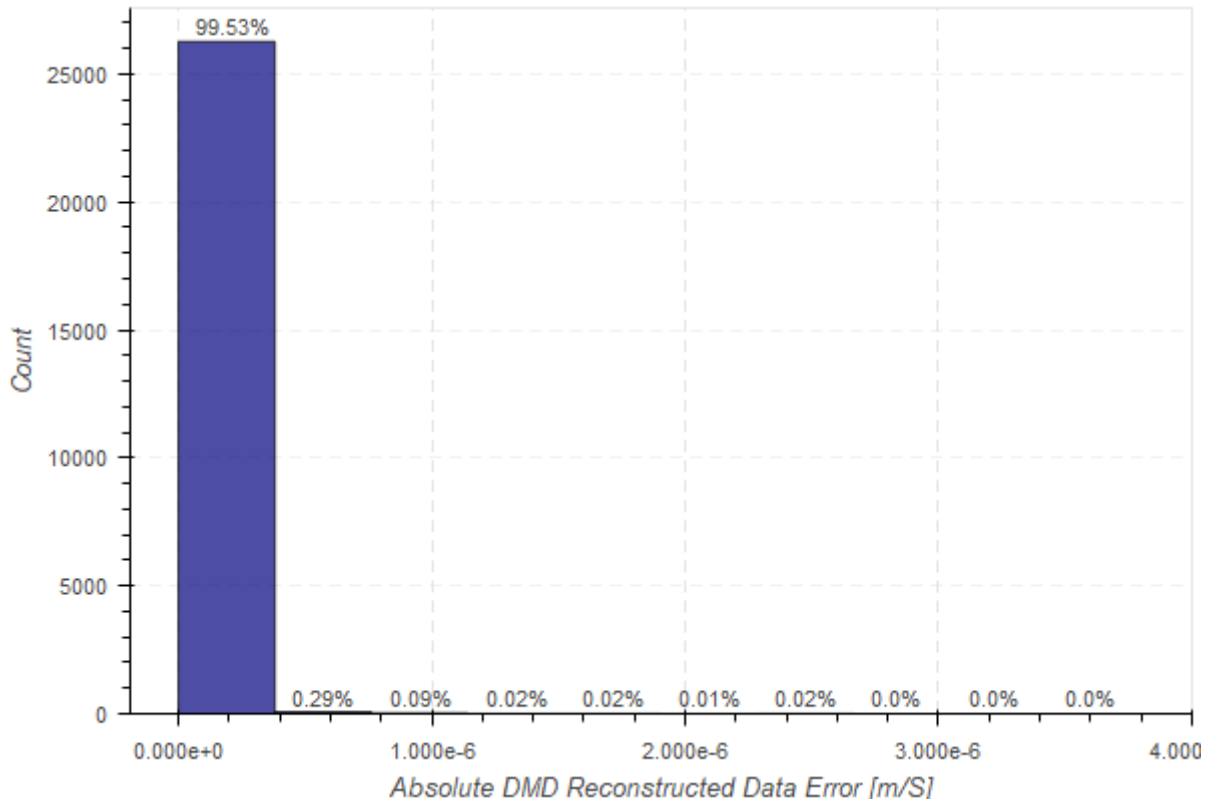
Append. D.6 *DMD reconstruction of the interval No. 2 (44 - 111)*



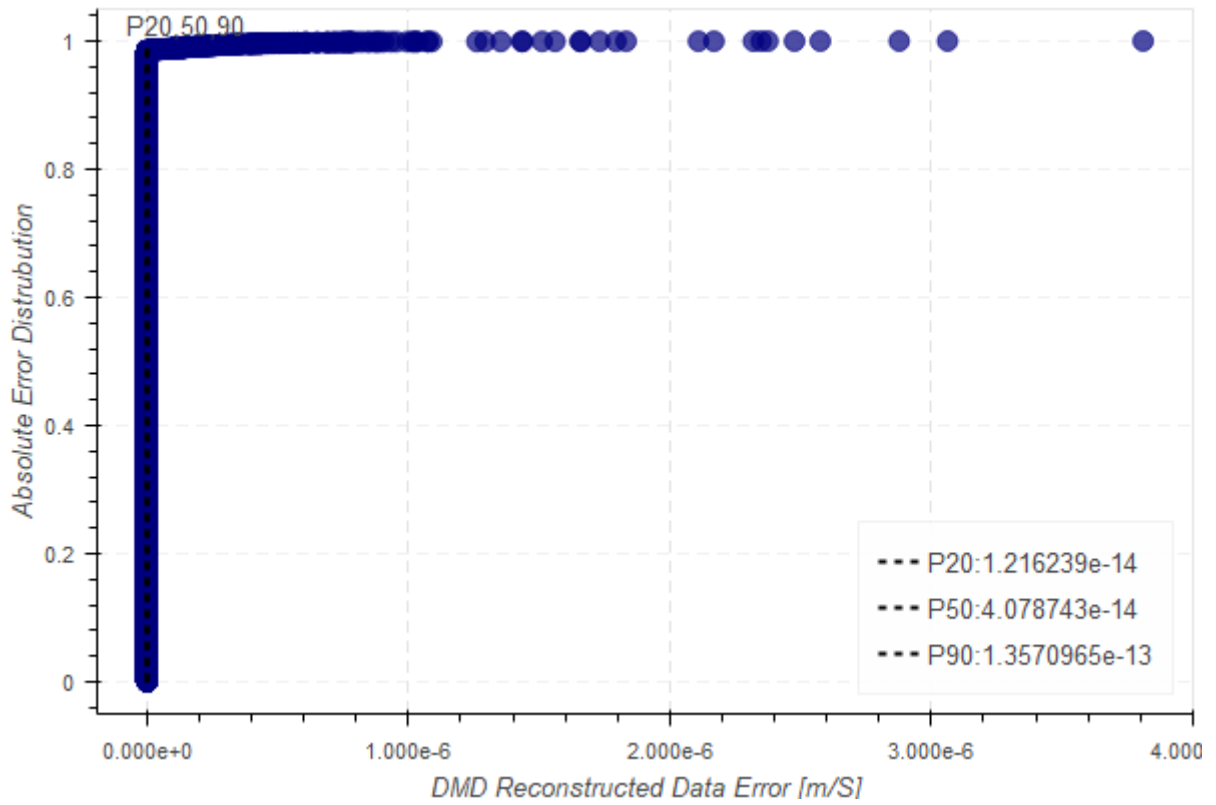
Append. D.7 *Original data interval No. 2 (44 - 111)*



Append. D.8 DMD reconstructed data error interval No. 2 (44 - 111)

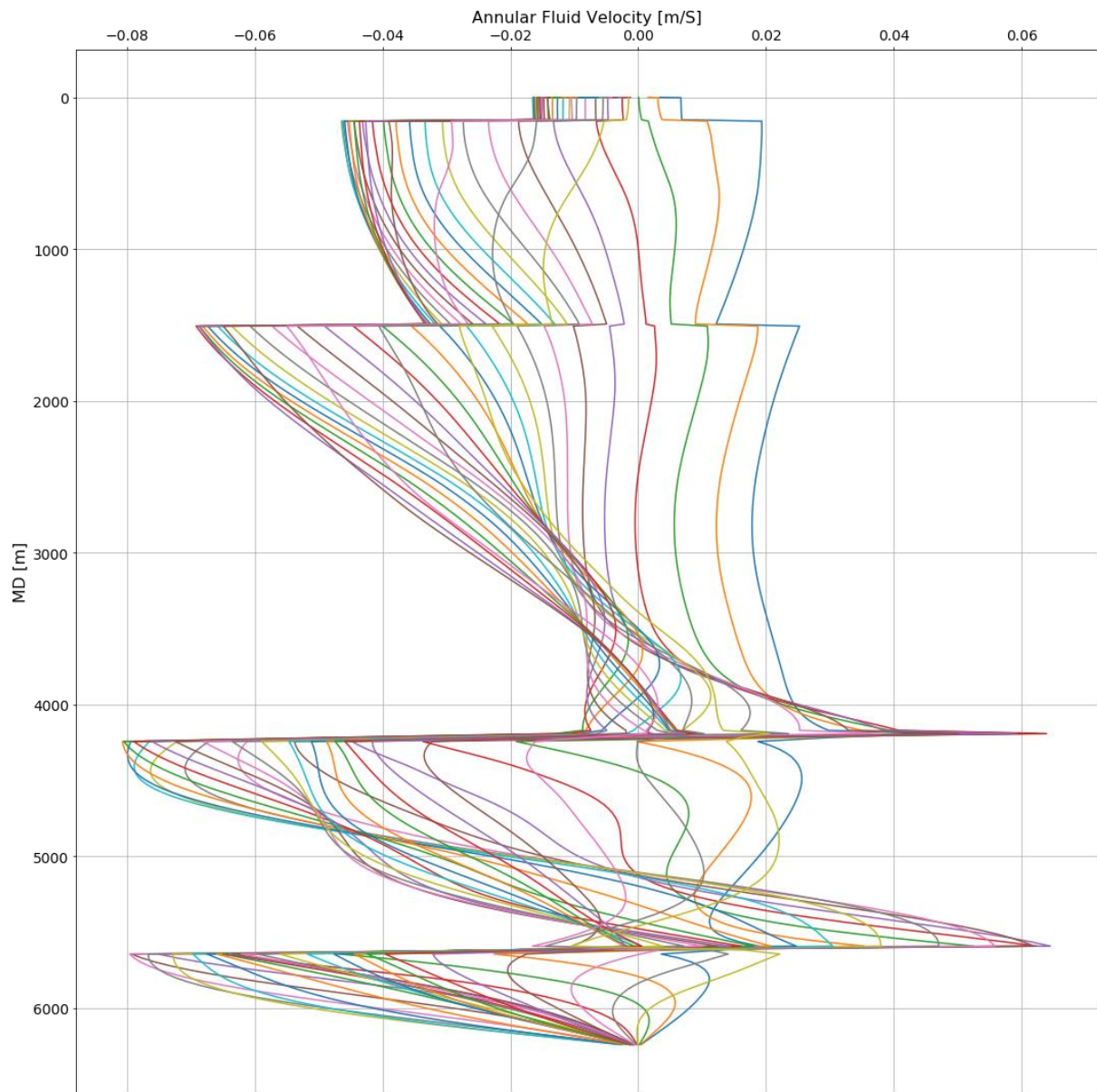


Append. D.9 The frequency distribution of the absolute DMD reconstructed data error, interval No. 2 (44 - 111)

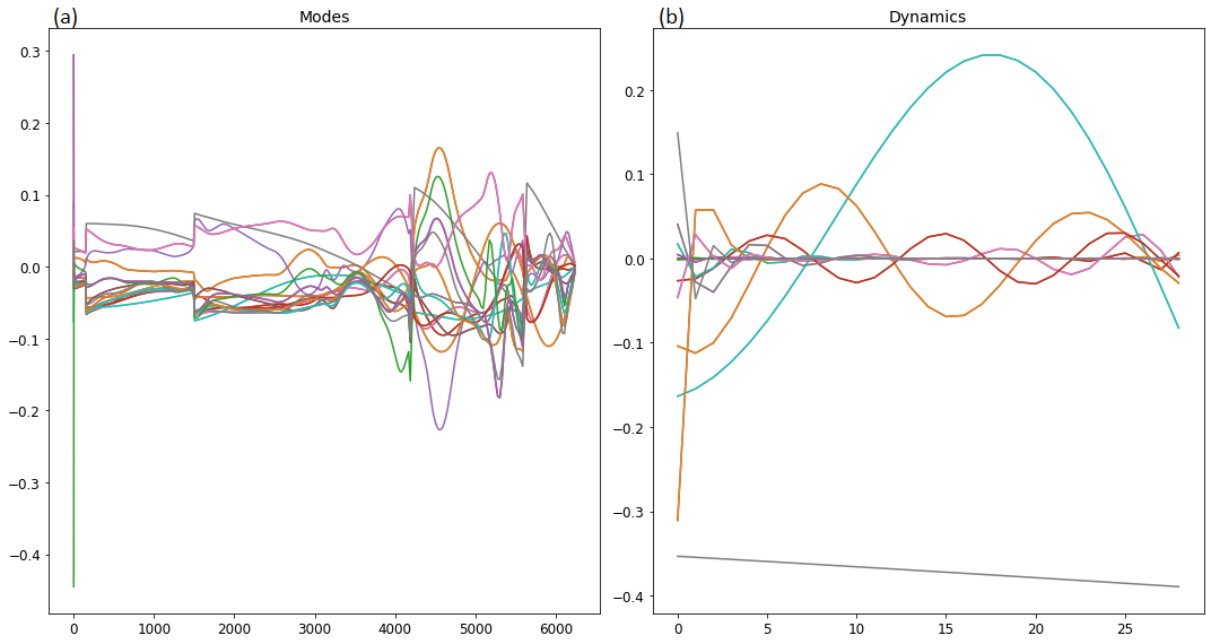


Append. D.10 *The distribution of the absolute DMD reconstructed data error, interval No. 2 (44 - 111)*

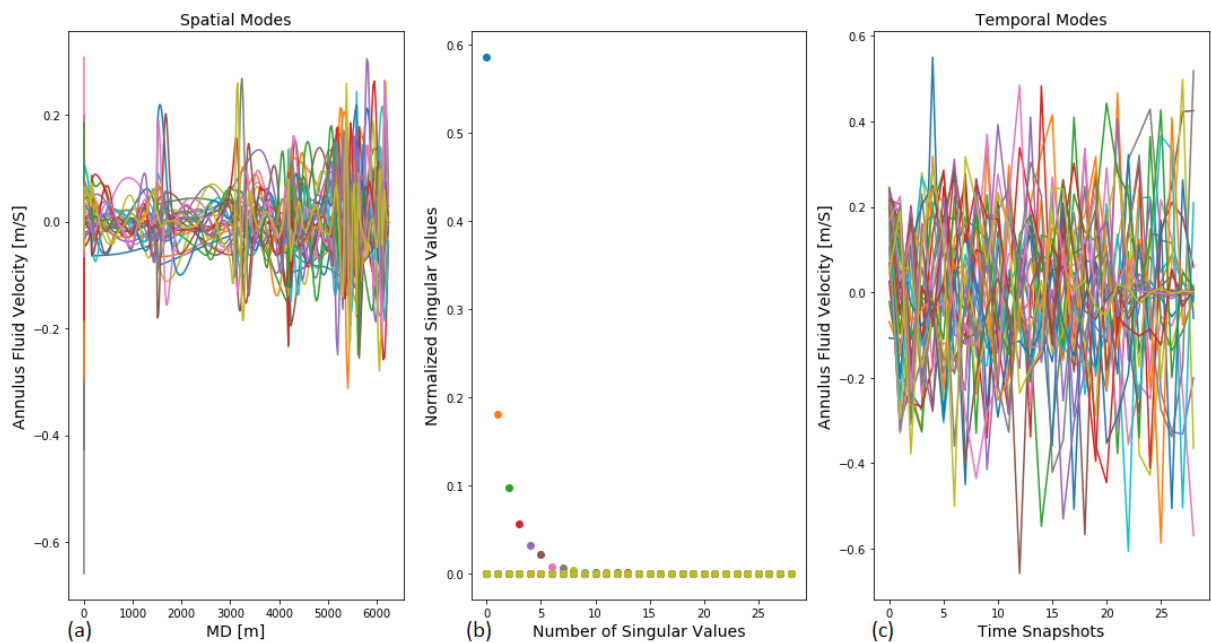
- Interval No. 3: 112 – 140



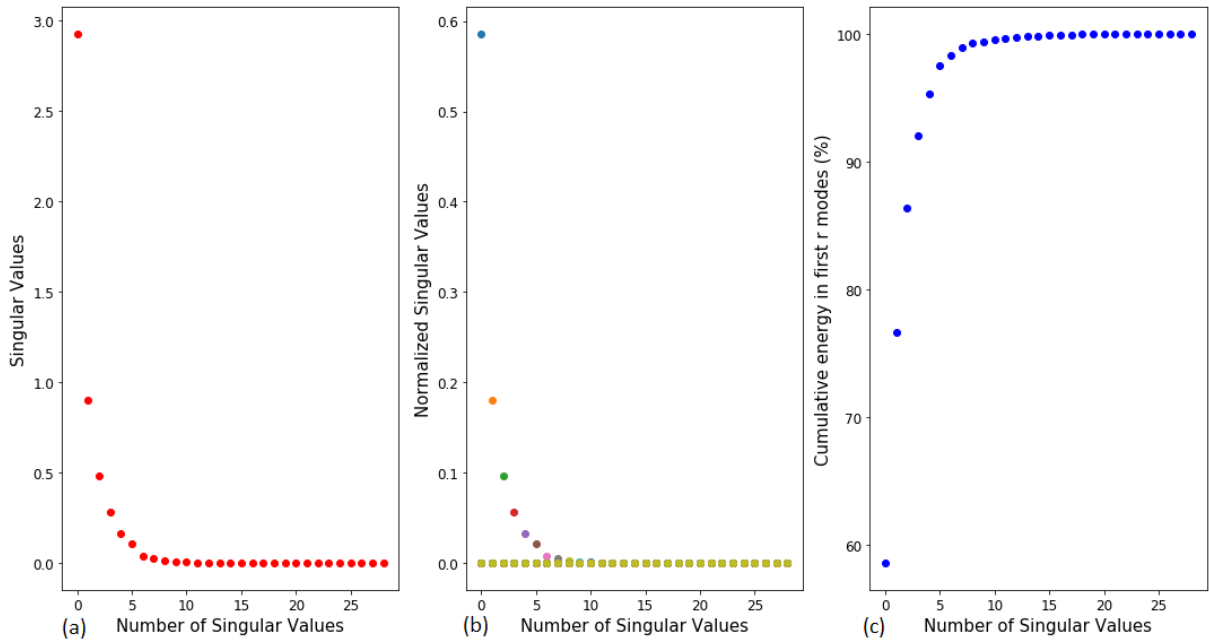
Append. D.11 Annular fluid velocity, interval No. 3 (112 - 140)



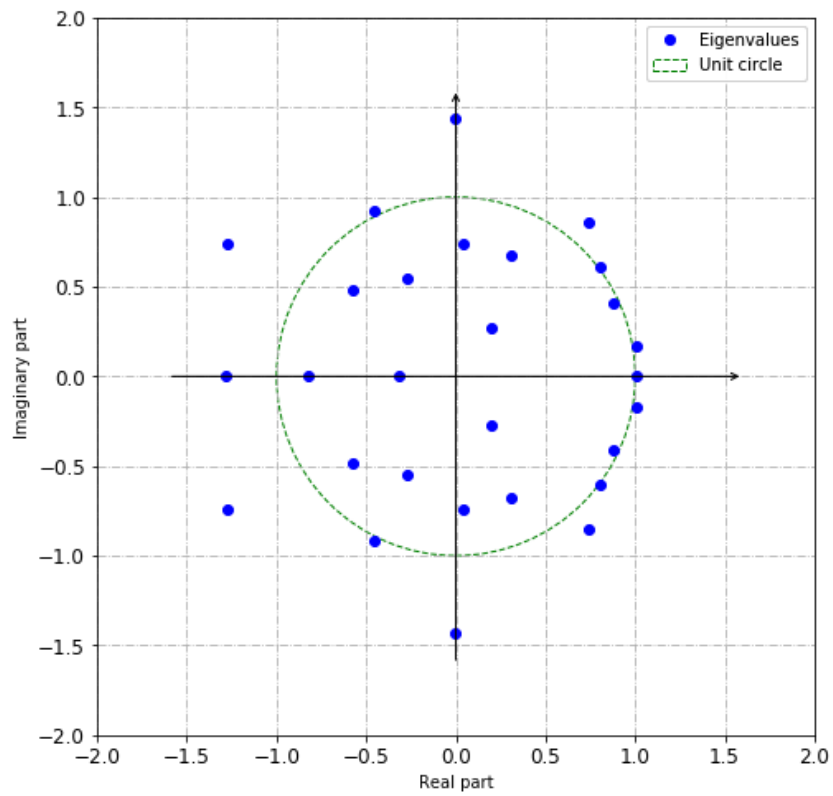
Append. D.12 *The Spatial modes and their dynamics in time, interval 3 (112 - 140)*



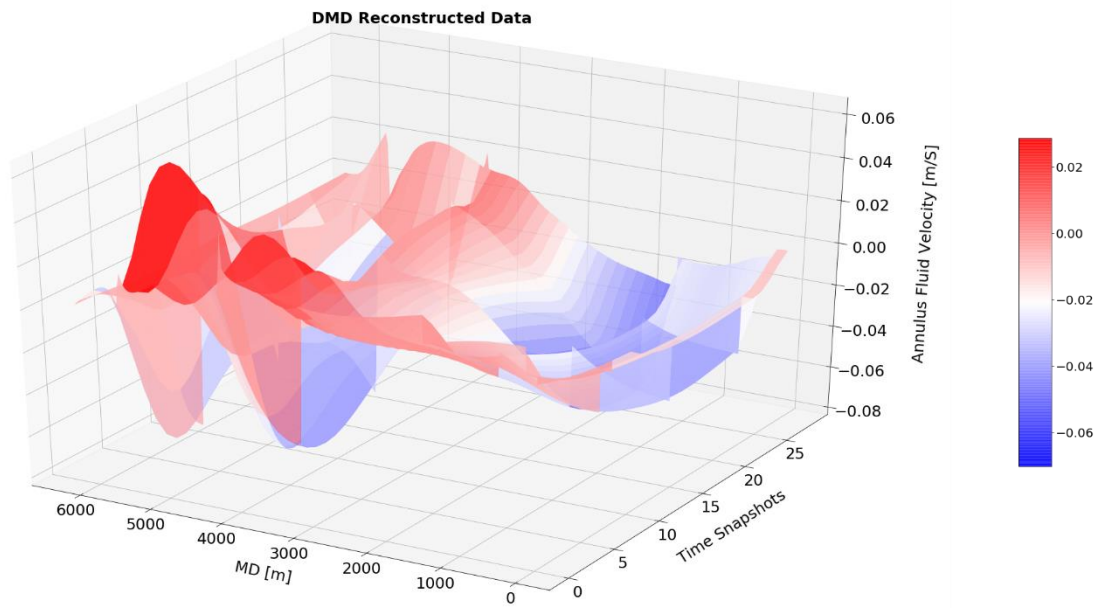
Append. D.13 *Singular value decomposition (SVD) of interval 3 (112 - 140)*



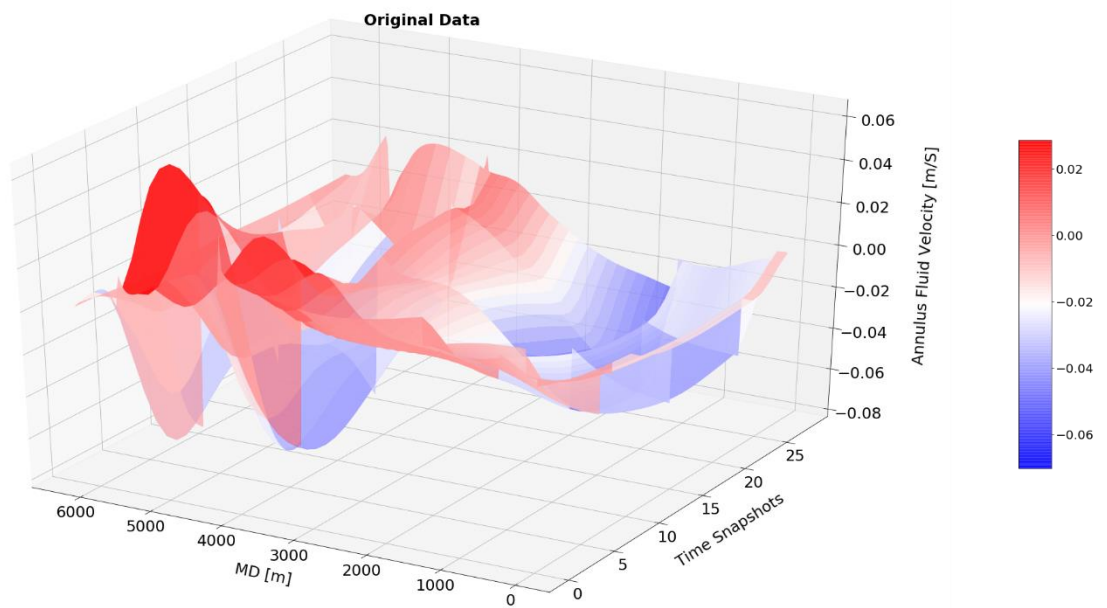
Append. D.14 Singular values of interval No. 3 (112 - 140)



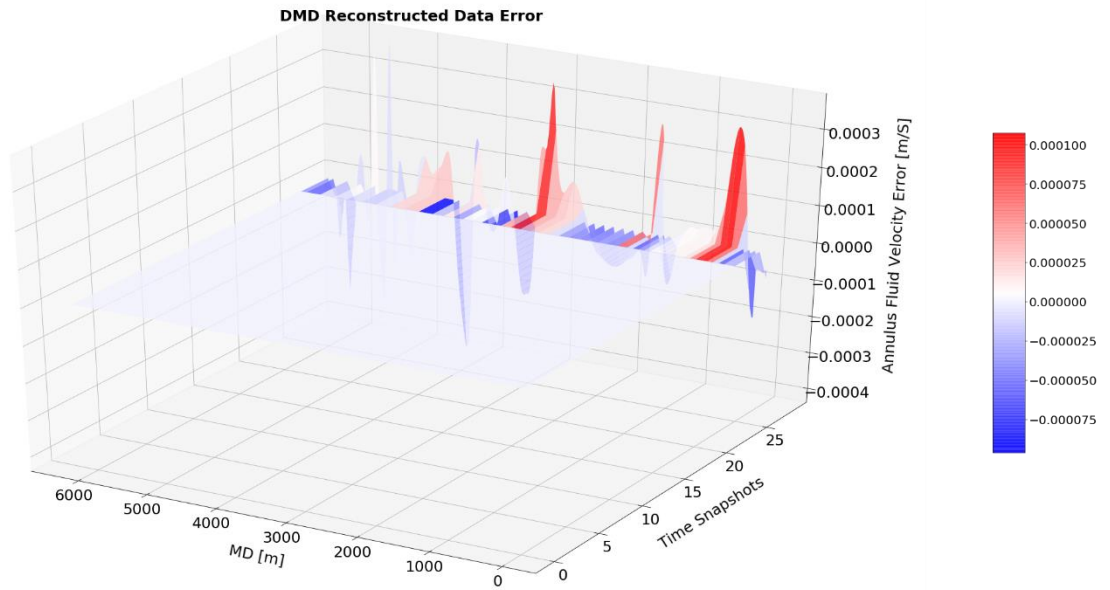
Append. D.15 Eigenvalues of interval 3 (112 - 140)



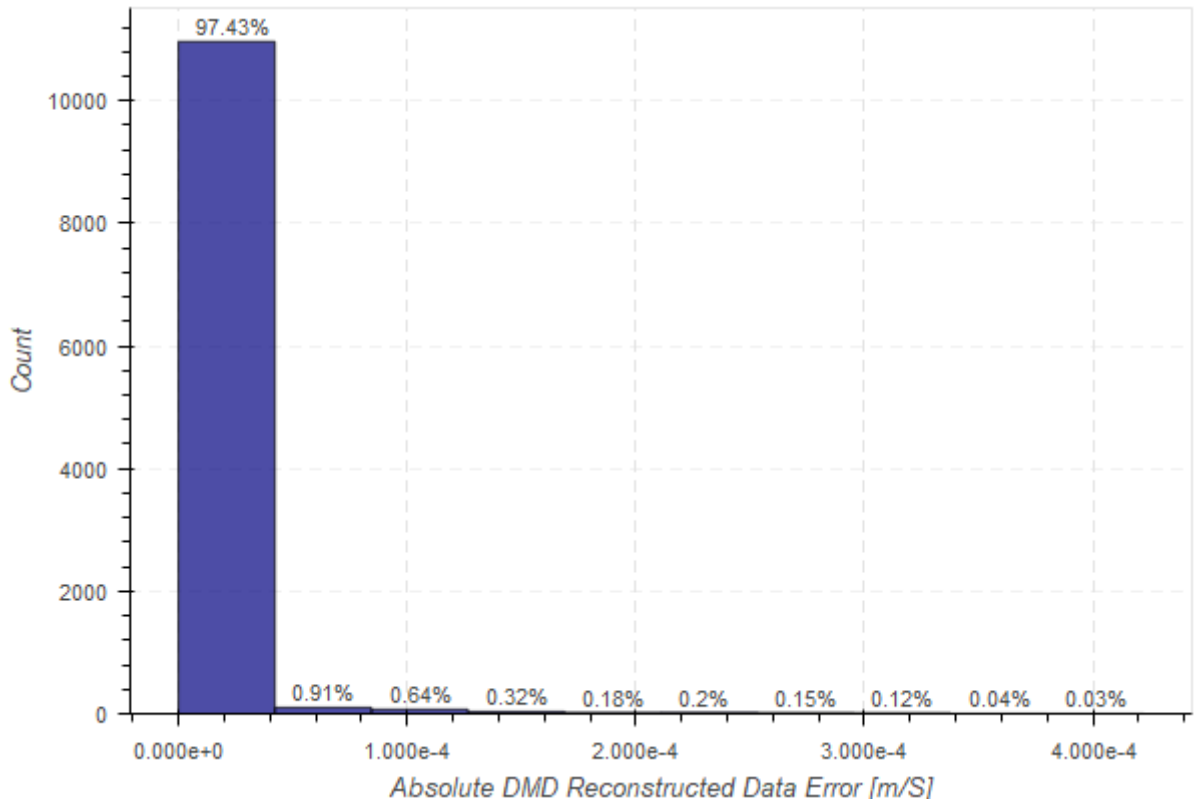
Append. D.16 *DMD reconstruction of the interval No. 3 (112 - 140)*



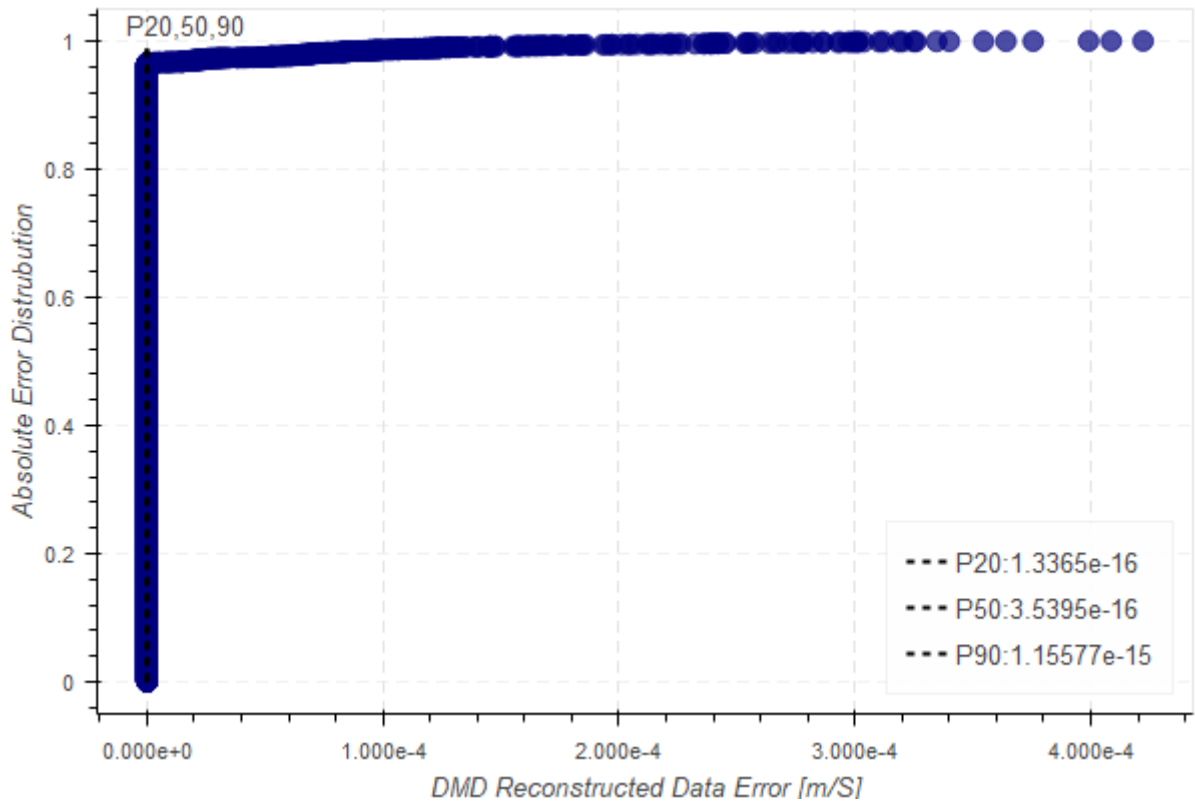
Append. D.17 *Original data interval No. 3 (112 - 140)*



Append. D.18 DMD reconstructed data error interval No. 3 (112 - 140)

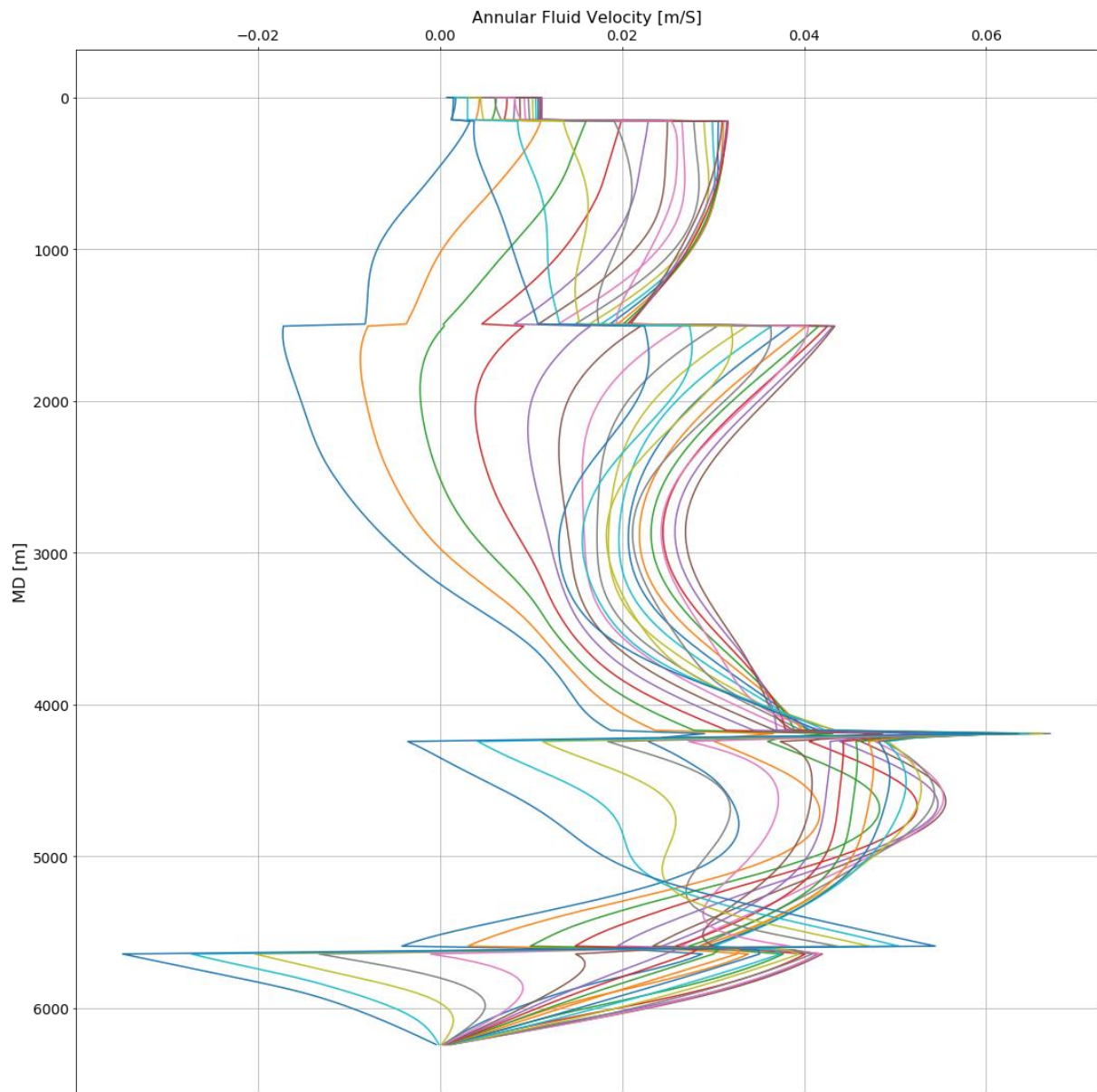


Append. D.19 The frequency distribution of the absolute DMD reconstructed data error, interval No. 3 (112 - 140)

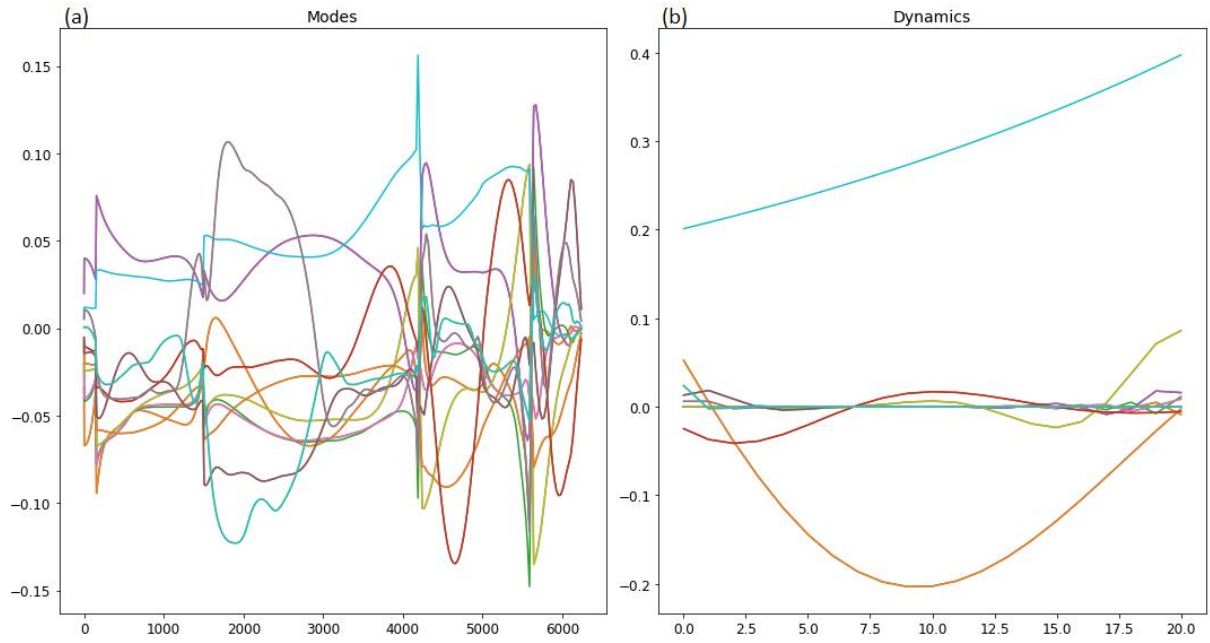


Append. D.20 *The distribution of the absolute DMD reconstructed data error, interval No. 3 (112 - 140)*

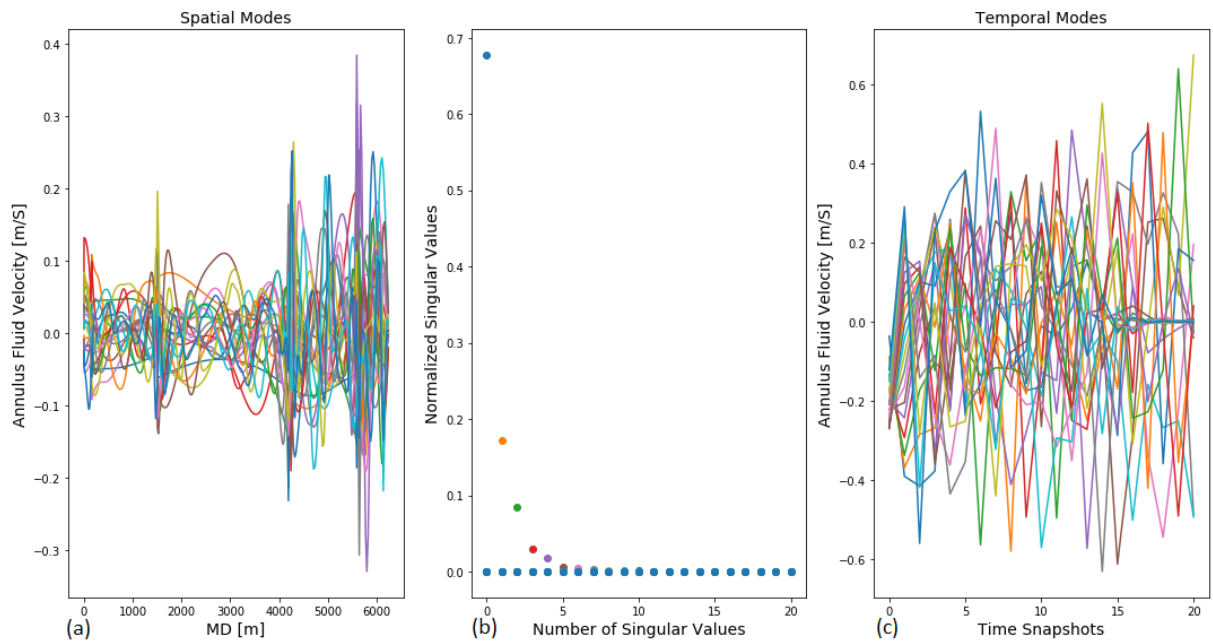
- Interval No. 4: 141 – 161



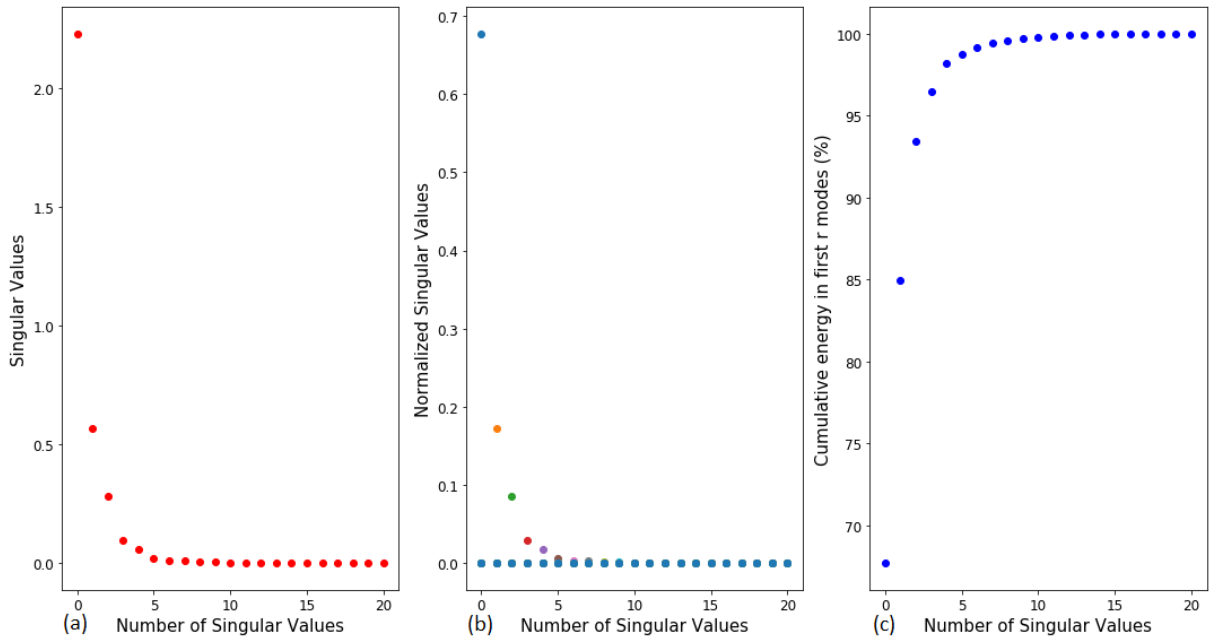
Append. D.21 *The distribution of the absolute DMD reconstructed data error, interval No. 4 (141 - 161)*



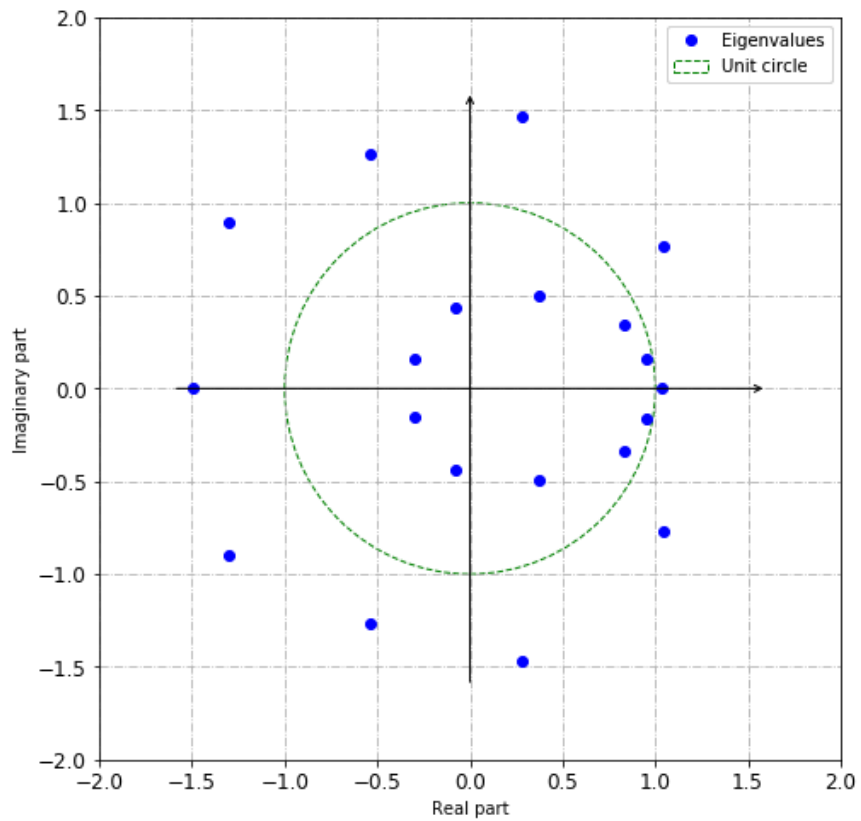
Append. D.22 *The Spatial modes and their dynamics in time, interval 4 (141 - 161)*



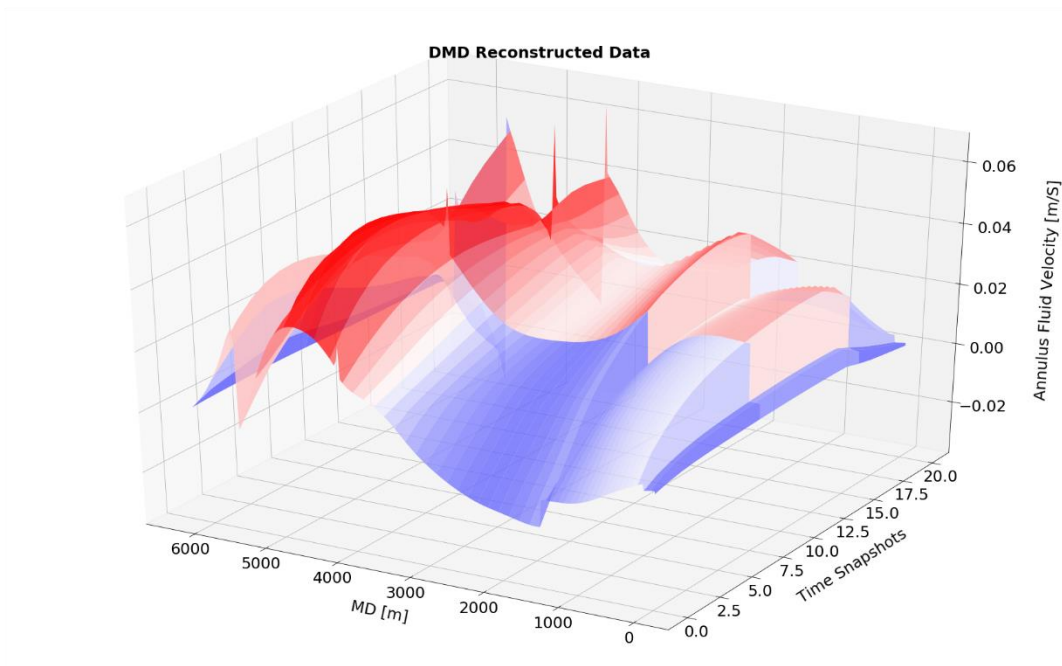
Append. D.23 *Singular value decomposition (SVD) of interval 4 (141 - 161)*



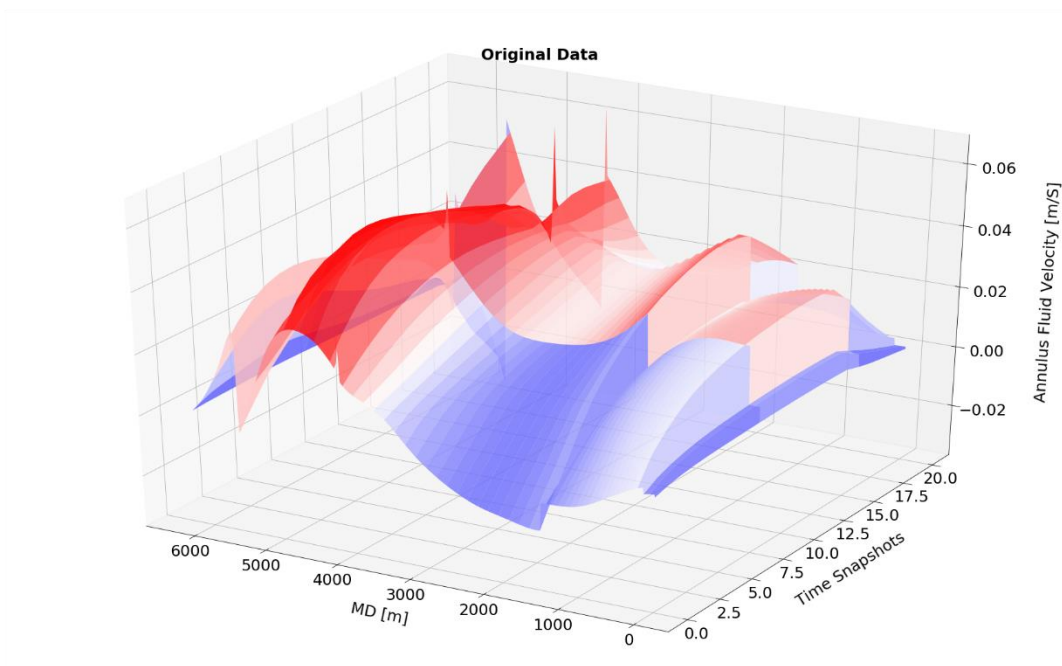
Append. D.24 Singular values of interval No. 4 (141 - 161)



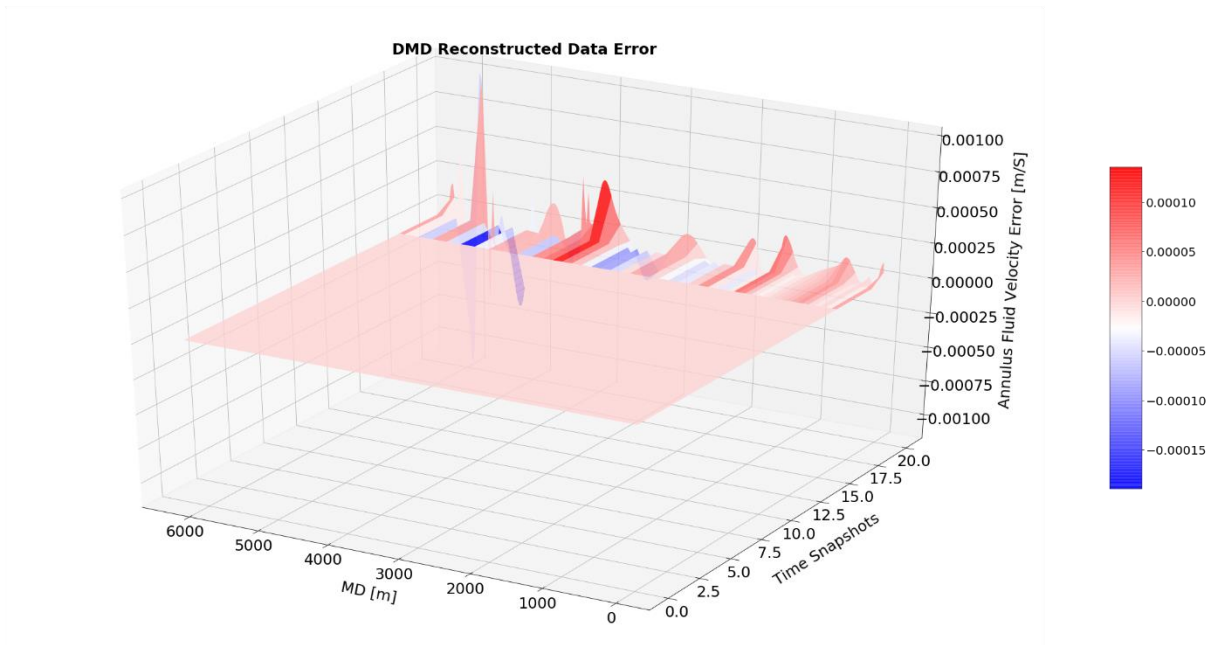
Append. D.25 Eigenvalues of interval 4 (141 - 161)



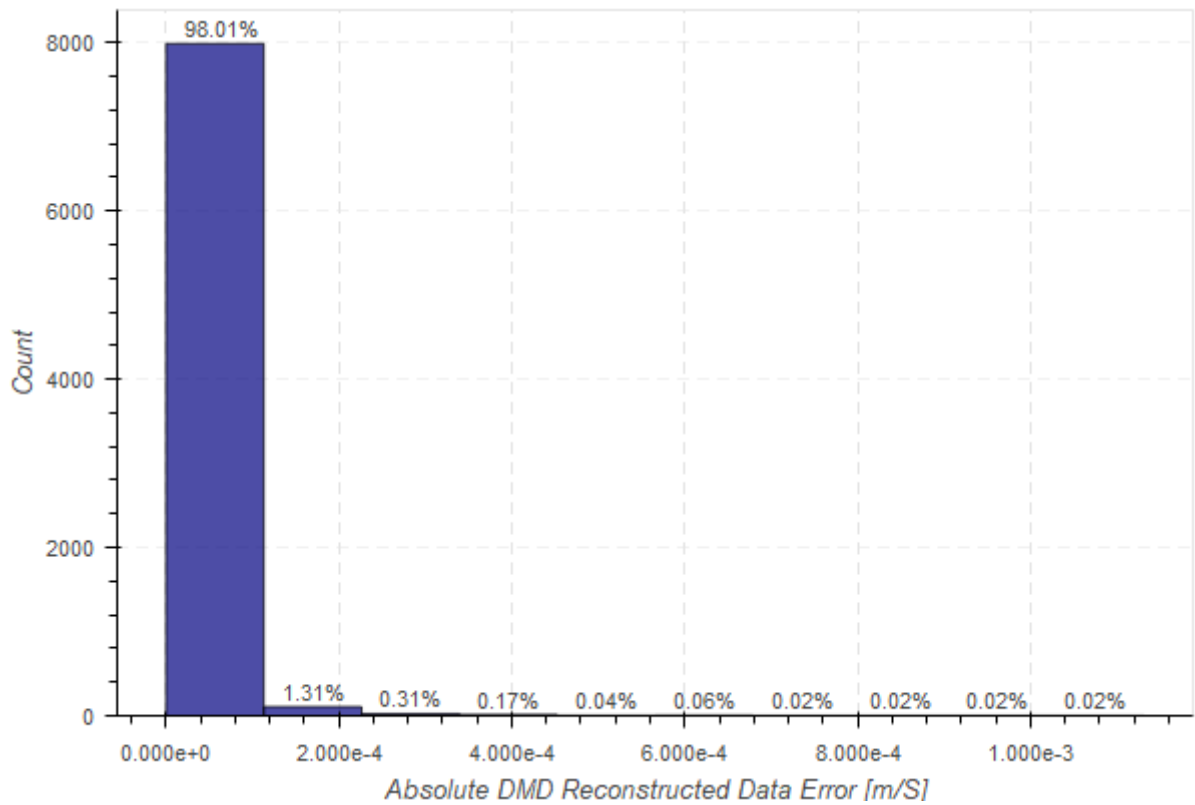
Append. D.26 *DMD reconstruction of the interval No. 4 (141 - 161)*



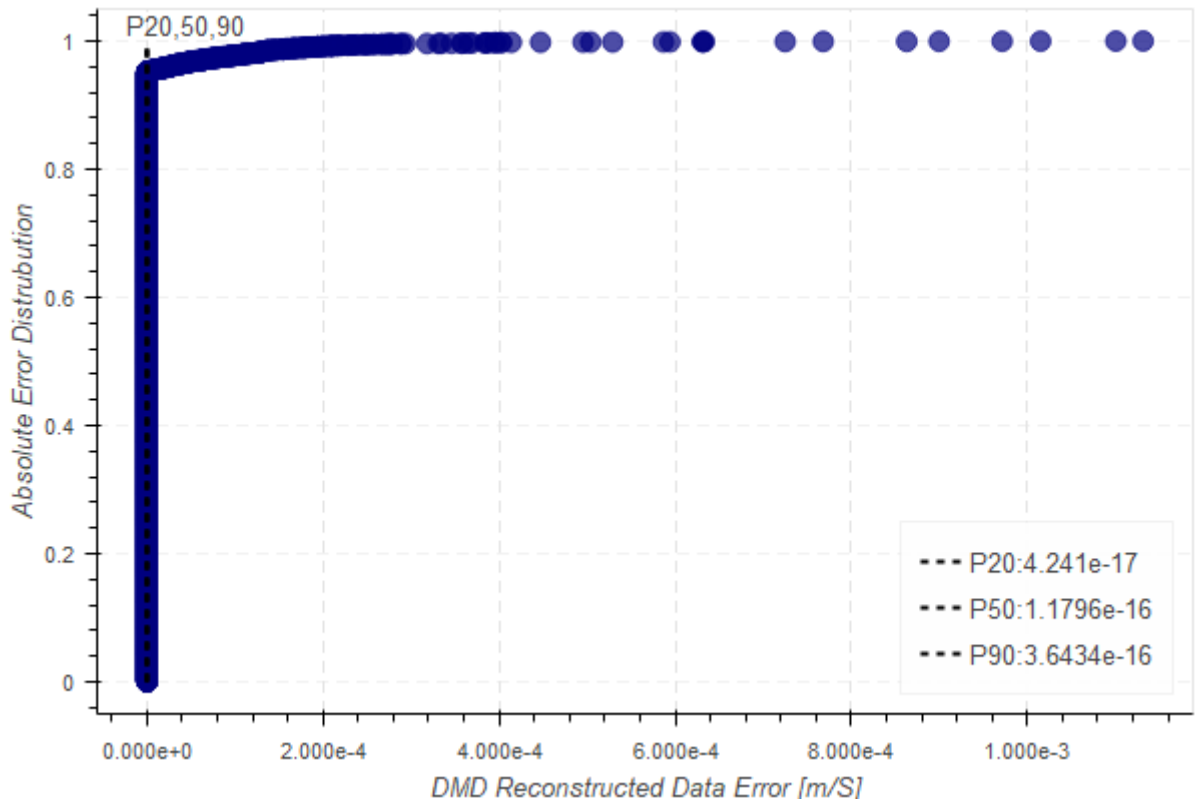
Append. D.27 *Original data interval No. 4 (141 - 161)*



Append. D.28 DMD reconstructed data error interval No. 4 (141 - 161)

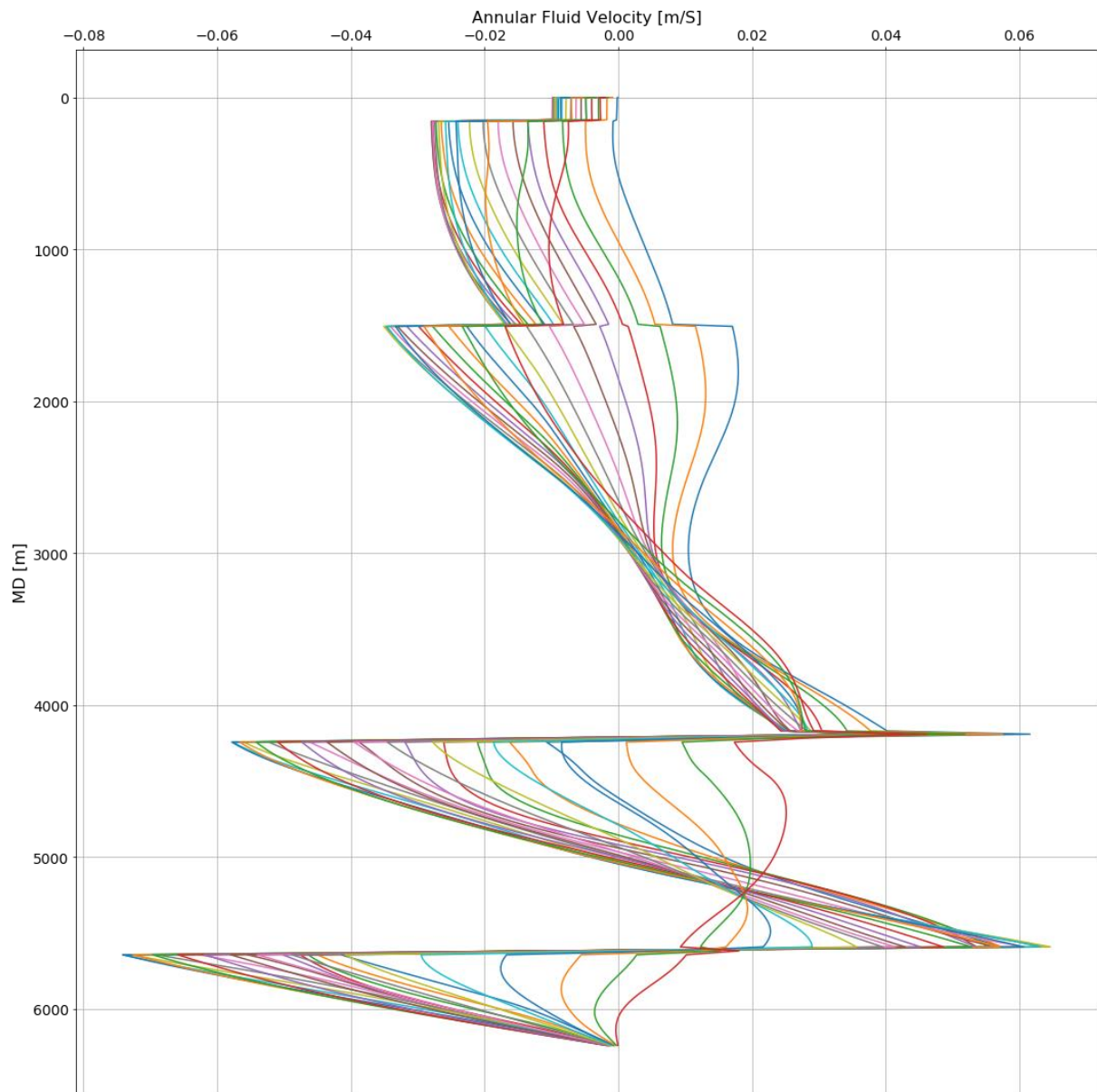


Append. D.29 The frequency distribution of the absolute DMD reconstructed data error, interval No. 4 (141 - 161)

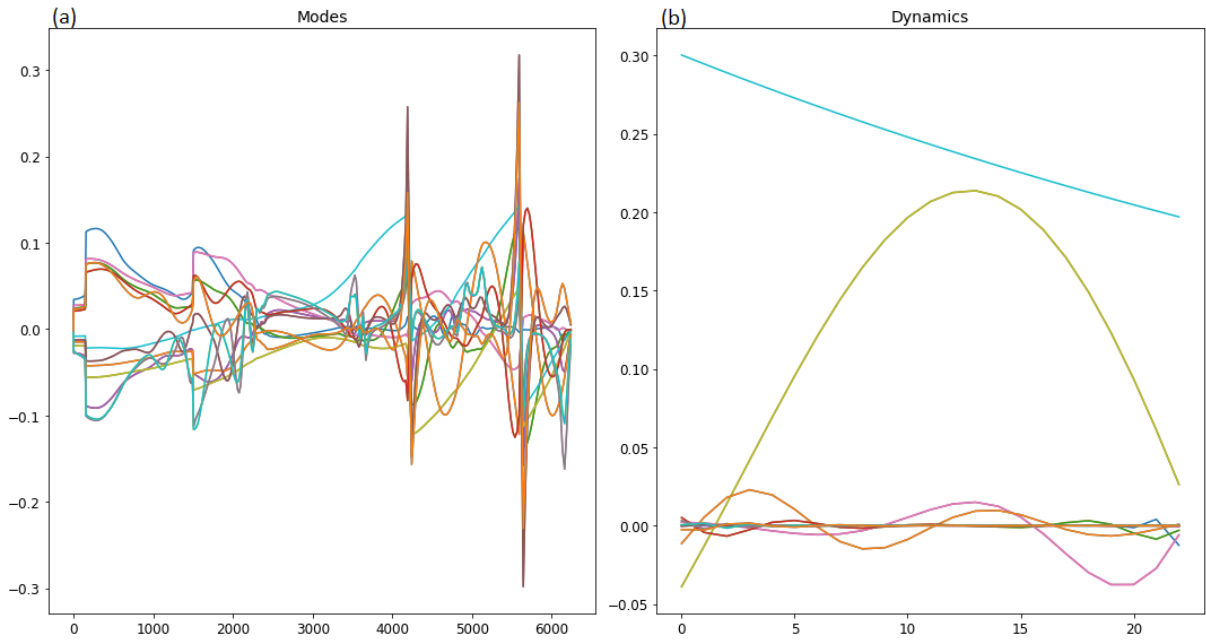


Append. D.30 *The distribution of the absolute DMD reconstructed data error, interval No. 4 (141 - 161)*

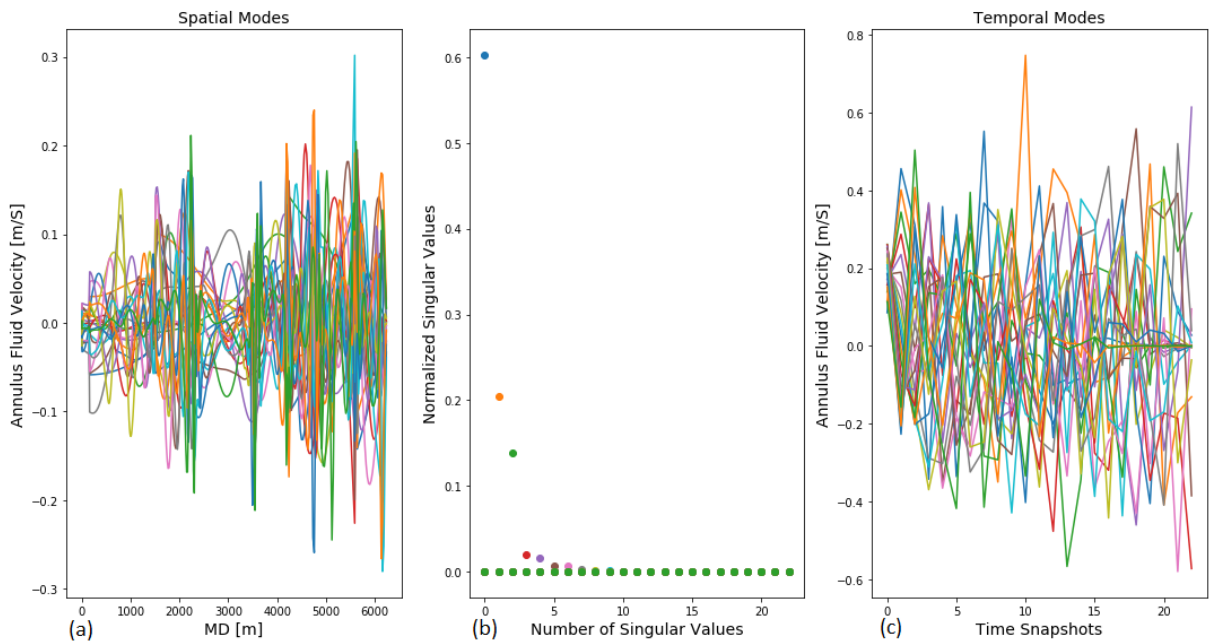
- Interval No. 5: 162 – 185



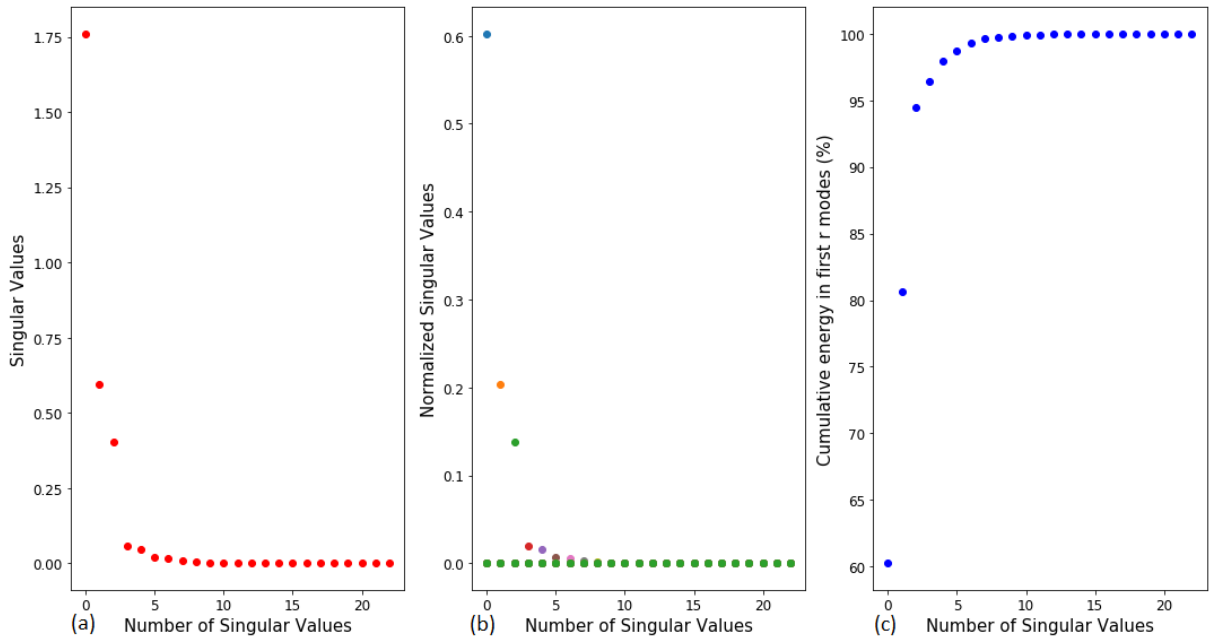
Append. D.31 Annular fluid velocity, interval No. 5 (162 - 185)



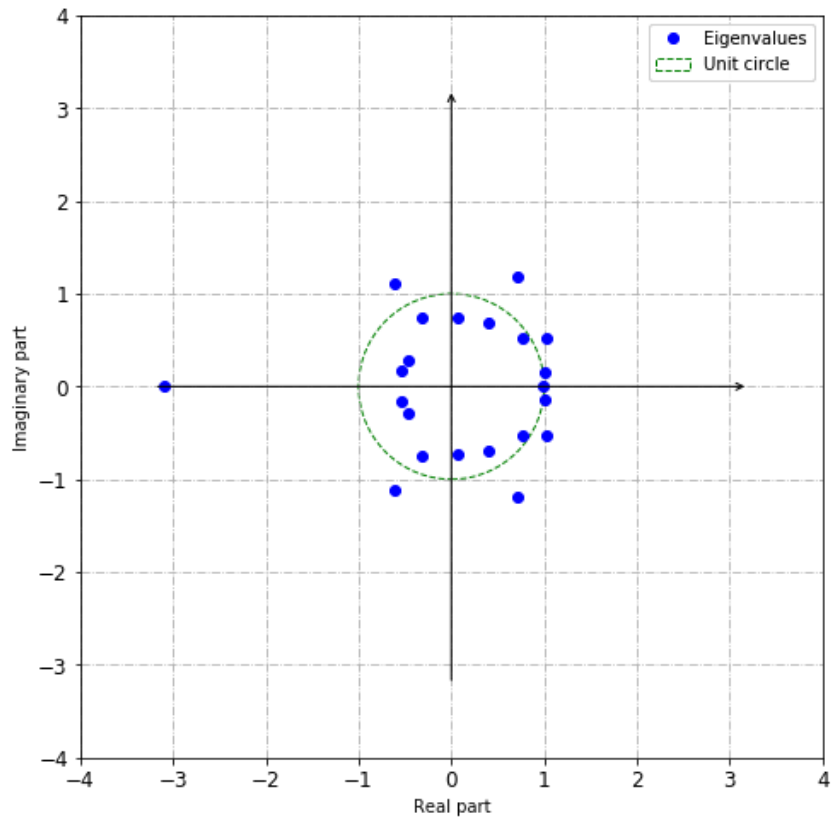
Append. D.32 *The Spatial modes and their dynamics in time, interval 5 (162 - 185)*



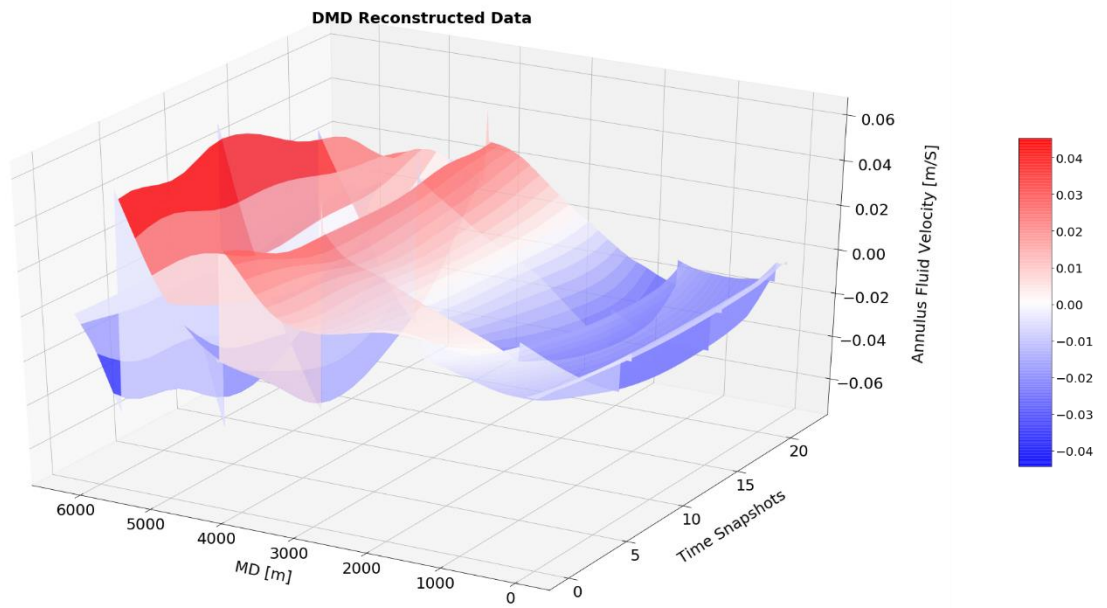
Append. D.33 *Singular value decomposition (SVD) of interval 5 (162 - 185)*



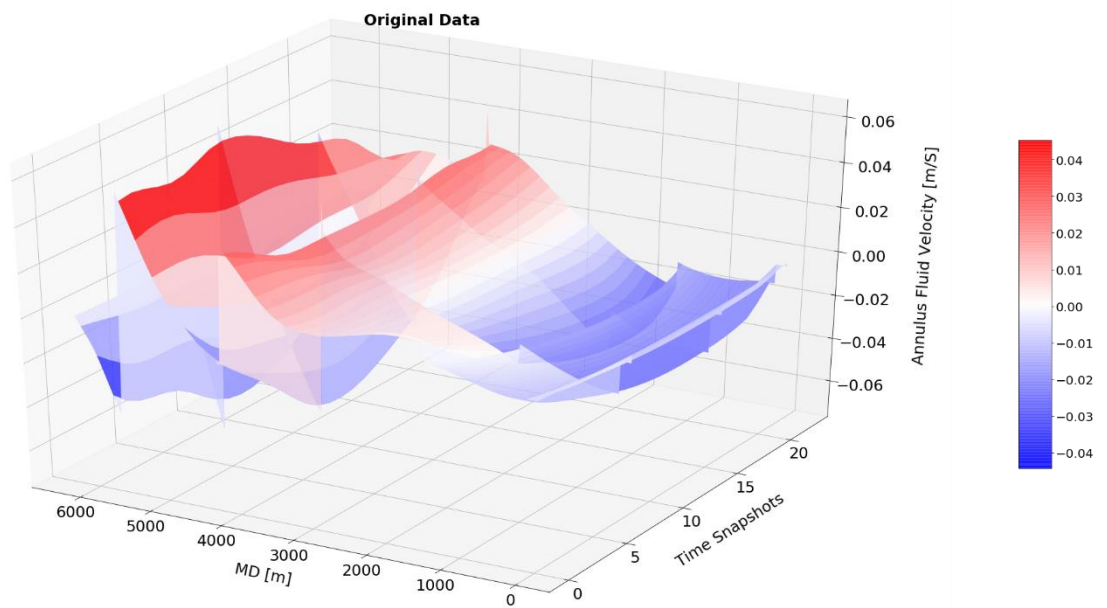
Append. D.34 Singular values of interval No. 5 (162 - 185)



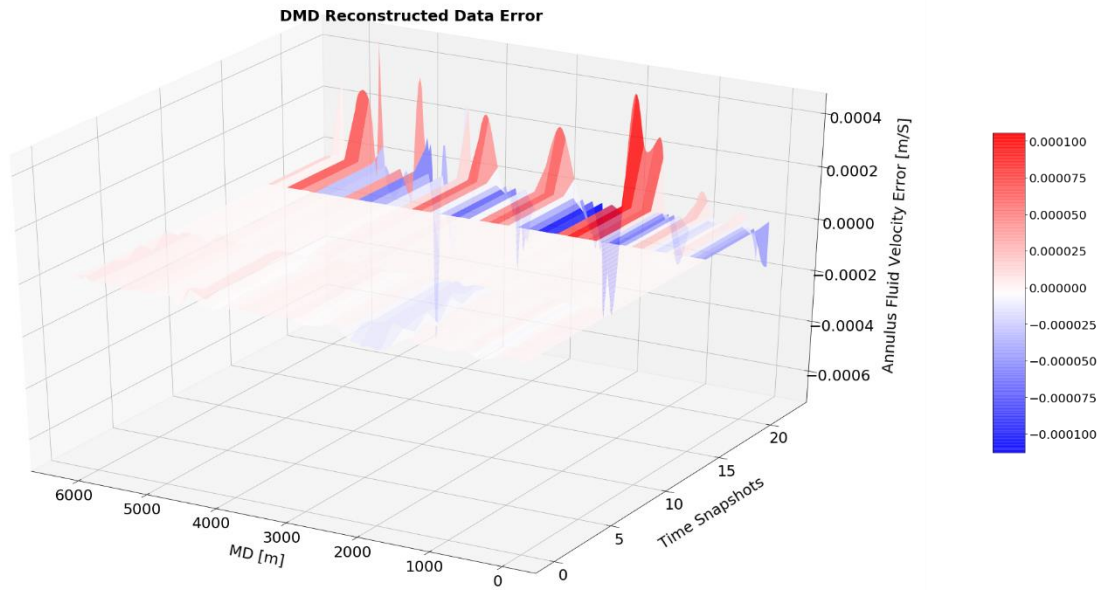
Append. D.35 Eigenvalues of interval 5 (162 - 185)



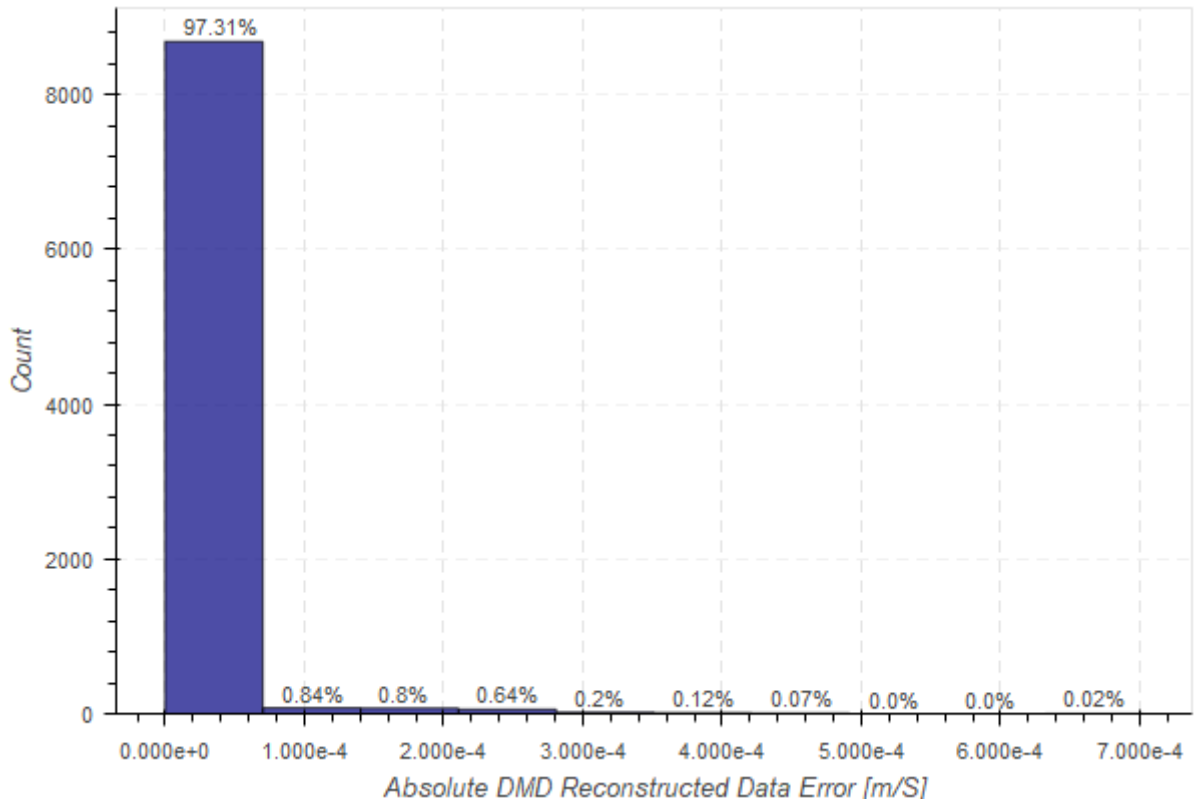
Append. D.36 DMD reconstruction of the interval No. 5 (162 - 185)



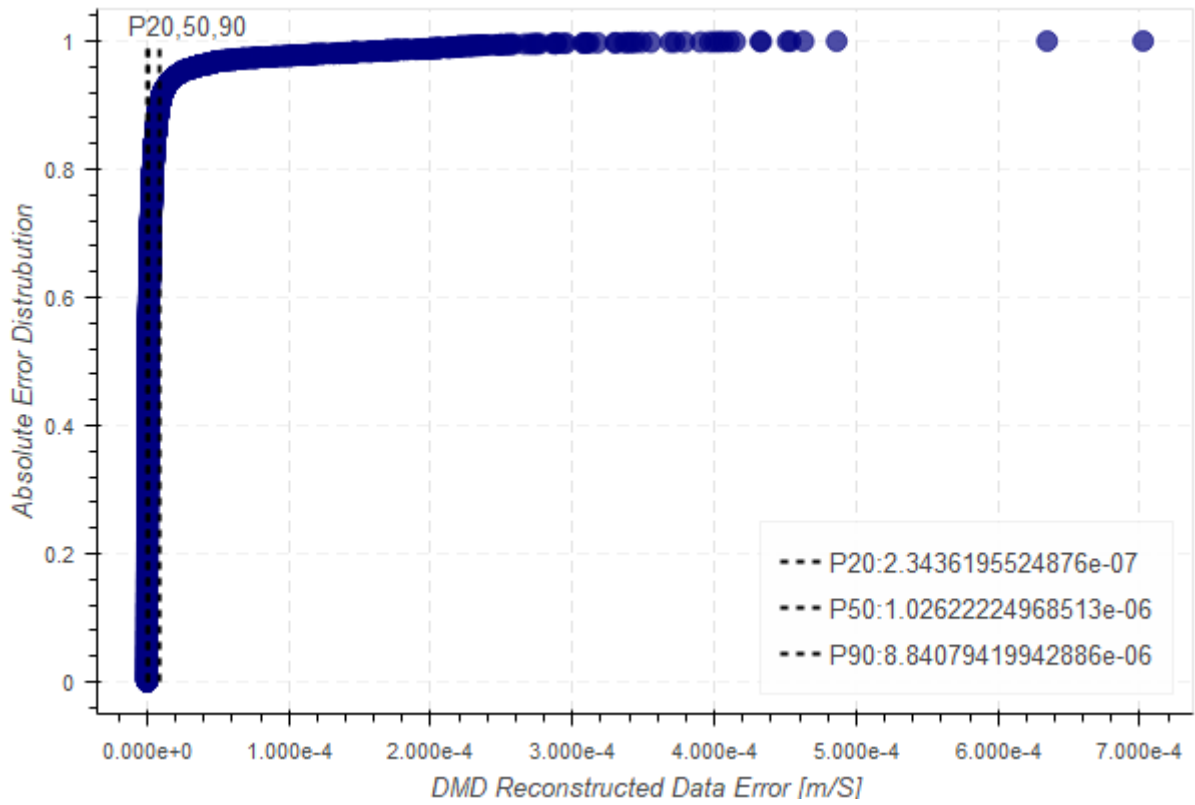
Append. D.37 Original data interval No. 5 (162 - 185)



Append. D.38 DMD reconstructed data error interval No. 5 (162 - 185)



Append. D.39 The frequency distribution of the absolute DMD reconstructed data error, interval No. 5 (162 - 185)



Append. D.40 The distribution of the absolute DMD reconstructed data error, interval No. 5 (162 - 185)

Appendix E – Python codes

The codes were written in Jupiter Notebook, so different cells are separated with dash lines.

Chapter 2: DMD example codes

```
# Import the necessary libraries
```

```
%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np
from pydmd import DMD
```

```
# Define the original signal
```

```
def f1(x,t):
    return np.sin(x)*np.exp(10.3j*t)
def f2(x,t):
    return np.cos(x)*np.exp(1.8j*t)
x = np.linspace(0, 10, 128)
t = np.linspace(0, 4*np.pi, 128)
xgrid, tgrid = np.meshgrid(x, t)
X1 = f1(xgrid, tgrid)
X2 = f2(xgrid, tgrid)
X = X1 + X2
```

```
# Plotting the original signals
```

```
titles = ['$f_1(x,t)$', '$f_2(x,t)$', '$F(x,t)$']
data = [X1, X2, X]
fig = plt.figure(figsize=(17,6))
for n, title, d in zip(range(131,134), titles, data):
    plt.subplot(n)
    plt.pcolor(xgrid, tgrid, d.real)
    plt.title(title,size=15)
    matplotlib.rc('xtick', labelsize=12)
    matplotlib.rc('ytick', labelsize=12)
plt.colorbar()
plt.show()
```

```
# SVD
```

```
UU, ss, VV = np.linalg.svd(X,full_matrices=True);
plt.figure(figsize=(15,8))
plt.subplot(1,3,1)
plt.plot(x[:],UU[:,0:ss.shape[0]].real);
plt.xlabel('x',fontsize=15)
plt.ylabel('F(x,t)',fontsize=15)
plt.title('Spatial Modes',fontsize=15)
plt.subplot(1,3,2)
plt.plot((np.diag(ss))/(np.trace(np.diag(ss))),'o');
plt.xlabel('Number of Singular Values',fontsize=15)
plt.ylabel('Normalized Singular Values',fontsize=15)
plt.subplot(1,3,3)
plt.plot(np.array(list(range(0,X.shape[1]))),VV[:,0:ss.shape[0]].real);
plt.xlabel('time',fontsize=15)
plt.ylabel('F(x,t)',fontsize=15)
```

```
plt.title('Temporal Modes',fontsize=15)
plt.tight_layout()
matplotlib.rc('xtick', labels=12)
matplotlib.rc('ytick', labels=12)
plt.show()
```

Singular values

```
plt.figure(figsize=(15,8))
plt.subplot(1,3,1)
plt.plot(ss,'ro');
plt.xlabel('Number of Singular Values',fontsize=15)
plt.ylabel('Singular Values',fontsize=15)
plt.subplot(1,3,2)
plt.plot((np.diag(ss))/(np.trace(np.diag(ss))),'o');
plt.xlabel('Number of Singular Values',fontsize=15)
plt.ylabel('Normalized Singular Values',fontsize=15)
plt.subplot(1,3,3)
PercentSS=np.diag((np.diag(ss))/(np.trace(np.diag(ss))))*100
CumulSS=np.empty((PercentSS.shape[0],1))
CumulSS[0,0]=PercentSS[0].copy()
for i in range(1,PercentSS.shape[0]):
    CumulSS[i,0]=CumulSS[i-1,0]+PercentSS[i]
plt.plot(range(PercentSS.shape[0]),CumulSS,'bo');
plt.xlabel('Number of Singular Values',fontsize=15)
plt.ylabel('Cumulative energy in first r modes (%)',fontsize=15)
plt.tight_layout()
matplotlib.rc('xtick', labels=12)
matplotlib.rc('ytick', labels=12)
plt.show()
```

Applying the DMD

```
dmd = DMD(svd_rank=0,opt=True)
dmd.fit(X.T)
```

Eigenvalues on the unit circle

```
for eig in dmd.eigs:
    print('Eigenvalue: distance from unit circle '.format(eig, np.abs(eig.imag**2+eig.real**2 - 1)))
dmd.plot_eigs(show_axes=True, show_unit_circle=True)
```

Plotting DMD modes and dynamics

```
plt.figure(figsize=(16,8))
plt.subplot(1,2,1)
for mode in dmd.modes.T:
    plt.plot(x, mode.real)
    plt.title('Modes')
plt.show()
plt.subplot(1,2,2)
for dynamic in dmd.dynamics:
    plt.plot(t, dynamic.real)
    plt.title('Dynamics')
plt.show()
```

Plotting the DMD reconstructed data

```
fig = plt.figure(figsize=(17,6))
titles = ['Detected mode 1', 'Detected mode 2']
```

```

for n,title, mode, dynamic in zip(range(131, 133),titles, dmd.modes.T, dmd.dynamics):
    plt.subplot(n)
    plt.pcolor(xgrid, tgrid, (mode.reshape(-1, 1).dot(dynamic.reshape(1, -1))).real.T)
    plt.title(title,size=15)
plt.subplot(133)
plt.pcolor(xgrid, tgrid, dmd.reconstructed_data.T.real)
plt.title('Reconstructed F(x,t)',size=15)
matplotlib.rc('xtick', labels=12)
matplotlib.rc('ytick', labels=12)
plt.colorbar()
plt.show()

```

Frequency distribution of the absolute DMD reconstructed data error by Histogram

```

import scipy.special
from bokeh.layouts import gridplot
from bokeh.plotting import figure, show
from bokeh.io import output_notebook
from bokeh.models import NumeralTickFormatter
from bokeh.models import ColumnDataSource, ranges, LabelSet
from bokeh.layouts import layout
output_notebook()
RecError=dmd.reconstructed_data-X.T
RecErrorPer=np.absolute(RecError.flatten()) # Error data set
hist, edges = np.histogram(RecErrorPer, density=False, bins=10 )
p=figure(title="")
p.quad(top=hist, bottom=0, left=edges[:-1], right=edges[1:],fill_color="navy", line_color="black", alpha=.7)
p.y_range.start = 0
p.xaxis.axis_label = 'Absolute DMD Reconstructed Data Error '
p.yaxis.axis_label = 'Count'
p.title.align = "right"
p.yaxis[0].formatter = NumeralTickFormatter(format="0")
p.ygrid.grid_line_alpha = 0.5
p.ygrid.grid_line_alpha = 0.5
p.ygrid.grid_line_dash = [6, 4]
p.xgrid.grid_line_dash = [6, 4]
source = ColumnDataSource(dict(x=[round(((edges[i+1]-edges[i])/2)+edges[i],15) for i in range(edges.shape[0]-1)],y=hist,z=[str(round(i,2))+'% ' for i in (hist/np.sum(hist))*100]))
labels = LabelSet(x='x', y='y', text='z', level='glyph',x_offset=-10, y_offset=0,source=source,
render_mode='canvas',text_font_size='8pt')
p.add_layout(labels)
show(gridplot([p],ncols=2,plot_height=400,plot_width=600, toolbar_location=None));

```

The distribution of the absolute DMD reconstructed data error

```

def ecdf(data):
    """ Compute ECDF """
    x = np.sort(data)
    n = x.size
    y = np.arange(1, n+1) / n
    return(x,y)
from bokeh.plotting import figure, show
from bokeh.io import output_notebook
from bokeh.models import Label
from bokeh.models import Legend, LegendItem
output_notebook()
Ex,Ey = ecdf(np.absolute(RecError.flatten()))
p2=figure()
p2.circle(x=Ex, y=Ey,size=10, color='navy', alpha=0.7)
p2.title.text = "

```

```

p2.xaxis.axis_label = 'Absolute DMD Reconstructed Data Error '
p2.yaxis.axis_label = 'Absolute Error Distrubution'
p2.ygrid.grid_line_alpha = 0.5
p2.ygrid.grid_line_alpha = 0.5
p2.ygrid.grid_line_dash = [6, 4]
p2.xgrid.grid_line_dash = [6, 4]
#P20
p2.line([Ex[np.where(Ey> 0.2)[0][0]-1],Ex[np.where(Ey> 0.2)[0][0]-1]],[0,1],line_dash="4
4",color='black',line_width=2,legend_label='P20:'+str(round(Ex[np.where(Ey> 0.2)[0][0]-1].real,15)))
P20 = Label(x=Ex[np.where(Ey > 0.2)[0][0]-1], y=1, text='P20',x_offset=-20, y_offset=0,text_font_size='10pt')
p2.add_layout(P20)
#P50
p2.line([Ex[np.where(Ey > 0.5)[0][0]-1],Ex[np.where(Ey > 0.5)[0][0]-1]],[0,1],line_dash="4
4",color='black',line_width=2,legend_label='P50:'+str(round(Ex[np.where(Ey > 0.5)[0][0]-1].real,15)))
P50 = Label(x=Ex[np.where(Ey > 0.5)[0][0]-1], y=1, text='P50',x_offset=-10, y_offset=0,text_font_size='10pt')
p2.add_layout(P50)
#P90
p2.line([Ex[np.where(Ey > 0.9)[0][0]-1],Ex[np.where(Ey > 0.9)[0][0]-1]],[0,1],line_dash="4
4",color='black',line_width=2,legend_label='P90:'+str(round(Ex[np.where(Ey > 0.9)[0][0]-1].real,15)))
P90 = Label(x=Ex[np.where(Ey > 0.9)[0][0]-1], y=1, text='P90',x_offset=0, y_offset=0,text_font_size='10pt')
p2.add_layout(P90)
p2.legend.location = 'bottom_right'
p2.legend.background_fill_color = "#fefefe"
show(gridplot([p2],ncols=2,plot_height=400,plot_width=400, toolbar_location=None));

```

Chapter 3: DMD reconstruction, interpolation and extrapolation codes

Importing processed data from data base

```

%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np
import os
from pydmd import DMD
os.chdir('C:\\Users\\ext_mosh\\Desktop\\MasterThesisData\\AnnulusFluidVelocityFolder\\AnnulusFluidVelocity
EligExcel\\RawData') # my database address
AnnFluidVel=np.genfromtxt("AnnulusFluidVelocity.txt",delimiter=',')
Time=np.genfromtxt("SamplingTime.txt",delimiter=',')
MD=np.genfromtxt("PreparedCurvilinear.txt",delimiter=',')

```

Determining the data which going to be used for analysis

```

start=45          # starting time snapshot
step=225         # Number of Column or timestep

SelectedMatrix=AnnFluidVel[:,start:1669].copy()
X=SelectedMatrix[:,0:step:2].copy().T
TX=Time[start:start+step:2].copy()
t=TX

```

SVD

```

%matplotlib inline
import matplotlib
UU, ss, VV = np.linalg.svd(X.T,full_matrices=True);
plt.figure(figsize=(15,8))
plt.subplot(1,3,1)
plt.plot(MD[:,],UU[:,0:ss.shape[0]].real);

```

```

plt.xlabel('MD [m]',fontsize=14)
plt.ylabel('Annulus Fluid Velocity [m/S]',fontsize=14)
plt.title('Spatial Modes',fontsize=14)
plt.subplot(1,3,2)
plt.plot((np.diag(ss))/(np.trace(np.diag(ss))),'o');
plt.xlabel('Number of Singular Values',fontsize=14)
plt.ylabel('Normalized Singular Values',fontsize=14)
plt.subplot(1,3,3)
plt.plot(np.array(list(range(0,X.T.shape[1]))),VV[:,0:ss.shape[0]].real);
plt.xlabel('Time Snapshots',fontsize=14)
plt.ylabel('Annulus Fluid Velocity [m/S]',fontsize=14)
plt.title('Temporal Modes',fontsize=14)
matplotlib.rc('xtick', labels=12)
matplotlib.rc('ytick', labels=12)
plt.tight_layout()
plt.show()

```

Singular values

```

import matplotlib
plt.figure(figsize=(15,8))
plt.subplot(1,3,1)
plt.plot(ss,'ro');
plt.xlabel('Number of Singular Values',fontsize=15)
plt.ylabel('Singular Values',fontsize=15)
plt.subplot(1,3,2)
plt.plot((np.diag(ss))/(np.trace(np.diag(ss))),'o');
plt.xlabel('Number of Singular Values',fontsize=15)
plt.ylabel('Normalized Singular Values',fontsize=15)
plt.subplot(1,3,3)
PercentSS=np.diag((np.diag(ss))/(np.trace(np.diag(ss))))*100
CumulSS=np.empty((PercentSS.shape[0],1))
CumulSS[0,0]=PercentSS[0].copy()
for i in range(1,PercentSS.shape[0]):
    CumulSS[i,0]=CumulSS[i-1,0]+PercentSS[i]
plt.plot(range(PercentSS.shape[0]),CumulSS,'bo');
plt.xlabel('Number of Singular Values',fontsize=15)
plt.ylabel('Cumulative energy in first r modes (%)',fontsize=15)
plt.tight_layout()
matplotlib.rc('xtick', labels=12)
matplotlib.rc('ytick', labels=12)
plt.show()

```

Applying the DMD

```

dmd = DMD(svd_rank=-1,opt=True)
dmd.fit(X.T)

```

Eigenvalues on the unit circle

```

i=0
for eig in dmd.eigs:
    print('Eigenvalue {}: distance from unit circle {}'.format(eig, np.abs(eig.imag**2+eig.real**2 - 1)))
    print(i)
    i=i+1
print('total number of eigenvalue used: '+ str(i))
print('the energy used in reduced version is: '+str(CumulSS[i-1][0]))
dmd.plot_eigs(show_axes=True, show_unit_circle=True)

```

Plotting DMD modes and dynamics

```

%matplotlib inline
plt.figure(figsize=(15,8))
plt.subplot(1,2,1)
for mode in dmd.modes.T:
    plt.plot(MD[:,mode.real],label='Modes')
plt.title('Modes',fontsize=14)
plt.subplot(1,2,2)
for dynamic in dmd.dynamics:
    plt.plot(dmd.dmd_timesteps,dynamic.real,label='Dynamics')
plt.title('Dynamics',fontsize=14)
plt.tight_layout()
plt.show()

```

3D plotting of the DMD reconstructed data

```

from mpl_toolkits.mplot3d import axes3d
import matplotlib.pyplot as plt
from matplotlib import cm
Tgrid, MDgrid = np.meshgrid(dmd.dmd_timesteps, MD)
fig = plt.figure(figsize=(30,15))
ax = fig.gca(projection='3d')
surf = ax.plot_surface(MDgrid,Tgrid,dmd.reconstructed_data.real, rstride=8, cstride=8, alpha=0.8,
cmap=cm.bwr)
cbar=fig.colorbar(surf, ax=ax, shrink=0.5, aspect=10)
plt.gca().invert_xaxis()
ax.set_xlabel('MD [m]',fontsize=25,labelpad=30)
ax.set_ylabel('Time Snapshots',fontsize=25,labelpad=30)
ax.set_zlabel('Annulus Fluid Velocity [m/S]',fontsize=25,labelpad=60)
ax.set_title('DMD Reconstructed Data',fontsize=25,weight='bold')
ax.xaxis.set_tick_params(labelsize=25)
ax.yaxis.set_tick_params(labelsize=25)
ax.zaxis.set_tick_params(labelsize=25,pad=20)
cbar.ax.tick_params(labelsize=20)
plt.tight_layout()
plt.show

```

3D plotting of the original data

```

fig2 = plt.figure(figsize=(30,15))
ax2 = fig2.gca(projection='3d')
surf = ax2.plot_surface(MDgrid,Tgrid,X.T, rstride=8, cstride=8, alpha=0.8, cmap=cm.bwr)
cbar=fig2.colorbar(surf, ax=ax2, shrink=0.5, aspect=10)
plt.gca().invert_xaxis()
ax2.set_xlabel('MD [m]',fontsize=25,labelpad=30)
ax2.set_ylabel('Time Snapshots',fontsize=25,labelpad=30)
ax2.set_zlabel('Annulus Fluid Velocity [m/S]',fontsize=25,labelpad=60)
ax2.set_title('Original Data',fontsize=25,weight='bold')
ax2.xaxis.set_tick_params(labelsize=25)
ax2.yaxis.set_tick_params(labelsize=25)
ax2.zaxis.set_tick_params(labelsize=25,pad=20)
cbar.ax.tick_params(labelsize=20)
plt.tight_layout()
plt.show

```

DMD reconstructed data error

```

RecError=dmd.reconstructed_data-X.T
fig3 = plt.figure(figsize=(30,15))
ax3 = fig3.gca(projection='3d')

```

```

surf = ax3.plot_surface(MDgrid,Tgrid,RecError.real, rstride=8, cstride=8, alpha=0.8, cmap=cm.bwr)
cbar=fig3.colorbar(surf, ax=ax3, shrink=0.5, aspect=10)
plt.gca().invert_xaxis()
ax3.set_xlabel('MD [m]',fontsize=25,labelpad=30)
ax3.set_ylabel('Time Snapshots',fontsize=25,labelpad=30)
ax3.set_zlabel('Annulus Fluid Velocity Error [m/S]',fontsize=25,labelpad=55)
ax3.set_title('DMD Reconstructed Data Error',fontsize=25,weight='bold')
ax3.xaxis.set_tick_params(labelsize=25)
ax3.yaxis.set_tick_params(labelsize=25)
ax3.zaxis.set_tick_params(labelsize=25,pad=20)
cbar.ax.tick_params(labelsize=20)
plt.tight_layout()
plt.show

```

Distribution function

```

# Multiple distribution of error
def ecdf(data):
    """ Compute ECDF """
    x = np.sort(data)
    n = x.size
    y = np.arange(1, n+1) / n
    return(x,y)

```

frequency distribution of the absolute DMD reconstructed data error by Histogram

```

import numpy as np
import scipy.special
from bokeh.layouts import gridplot
from bokeh.plotting import figure, show
from bokeh.io import output_notebook
from bokeh.models import NumeralTickFormatter
from bokeh.models import ColumnDataSource, ranges, LabelSet
from bokeh.layouts import layout
output_notebook()
RecErrorPer=np.absolute(RecError.flatten()) # Error data set
hist, edges = np.histogram(RecErrorPer, density=False, bins=10)
p = figure(title="", tools=""); #title_location="left"
p.quad(top=hist, bottom=0, left=edges[:-1], right=edges[1:],fill_color="navy", line_color="black", alpha=.7)
p.y_range.start = 0
p.xaxis.axis_label = 'Absolute DMD Reconstructed Data Error [m/S]'
p.yaxis.axis_label = 'Count'
p.title.align = "right"
p.yaxis[0].formatter = NumeralTickFormatter(format="0")
p.ygrid.grid_line_alpha = 0.5
p.ygrid.grid_line_alpha = 0.5
p.ygrid.grid_line_dash = [6, 4]
p.xgrid.grid_line_dash = [6, 4]
source = ColumnDataSource(dict(x=[round(((edges[i+1]-edges[i])/2)+edges[i],20) for i in range(edges.shape[0]-1)],y=hist,z=[str(round(i,2))+'% ' for i in (hist/np.sum(hist))*100]))
labels = LabelSet(x='x', y='y', text='z', level='glyph',x_offset=-15, y_offset=0,source=source,
render_mode='canvas',text_font_size='8pt')
p.add_layout(labels)
show(gridplot([p],ncols=2,plot_height=400,plot_width=600,toolbar_location=None));

```

The distribution of the absolute DMD reconstructed data error

```

from bokeh.plotting import figure, show
from bokeh.io import output_notebook
from bokeh.models import Label

```



```

from bokeh.models import Legend, LegendItem
output_notebook()
Ex,Ey = ecdf(np.absolute(RecError.flatten()))
p2=figure()
p2.circle(x=Ex, y=Ey,size=10, color='navy', alpha=0.7)
p2.title.text = "
p2.xaxis.axis_label = 'DMD Reconstructed Data Error [m/S]'
p2.yaxis.axis_label = 'Absolute Error Distrubution'
p2.ygrid.grid_line_alpha = 0.5
p2.ygrid.grid_line_alpha = 0.5
p2.ygrid.grid_line_dash = [6, 4]
p2.xgrid.grid_line_dash = [6, 4]
#P20
p2.line([Ex[np.where(Ey> 0.2)[0][0]-1],Ex[np.where(Ey> 0.2)[0][0]-1]],[0,1],line_dash="4
4",color='black',line_width=2,legend_label='P20:'+str(round(Ex[np.where(Ey> 0.2)[0][0]-1].real,20)))
P20 = Label(x=Ex[np.where(Ey > 0.2)[0][0]-1], y=1, text='P20',x_offset=-20, y_offset=0,text_font_size='10pt')
p2.add_layout(P20)
#P50
p2.line([Ex[np.where(Ey > 0.5)[0][0]-1],Ex[np.where(Ey > 0.5)[0][0]-1]],[0,1],line_dash="4
4",color='black',line_width=2,legend_label='P50:'+str(round(Ex[np.where(Ey > 0.5)[0][0]-1].real,20)))
P50 = Label(x=Ex[np.where(Ey > 0.5)[0][0]-1], y=1, text='P20',x_offset=-10, y_offset=0,text_font_size='10pt')
p2.add_layout(P50)
#P90
p2.line([Ex[np.where(Ey > 0.9)[0][0]-1],Ex[np.where(Ey > 0.9)[0][0]-1]],[0,1],line_dash="4
4",color='black',line_width=2,legend_label='P90:'+str(round(Ex[np.where(Ey > 0.9)[0][0]-1].real,20)))
P90 = Label(x=Ex[np.where(Ey > 0.9)[0][0]-1], y=1, text='P90',x_offset=0, y_offset=0,text_font_size='10pt')
p2.add_layout(P90)
p2.legend.location = 'bottom_right'
p2.legend.background_fill_color = "#fefefe"
show(gridplot([p2],ncols=2,plot_height=400,plot_width=400, toolbar_location=None));

```

3D plotting of DMD data interpolation or extrapolation

```

dmd.dmd_time['dt'] = 0.5          *** this line should be active in case of interpolation***
#dmd.dmd_time['tend'] *= 1.5     *** this line should be active in case of extrapolation***
***for deactivate a comment, # sign should be typed at the beginning of a line***

```

```

TTgrid, MDDgrid = np.meshgrid(dmd.dmd_timesteps, MD)
fig = plt.figure(figsize=(30,15))
ax = fig.gca(projection='3d')
surf = ax.plot_surface(MDDgrid,TTgrid,dmd.reconstructed_data.real, rstride=8, cstride=8, alpha=0.8,
cmap=cm.bwr)
cbar=fig.colorbar(surf, ax=ax, shrink=0.5, aspect=10)
plt.gca().invert_xaxis()
ax.set_xlabel("MD [m]",fontsize=25,labelpad=30)
ax.set_ylabel("Time Snapshots",fontsize=25,labelpad=30)
ax.set_zlabel('Annulus Fluid Velocity [m/S]',fontsize=25,labelpad=60)
ax.set_title('DMD Data Interpolation',fontsize=25,weight='bold')
ax.xaxis.set_tick_params(labelsize=25)
ax.yaxis.set_tick_params(labelsize=25)
ax.zaxis.set_tick_params(labelsize=25,pad=20)
cbar.ax.tick_params(labelsize=20)
plt.tight_layout()
plt.show

```

3D plotting of the original data for comparing to DMD prediction

```

from mpl_toolkits.mplot3d import axes3d
import matplotlib.pyplot as plt
from matplotlib import cm

```

```

fig = plt.figure(figsize=(30,15))
ax = fig.gca(projection='3d')
surf=ax.plot_surface(MDDgrid,TTgrid,AnnFluidVel[:,start:start+(dmd.dmd_timesteps.shape[0])].copy(),
rstride=8, cstride=8, alpha=0.8, cmap=cm.bwr)
cbar=fig.colorbar(surf, ax=ax, shrink=0.5, aspect=10)
plt.gca().invert_xaxis()
ax.set_xlabel('MD [m]',fontsize=25,labelpad=30)
ax.set_ylabel('Time Snapshots',fontsize=25,labelpad=30)
ax.set_zlabel('Annulus Fluid Velocity [m/S]',fontsize=25,labelpad=60)
ax.set_title('Original Data',fontsize=25,weight='bold')
ax.xaxis.set_tick_params(labelsize=25)
ax.yaxis.set_tick_params(labelsize=25)
ax.zaxis.set_tick_params(labelsize=25,pad=20)
cbar.ax.tick_params(labelsize=20)
plt.tight_layout()
plt.show

```

3D plotting of DMD data interpolation error

```

PreError=dmd.reconstructed_data-AnnFluidVel[:,start:start+(dmd.dmd_timesteps.shape[0])].copy()
fig = plt.figure(figsize=(30,15))
ax = fig.gca(projection='3d')
surf = ax.plot_surface(MDDgrid,TTgrid,PreError.real, rstride=8, cstride=8, alpha=0.8, cmap=cm.bwr)
cbar=fig.colorbar(surf, ax=ax, shrink=0.5, aspect=10)
plt.gca().invert_xaxis()
ax.set_xlabel('MD [m]',fontsize=25,labelpad=30)
ax.set_ylabel('Time Snapshots',fontsize=25,labelpad=30)
ax.set_zlabel('Annulus Fluid Velocity Error [m/S]',fontsize=25,labelpad=55)
ax.set_title('DMD Interolated Data Error',fontsize=25,weight='bold')
ax.xaxis.set_tick_params(labelsize=25)
ax.yaxis.set_tick_params(labelsize=25)
ax.zaxis.set_tick_params(labelsize=25,pad=20)
cbar.ax.tick_params(labelsize=20)
plt.tight_layout()
plt.show

```

frequency distribution of the absolute DMD interpolated data error by Histogram

```

import numpy as np
import scipy.special
from bokeh.layouts import gridplot
from bokeh.plotting import figure, show
from bokeh.io import output_notebook
from bokeh.models import NumeralTickFormatter
from bokeh.models import ColumnDataSource, ranges, LabelSet
from bokeh.layouts import layout
output_notebook()
RecErrorPer=np.absolute(PreError.flatten()) # Error dataset
hist, edges = np.histogram(RecErrorPer, density=False, bins=10)
p = figure(title="", tools=""); #title_location="left"
p.quad(top=hist, bottom=0, left=edges[:-1], right=edges[1:],fill_color="navy", line_color="black", alpha=.7)
p.y_range.start = 0
p.xaxis.axis_label = 'Absolute DMD Interpolated Data Error [m/S]'
p.yaxis.axis_label = 'Count'
p.title.align = "right"
p.yaxis[0].formatter = NumeralTickFormatter(format="0")
p.ygrid.grid_line_alpha = 0.5
p.ygrid.grid_line_alpha = 0.5
p.ygrid.grid_line_dash = [6, 4]
p.xgrid.grid_line_dash = [6, 4]

```

```

source = ColumnDataSource(dict(x=[round(((edges[i+1]-edges[i])/2)+edges[i],20) for i in range(edges.shape[0]-1)],y=hist,z=[str(round(i,2))+%' for i in (hist/np.sum(hist))*100]))
labels = LabelSet(x='x', y='y', text='z', level='glyph',x_offset=-15, y_offset=0,source=source,
render_mode='canvas',text_font_size='8pt')
p.add_layout(labels)
show(gridplot([p],ncols=2,plot_height=400,plot_width=600, toolbar_location=None));

```

The distribution of the absolute DMD interpolated data error

```

from bokeh.plotting import figure, show
from bokeh.io import output_notebook
from bokeh.models import Label
from bokeh.models import Legend, LegendItem
output_notebook()
Ex,Ey = ecdf(np.absolute(PreError.flatten()))
p2=figure()
p2.circle(x=Ex, y=Ey,size=10, color='navy', alpha=0.7)
p2.title.text = ' '
p2.xaxis.axis_label = 'DMD Interpolated Data Error [m/S]'
p2.yaxis.axis_label = 'Absolute Error Distrubution'
p2.ygrid.grid_line_alpha = 0.5
p2.ygrid.grid_line_alpha = 0.5
p2.ygrid.grid_line_dash = [6, 4]
p2.xgrid.grid_line_dash = [6, 4]
#P20
p2.line([Ex[np.where(Ey> 0.2)[0][0]-1],Ex[np.where(Ey> 0.2)[0][0]-1]],[0,1],line_dash="4
4",color='black',line_width=2,legend_label='P20:'+str(round(Ex[np.where(Ey> 0.2)[0][0]-1].real,20)))
P20 = Label(x=Ex[np.where(Ey > 0.2)[0][0]-1], y=1, text='P20',x_offset=-20, y_offset=0,text_font_size='10pt')
p2.add_layout(P20)
#P50
p2.line([Ex[np.where(Ey > 0.5)[0][0]-1],Ex[np.where(Ey > 0.5)[0][0]-1]],[0,1],line_dash="4
4",color='black',line_width=2,legend_label='P50:'+str(round(Ex[np.where(Ey > 0.5)[0][0]-1].real,20)))
P50 = Label(x=Ex[np.where(Ey > 0.5)[0][0]-1], y=1, text='P20',x_offset=-10, y_offset=0,text_font_size='10pt')
p2.add_layout(P50)
#P90
p2.line([Ex[np.where(Ey > 0.9)[0][0]-1],Ex[np.where(Ey > 0.9)[0][0]-1]],[0,1],line_dash="4
4",color='black',line_width=2,legend_label='P90:'+str(round(Ex[np.where(Ey > 0.9)[0][0]-1].real,20)))
P90 = Label(x=Ex[np.where(Ey > 0.9)[0][0]-1], y=1, text='P90',x_offset=0, y_offset=0,text_font_size='10pt')
p2.add_layout(P90)
p2.legend.location = 'bottom_right'
p2.legend.background_fill_color = "#fefefe"
show(gridplot([p2],ncols=2,plot_height=400,plot_width=400, toolbar_location=None));

```

Pydmd package is used for implementing the DMD algorithm.(Demo et al., 2018)

References

- Bao, A., & Gildin, E. (2017). *Data-Driven Model Reduction Based on Sparsity-Promoting Methods for Multiphase Flow in Porous Media*. Paper presented at the SPE Latin America and Caribbean Petroleum Engineering Conference, Buenos Aires, Argentina. <https://doi.org/10.2118/185514-MS>
- Bjørkevoll, K. S. (2015a). *Use of High Fidelity Models for Real Time Status Detection with Field Examples from Automated MPD Operations in the North Sea*. Paper presented at the IFAC.
- Bjørkevoll, K. S. (2015b). *Use of High Fidelity Models for Real Time Status Detection with Field Examples from Automated MPD Operations in the North Sea*. Paper presented at the paper with review, presented at the 2nd IFAC Workshop on Automatic Control in Offshore Oil and Gas Production.
- Brunton, S. L., & Kutz, J. N. (2019). *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*: Cambridge University Press.
- Cayeux, E. (2018). *On the importance of boundary conditions for real-time transient drilling mechanical estimations*. Paper presented at the IADC/SPE Drilling Conference and Exhibition.
- Cayeux, E., Ambrus, A., Øy, L., Helleland, A., Brundtland, S. T., & Nevøy, H. (2020). *Analysis of Torsional Stick-Slip Situations Observed with Downhole High-Frequency Magnetometer Data*. Paper presented at the IADC/SPE International Drilling Conference and Exhibition.
- Cayeux, E., Shor, R., Ambrus, A., Pournazari, P., Ashok, P., & van Oort, E. (2018). From shallow horizontal drilling to ERD wells: How scale affects drillability and the management of drilling incidents. *Journal of Petroleum Science and Engineering*, 160, 91-105.
- Datta, B. N. (2010). *Numerical Linear Algebra and Applications, Second Edition*: Society for Industrial and Applied Mathematics (SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104).
- Demo et al. (2018). PyDMD: Python Dynamic Mode Decomposition. *Journal of Open Source Software*, 3(22), 530. doi:<https://doi.org/10.21105/joss.00530>
- Dvergsnes, E., & Cayeux, E. (2019). *On the Importance of the Coupling Between Transient Mechanical, Hydraulic and Thermal Effects for the Modelling of Real-Time Drilling Operations*.
- Eric, C., Per, S., Lars Jørgen, S., Håvard, U., & Espen, S. (2019). *Reconstruction of Pipe Movement from Downhole High Frequency Measurements*. Paper presented at the SPE Norway One Day Seminar.
- Foster, R., & Macmillan, R. (2018). *High speed telemetry on wired drill pipe, history, and impact on drilling process*. Paper presented at the Offshore Technology Conference.
- Gentle, J. E. (2012). *Numerical Linear Algebra for Applications in Statistics*: Springer New York.
- Israel, R., McCrae, D., Sperry, N., Gorham, B., Thompson, J., Raese, K., . . . Coit, A. (2018). *Delivering Drilling Automation II—Novel Automation Platform and Wired Drill Pipe Deployed on Arctic Drilling Operations*. Paper presented at the SPE Annual Technical Conference and Exhibition.
- Jovanović, M. R., Schmid, P. J., & Nichols, J. W. (2014). Sparsity-promoting dynamic mode decomposition. *Physics of Fluids*, 26(2), 024103.
- Kutz, J. N. (2013). *Data-driven modeling & scientific computation: methods for complex systems & big data*: Oxford University Press.

- Kutz, J. N., Brunton, S. L., Brunton, B. W., & Proctor, J. L. (2016). *Dynamic mode decomposition: data-driven modeling of complex systems*: SIAM.
- Macpherson, J., Roders, I., Schoenborn, K., Mieting, R., & Lopez, F. (2019). *Smart Wired Pipe: Drilling Field Trials*. Paper presented at the SPE/IADC International Drilling Conference and Exhibition.
- Motter, A. E., & Campbell, D. K. (2013). Chaos at fifty. *physics today*, 66(5), 27-33. doi:10.1063/PT.3.1977
- NOV, N. O. V. (2020). Intelliserv Wired Drill Pipe. Retrieved from <https://www.nov.com/products/intelliserv-wired-drill-pipe>
- Schils, S., Teelken, R., van Burkleo, B., Rossa, O. J., & Edwards, N. (2016). *The use of wired drillpipe technology in a complex drilling environment increased drilling efficiency and reduced well times*. Paper presented at the IADC/SPE Drilling Conference and Exhibition.
- Schmid, P. J. (2011). Application of the dynamic mode decomposition to experimental data. *Experiments in fluids*, 50(4), 1123-1130.
- Sethares, W. A. (2001). Repetition and pseudo-periodicity. *Tatra Mt. Math. Publ*, 23(1), 1-16.
- Silvester, I., Høgset, T., Torvund, S., & Saxena, S. (2020). *Qualification & Testing of a Powered Wired Drill Pipe Solution*. Paper presented at the IADC/SPE International Drilling Conference and Exhibition.
- Strang, G. (2016). *Introduction to Linear Algebra*: Wellesley-Cambridge Press.
- Sugiura, J., Samuel, R., Oppelt, J., Ostermeyer, G., Hedengren, J., & Pastusek, P. (2015). *Drilling modeling and simulation: Current state and future goals*. Paper presented at the SPE/IADC Drilling Conference and Exhibition.
- Temizel, C., Canbaz, C. H., Palabiyik, Y., Putra, D., Asena, A., Ranjith, R., & Jongkittinarukorn, K. (2019). *A Comprehensive Review of Smart/Intelligent Oilfield Technologies and Applications in the Oil and Gas Industry*. Paper presented at the SPE Middle East Oil and Gas Show and Conference.
- Tu, J. H. (2013). *Dynamic Mode Decomposition: Theory and Applications*. Princeton University,
- Tu, J. H., Rowley, C. W., Luchtenburg, D. M., Brunton, S. L., & Kutz, J. N. (2013). On dynamic mode decomposition: Theory and applications. *arXiv preprint arXiv:1312.0041*.
- Whittaker, A. (1985). *Theory and Applications of Drilling Fluid Hydraulics*: Springer Netherlands.