

# Data-Driven Multiscale Model Identification

A thesis submitted to the University of Sheffield for the degree of Doctor of  
Philosophy

**Christos Bartzis**

Department of Automatic Control and Systems Engineering

April 2022



The  
University  
Of  
Sheffield.

---



To my beloved parents



## Acknowledgements

First and foremost, I would like to express my unlimited respect and appreciation to my supervisor Professor Visakan Kadiramanathan for his amazing and unstoppable support, patience, guidance, encouragement and work ethics throughout my whole journey in my PhD years of study. A person that can inspire you and make you feel confident even under difficult circumstances and obstacles throughout all the research challenges all these years. I am grateful that I was given the opportunity to work in this level and explore research challenges and paths that I could not imagine and put myself into.

I would also like to express my respect and gratitude to my second supervisor Bryn Jones for being a catalyst with extremely helpful guidance, advice and review in critical moments of my research when an alternative approach was needed.

Many thanks to all the research team and colleagues for all the important group meetings, chats, advice and comments over the estimation and modelling methods that helped me overcome obstacles and think out of the box alternative solutions or paths on my research. Thank you Harry, Jinny, Anastasia, Ali, Shiv !

Special thanks to all my family and friends for the countless conversations and their mental support that helped me massively to move on and step by step fulfil my goals. The catalysts in my life, my dad Konstantinos, my mom Katerina, my sister Anastasia that believe in me more than I do to myself!! To all my friends that were supporting me from far away, Dimitris, Dionisis, Chris, Petros, Konstantinos, Vasilis, Efi, Ioanna and my special one Vicky !

## Abstract

In the modern industry, the design and manufacturing processes demand high product quality and lower production costs and times. Moreover, they require the combination of a variety of processes in different levels. These characteristics render them as complex systems. Therefore, the accurate understanding and representation of these systems behaviour and dynamics are essential for the development of new techniques aiming at improving the product quality and minimize the costs.

The precise mathematical description of these systems is of main challenge. The combination of system identification and model reduction techniques that provide full insight on the system behaviour using the minimum possible amount of data has gained much attention in recent years. The data driven modelling techniques offer flexibility and better accuracy when it comes to system identification. Thus, they render as attractive tools for system identification purposes.

These methods though, are prone to fail when it comes to models with abrupt changes and structures that are mixing high and low frequency effects, the so called multiscale models. This thesis addresses the existing data-driven algorithms by presenting various examples, justifying the necessity of the development of more efficient algorithms into this direction.

The first part of this thesis analyses the existing reduced order algorithms for data-driven models for temporal and spatio-temporal datasets. The dominant methods and their variations are presented and compared for specific examples that comprise multiscale characteristics. Reduced order models, such as Subspace identification method (SID) for one-dimensional data, Principal Component Analysis (PCA), Dynamic Mode Decomposition (DMD) and their extensions for both one and two-dimensional data will be addressed. The estimated dynamic systems are analysed and their performance is compared against the original datasets in each example respectively. Their inaccuracies and incapability of fully obtaining the system behaviour will be the foundation for the development of identification methods that are more accurate and capable of revealing all system characteristics.

The second part of this thesis proposes a novel multiscale reduced order POD and DMD method that gives full insight on complex system characteristics. These approaches take advantage of the wavelet decomposition method and divide the given datasets into different levels of resolution before applying the POD or DMD algorithm. By using the proposed algorithm, a novel system identification algorithm is formulated. Complex models with mixed frequency effects and abrupt changes can be estimated accurately. Due the wavelet decomposition properties, they are able to address complex struc-

tures into different levels of resolution. Therefore, they can reveal dynamics and model behaviours that cannot be represented from a single scale model efficiently but neither from the existing multiresolution algorithms.



# Contents

<b>List of Acronyms</b>	<b>i</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Research Aims . . . . .	5
1.3 Thesis Structure & Contribution . . . . .	6
1.3.1 Contribution . . . . .	6
1.3.2 Thesis Structure . . . . .	8
<b>2 Background to Signal Decomposition and System Identification</b>	<b>11</b>
2.1 Reduced Order Models . . . . .	12
2.1.1 Subspace Identification Method . . . . .	14
2.1.2 Subspace Identification Method For Fast-Sampled Datasets	16
2.1.3 Spatio-temporal System Identification Methods . . . . .	17
2.2 Conclusion . . . . .	23
<b>3 Identification of Temporal Multiscale Systems</b>	<b>25</b>
3.1 Introduction . . . . .	25
3.2 Subspace System Identification Algorithm . . . . .	26
3.2.1 $\delta$ -operator Subspace System Identification Algorithm .	27
3.3 SID of Stiff Systems . . . . .	31
3.3.1 Experimental Results & Simulations . . . . .	33
3.3.2 Noise Free Measurements . . . . .	33
3.3.3 Measurements With Added Noise . . . . .	38
3.4 Summary . . . . .	44
<b>4 Multiscale Spatio-temporal Signal Modelling</b>	<b>45</b>
4.1 Proper Orthogonal Decomposition . . . . .	46
4.2 Dynamic Mode Decomposition (DMD) . . . . .	47
4.2.1 Hankel DMD . . . . .	51

---

4.3	Multiresolution Model Representations . . . . .	52
4.3.1	Multiscale POD . . . . .	53
4.3.2	Multiresolution DMD . . . . .	56
4.4	Comparison of DMD Methods . . . . .	59
4.4.1	Periodical time series . . . . .	60
4.4.2	Sinusoidal time series . . . . .	63
4.4.3	Diffusion Equation Modelling . . . . .	65
4.4.4	Moving Image Data . . . . .	67
4.4.5	Multiresolution POD and DMD over moving image . .	70
4.4.5.1	Multiresolution DMD . . . . .	70
4.4.5.2	Multiresolution POD . . . . .	71
4.5	Conclusions . . . . .	71
<b>5</b>	<b>Spatial Multiscale DMD</b>	<b>73</b>
5.1	2-D Wavelet Decomposition . . . . .	74
5.2	Wavelet Decomposition DMD . . . . .	81
5.2.1	Analysis of the similarity between Hankel-DMD & Wavelet DMD . . . . .	84
5.3	Wavelet-DMD Examples . . . . .	87
5.3.1	Wave Diffusion Example . . . . .	88
5.3.1.1	Hankel - DMD . . . . .	89
5.3.1.2	Wavelet Decomposition DMD . . . . .	92
5.3.1.3	Proper Orthogonal Decomposition POD . . . . .	96
5.3.1.4	Wavelet Decomposition POD . . . . .	97
5.3.1.5	Conclusions . . . . .	98
5.3.2	Manufacturing of 3D Printing Powder Process . . . . .	98
5.3.2.1	Hankel-DMD . . . . .	100
5.3.2.2	Wavelet Decomposition DMD . . . . .	102
5.3.2.3	POD . . . . .	106
5.3.2.4	Wavelet Decomposition POD . . . . .	106
5.4	Summary . . . . .	109
<b>6</b>	<b>Results &amp; Discussions</b>	<b>111</b>
6.1	Wave Diffusion Simulation . . . . .	111
6.2	Flow of molten 3-D printing powder . . . . .	119
<b>7</b>	<b>Conclusions &amp; Future Work</b>	<b>129</b>
7.1	Summary of Contributions . . . . .	129
7.2	Future Work . . . . .	130
	<b>Appendices</b>	<b>133</b>

---

<b>A Orthogonality Properties</b>	<b>135</b>
<b>B Kalman Filtering &amp; Kalman Smoothing</b>	<b>139</b>
<b>C Parameter Estimation</b>	<b>145</b>
C.1 Autoregressive Model with exogenous input (ARX) . . . . .	145
C.2 Autoregressive moving-average with exogenous input (ARMAX)	146
C.3 Box-Jenkins (BJ) . . . . .	148
<b>D SVD Geometrical Derivation</b>	<b>151</b>
<b>E Combined (Stochastic - Deterministic) SID</b>	<b>155</b>
<b>F ERA</b>	<b>165</b>
<b>G 1-D Wavelet Transform</b>	<b>169</b>
G.0.0.1 Wavelet tree structure and multiresolution analysis . . . . .	170



# List of Figures

1.1.1 Fundamental Multiscale Model. Multi-physics hierarchy . . . .	2
2.1.1 Single and Multiscale System Identification Method Categories	12
2.1.2 SID (Left hand side): The (Kalman filter) states are estimated directly (either implicitly or explicitly) from input-output data, then the system matrices can be obtained. Classical approach (Right hand side): first obtain the system matrices, then estimate the states. . . . .	15
2.1.3 SID steps for state space matrices identification . . . . .	16
2.1.4 The Singular Value Decomposition (SVD) . . . . .	19
2.1.5 The Dynamic Mode Decomposition (DMD) . . . . .	20
3.3.1 Vehicle suspension & Physics Representation . . . . .	31
3.3.2 System response of the derived state space model using subspace system identification methods for $\mathbf{Fs} = 2^{16} \mathbf{Hz}$ and $\epsilon = 10^{-5}$ . . . . .	36
3.3.3 Poles of the derived state space model using subspace system identification methods for $\mathbf{Fs} = 2^{16} \mathbf{Hz}$ and $\epsilon = 10^{-5}$ . . . . .	36
3.3.4 Poles of the derived state space model using subspace system identification methods for $\mathbf{Fs} = 2^{16} \mathbf{Hz}$ and $\epsilon = 10^{-5}$ . . . . .	37
3.3.5 System response of the derived state space model using subspace system identification methods for $\mathbf{Fs} = 2^{12} \mathbf{Hz}$ and $\epsilon = 10^{-4}$ . . . . .	37
3.3.6 System response comparison of the derived state space model using $\delta$ subspace system identification methods for $\mathbf{Fs} = 2^{16} \mathbf{Hz}$ and $\epsilon = 10^{-5}$ & initial continuous system . . . . .	38
3.3.7 System response comparison of the derived state space model using $\delta$ subspace system identification methods for $\mathbf{Fs} = 2^{12} \mathbf{Hz}$ and $\epsilon = 10^{-4}$ & initial continuous system . . . . .	38
3.3.8 Singular values of the derived state space models for : $\mathbf{Fs} = 2^{16} \mathbf{Hz}$ , $N=200.000$ samples and $\text{SNR} = 70\text{db}$ . . . . .	42

---

3.3.9 Singular values of the derived state space model using subspace system identification methods for $F_s = 2^{13}$ Hz, $N = 20000$ Samples and $SNR = 70\text{db}$ . . . . .	42
3.3.10 Singular values of the derived state space model using subspace system identification methods for $F_s = 2^{12}$ , $\varepsilon = 10^{-4}$ and $SNR = 70\text{dB}$ . . . . .	43
3.3.11 System response comparison of the derived state space model using $\delta$ subspace system identification methods for $F_s = 2^{12}$ Hz and $\varepsilon = 10^{-4}$ & initial continuous system . . . . .	43
3.3.12 System response comparison of the derived state space model using $\delta$ subspace system identification methods for $F_s = 2^{16}$ Hz and $\varepsilon = 10^{-5}$ & initial continuous system . . . . .	43
4.3.1 Multiscale POD . . . . .	54
4.3.2 Multiresolution DMD . . . . .	58
4.4.1 Comparison of the given dataset and the output series of the derived dynamic model after applying the conventional DMD . . . . .	61
4.4.2 Comparison of the given dataset and the output series of the derived dynamic model after applying the Hankel-2 DMD . . . . .	61
4.4.3 Comparison of the output series of the derived dynamic model after applying Hankel-16 and Hankel-32 DMD . . . . .	62
4.4.4 Eigenvalues presentation of the derived dynamic model after applying Hankel-16 and Hankel-32 DMD . . . . .	62
4.4.5 Comparison of the output series of the derived dynamic model after applying Hankel-40 and Hankel-48 DMD . . . . .	63
4.4.6 Singular Value Decomposition plot of the given dataset . . . . .	64
4.4.7 Comparison of the given dataset and the output series of the derived dynamic model after applying the conventional DMD . . . . .	64
4.4.8 Comparison of the given dataset and the output series of the derived dynamic model after applying the Hankel-2 DMD . . . . .	65
4.4.9 Comparison of the MSE of output series between the derived dynamic models after applying the conventional DMD and the Hankel-2 DMD . . . . .	65
4.4.10 Eigendecomposition of the two derived dynamic models based on conventional DMD and the Hankel-2 DMD . . . . .	66
4.4.11 Comparison of the output MSE between the dynamic models derived from conventional DMD and Hankel-2 DMD . . . . .	67
4.4.12 Simulation of the model of small $3 \times 3$ image moving inside a $10 \times 10$ image . . . . .	68
4.4.13 Eigendecomposition plot of the conventional DMD & the Hankel-2 DMD methods . . . . .	68

---

4.4.14	Simulation of specific reconstructed snapshots of the moving image based on the standard DMD method . . . . .	68
4.4.15	Simulation of specific reconstructed snapshots of the moving image based on Hankel-2 DMD method . . . . .	69
4.4.16	Comparison of the Mean Square Error in the reconstruction step between standard DMD and Hankel - 2 DMD . . . . .	69
4.4.17	Comparison of the Mean Square Error in the reconstruction step of the mr-DMD using respectively . . . . .	70
4.4.18	Simulation of specific snapshots of the moving image by using the vanilla DMD method . . . . .	71
5.1.1	Simulation of a DWT of an image . . . . .	75
5.2.1	Example of a reconstructed Image based only on the Approximation Coefficients of the coarsest scale . . . . .	82
5.3.1	Presentation of the first 7 DMD mode after applying the Hankel-2 DMD . . . . .	91
5.3.2	Presentation of the DMD modes of the low frequency effects after applying the Hankel-2 DMD . . . . .	91
5.3.3	Presentation of the first 7 DMD mode in the first scale of decomposition using the Symlet mother wavelet, revealing only the fast dynamics . . . . .	93
5.3.4	Presentation of the first 7 DMD mode in the second scale of decomposition using the Symlet mother wavelet, revealing the slow dynamics . . . . .	93
5.3.5	Presentation of the first 7 DMD mode in the third scale of decomposition using the Symlet mother wavelet, revealing immediately the slow dynamics of these modes . . . . .	94
5.3.6	Presentation of the first 7 DMD mode in the forth scale of decomposition using the Symlet mother wavelet, revealing immediately the slow dynamics of these modes . . . . .	95
5.3.7	Conclusive presentation of the first DMD mode at the different scales of resolution . . . . .	95
5.3.8	Presentation of POD mode and their temporal evolution, revealing that as the number of POD modes increases, the modes are revealing fast dynamics . . . . .	96
5.3.9	Presentation of the first 4 POD modes in the third scale of the detail and approximation coefficients of the wavelet decomposition using the Symlet mother wavelet, revealing that in the third scale, the slow dynamics are revealed . . . . .	98

---

5.3.10	Presentation of the second POD mode of the detail and approximation coefficients of the wavelet decomposition using the Symlet mother wavelet, revealing that in the third scale, the slow dynamics are revealed . . . . .	99
5.3.11	Presentation of the last POD mode of the detail and approximation coefficients of the wavelet decomposition using the Symlet mother wavelet, revealing that in the third scale, the slow dynamics are revealed . . . . .	100
5.3.12	Simulation of the real-time process of the powder flow based on frames taken by high resolution camera . . . . .	101
5.3.13	Analysis of the Hankel-DMD . . . . .	101
5.3.14	Simulation of the particles flow . . . . .	102
5.3.15	SVD plots of the approximation and detail coefficients for the 3 level wavelet decomposition . . . . .	103
5.3.16	DMD modes of the approximation coefficients at coarser level of decomposition . . . . .	104
5.3.17	DMD modes of the detail coefficients at the coarser level 5 of decomposition . . . . .	105
5.3.18	Example of a specific DMD mode presenting the detail coefficients at all levels of decomposition . . . . .	106
5.3.19	Presentation of the most dominant POD modes . . . . .	107
5.3.20	Presentation of the dominant POD modes of the approximation coefficients in the coarsest level 5 . . . . .	107
B.1	A schematic presenting of the Kalman filter steps . . . . .	143
B.2	A schematic presenting the Kalman smoothing steps . . . . .	144
D.1	Projection of vector $a_i$ . . . . .	151
D.2	Projection . . . . .	152
D.3	The SVD of a $m \times n$ matrix . . . . .	154
G.1	A two-level signal breakdown into approximation and detail coefficients . . . . .	173
G.2	One Scale Image Wavelet Decomposition . . . . .	174

# List of Acronyms

**SVD** Singular Value Decomposition

**SID** Subspace System Identification

**ROM** Reduced Order Model

**POD** Proper Orthogonal Decomposition

**DMD** Dynamic Mode Decomposition

**PCA** Principal Component Analysis

**ERA** Eigenvalue Realisation Algorithm

**MR** Multiresolution Analysis

**ARX** Autoregressive Model with Exogenous Input

**ARMAX** Autoregressive Moving Average with Exogenous Input

**BJ** Box-Jenkins

**ML** Maximum Likelihood

**UKF** Unscented Kalman filter

**EKF** Extended Kalman filter

**RTS** Extended Kalman filter

**ERTS** Extended Rauch Tung Streibel

**URTSS** Unscented Rauch Tung Streibel

**SNR** Signal to Noise Ratio

**FEM** Finite Element Method

**MRA** Multiresolution Analysis

**PEM** Prediction Error Method

# Chapter 1

## Introduction

### 1.1 Motivation

The introduction and development of techniques that can provide better understanding of complex processes by deriving robust and simple mathematical model representations is the key motivation in this thesis. This task becomes arduous as these processes exhibit increased complexity, leading often to ill-conditioned circumstances during the model development. One form of complexity emerges from the multiscale nature and the amount of available data of the underlying processes.

Multiscale modelling was introduced in the field of physics and chemistry. It incorporates the fundamental structure of the laws of physics [1–5]. It aims at calculating the real system behaviour using information or models on different levels of resolutions. A simple example of the multiscale structure in the field of physics, as shown in Figure 1.1, indicates that the macroscale level models can empirically be estimated and originate from the level of continuum mechanics while the microscale models originate from the level of quantum mechanics.

The idea of the multiscale modelling commences from the fact that in certain occasions the available macroscale models are not accurate enough, and/or on the other hand, the microscale models are not efficient enough and/or offer too much information [4]. The complexity of a given system makes the estimation of an accurate model either in the macroscale or in the microscale level infeasible. Hence, the combination of both worlds can lead to more attractive models striking a balance between accuracy and computational efficiency.

Multiscale modelling introduces a novel perspective in system identification and parameter prediction. Multiscale modelling refers to a style where

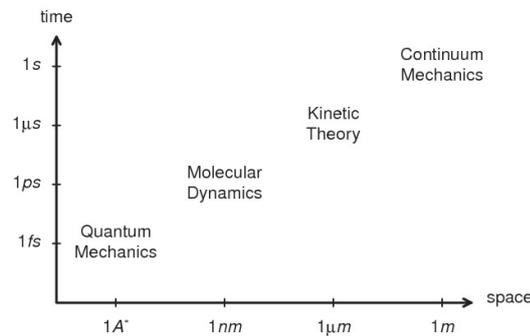


Figure 1.1.1: Fundamental Multiscale Model. Multi-physics hierarchy [8].

multiple models at different scales are used simultaneously to describe a system [3]. These scales are of different resolution and divide the signals into different scales, offering an attractive approach in model representation, both in time and frequency domain. These models are flexible since they capture the essential information, making system identification methods more efficient with attractive properties, such as low computational complexity and control actions at different scales. Furthermore, these structures provide improved abilities in noise filtering and error minimization.

Two types of multiscale models are found in nature. The type A models are the sequential multiscale models, where the macroscale model contains information which has been outsourced from the corresponding microscale model. This type of models are obtained in occasions where a limited amount of parameters are passed between the macroscale and the microscale world. This process is called parameter passing [8].

The second type of multiscale models is the Concurrent multiscale models. As the name of this type of models indicates, the macroscale and microscale models are used concurrently. When a smaller amount of parameters are needed in the macroscale world, they are computed on the fly on the microscale world and are embedded to the system. Hence, this concurrent coupling allows the evaluation of phenomena at the locations where they are needed [8].

The idea of multiscale structures came up in contexts where no multi-physics models are involved. Examples of classical multiscale algorithms are:

- Multi-grid method [6]. This is an algorithm for solving differential equations using a hierarchy of discretization. It is evolved in the class of multiresolution methods that are useful in obtaining the system behaviour in multiple scales. It comprises an iterative method that aims

at speeding up the convergence based on a global correction on the fine grid solution. The finite element method (FEM) is a typical example of the multigrid approach and is used for solving differential equations in the field of engineering and mathematical modelling. The FEM is used for solving partial differential equations in two or three space variables by dividing a large system into smaller, simpler parts. These parts are called finite elements.

- Fast multipole method [7]. The contribution of this method is to speed up the computation of  $f$  particle forces in the field of  $n$ -body problem or the field of electromagnetism by speeding up the solution of the iterative method of moments (MoM). It decomposes the system into different scales and each level of contribution is evaluated in a hierarchy of grids.
- Adaptive mesh refinement [9,10]. It comprises an alternative approach where the numerical grid or mesh is chosen adaptively. Based on a local error indicator from the current numerical solution, it modifies the mesh in order to fit a better numerical solution. The aim is to track features much smaller than the overall scale of the problem providing adequate higher spatial and temporal resolution where needed such as discontinuities, steep gradients, shocks, etc.
- Domain decomposition [11,12]. The computational domain is decomposed into several overlapping or non-overlapping subdomains. The numerical solution is obtained over the whole domain by iterating over the solutions on these subdomains.
- Multi-resolution representation [13,14,16]. This is a general strategy of decomposing functions or more generally signals into components at different scales. A well-known example is the wavelet representation. It was introduced by Stephane Mallat and Yves Meyer in 1989 and later on by Daubechies in 1992. Its predecessors were introduced in the micro-local analysis in the theory of differential equations and the pyramid methods of image processing and signal processing in 1981 by Peter J. Burt, Edward H. Adelson and James L. Crowley.

Multiscale modelling is often used to solve problems on multiple scales either temporal or spatial. A class of multiscale systems that will be addressed are called stiff systems. They are considered as a type of multiscale systems in the temporal domain, though no formal definition of what constitutes a stiff system exists. Stiff systems combine variables that respond very slowly

while some others respond very fast. Hence, a mathematical representation that includes the accurate estimation of all variables could be challenging. The stiff systems are considered as two-time scale multiscale systems.

Stiff systems can also be acknowledged as a class of differential equations for which certain numerical equation solving methods are unstable. The solution may arise when the size step is considered to be extremely small. Due to these characteristics, stiff system model estimation is challenging for computational and accuracy reasons hence model estimation over them will be of main interest [54, 55].

Another class of multiscale models in the spatial domain that will be addressed is called multiresolution analysis (MRA) [13]. This class of models provide compact representations for long-range statistical dependencies among far-apart variables by capturing such behaviour at coarser resolutions. The concept of multiresolution analysis is based on wavelet representations. The observed system is divided into multiple scales in time and frequency domain, in a hierarchical grading of time and frequency information. The advantage of the MRA is presented in both low and high frequency scales. In low frequency scales, there is information that tends to last and will not change quickly over time. On the other hand, in high frequency scales, there is less frequency resolution but there is information in smaller parts for data that are changing faster. Therefore, the use of the MRA can optimize the model representation.

The MRA and wavelets tend to represent a really good spectrogram. They are tailored to spend as much information in the different regions as necessary for the purpose of the application (i.e. low frequency does not require as much information, higher frequencies need more temporal resolution etc.)

On the other hand, it is not possible to coarse grain (simplify) everything, as it incurs a loss of information at each step. Coarse graining also involves the exchange of information between the fine scale and the coarse scale. This exchange of information between multiple scales leads to error propagation within the multiscale model, thus the accuracy and stability of the solution is affected [14]. The above suggests that a systematic approach to multiscale estimation which utilises the information across scales and is numerically conditioned, can lead to better algorithms. These specific algorithms increase estimation accuracy. Tools such as projection and model reduction could be used to reduce the complexity of the multiscale models in order to achieve an insightful representation and reveal the behaviour and system structure.

For instance, application fields such as biomechanics and fluid flow are of much attention in deriving multiscale models for defining the behaviour of complex systems. Examples such as the interpretation of the blood cells

behaviour or parts of the musculoskeletal system, constitute large range of spatial and temporal dimensions. Hence, the system representation based on a single space–time continuum is impossible and requires the consideration of multiple space–time continua, each representing the phenomenon of interest at a characteristic space–time scale. Consequently, the derivation of efficient algorithms that analyse and estimate these systems as well as overcome ill-conditioned phenomena due to system complexity is significant.

Nowadays, these high dimensional data are vast in every field, such as statistics, finance, computer vision and image processing for monitoring manufacturing processes. Therefore, reduced order representations of high dimensional dataset that introduce multiscale characteristics are of high interest in modern applications and a field that is not exploited extensively yet.

## 1.2 Research Aims

Modern manufacturing processes are governed by multiscale characteristics. The complexity of the systems described in the previous section, leads to estimation of models where the underlying system characteristics cannot be revealed. The estimation and system identification models aim at providing the most accurate representation of the system properties and underlying structures with the minimum computational cost. The consecutive need of implementing fast algorithms that combine parameters in different scales of resolution using the minimum amount of computational resources is a big challenge that has to be met.

The data-driven identification methods impose simple structure and estimate linear or non-linear systems via compact state space model representations. On the other hand though, as will be shown in latter chapters, these methodologies undergo inaccuracies, errors and limitations when it comes to data with abrupt changes or added noise. The overall aim of this thesis is the expansion and combination of multiscale structures in the field of data-driven estimation models that can lead into estimated linear model representations that can reveal hidden structures and characteristics.

Hence, the questions that will be addressed and answered in this thesis are :

- How to estimate a temporal multiscale model from observed data, avoiding ill-conditioning and with improved estimation accuracy compared to the existing methods when it comes to the limitations described above

- What insights can be obtained from the different approaches to multiscale spatio-temporal processes and what are the gaps and problem using these approaches
- How can multiresolution analysis be combined with mode decomposition approaches in order to provide insights into the complexity of the processes and derive the characteristics and system behaviour at minimum computational cost

Based on these questions, the objective of this project are:

- The derivation and implementation of multiscale algorithms applied in deterministic and stochastic temporal or spatio-temporal systems aiming at the accurate prediction and estimation of systems behaviour in the field of signal and image processing applied in manufacturing applications.
- The error reduction of the obtained model parameters that are used to reveal the different system dynamics and hidden structures for both linear and non-linear models.
- The expansion of existing frameworks that are implemented in both linear and non-linear cases.

Two specific examples will be exploited; one emerges in field of temporal data, exploiting the mathematical model of the car suspension where precise representation of the system dynamics is required. A second example will be addressed in the field of spatio-temporal data, and more precisely in the field of 3D printing powder manufacturing process where turbulences that occur in different areas affect the product quality. Aim is to reveal the model behaviour, aiming at optimizing the process for improved product quality.

## 1.3 Thesis Structure & Contribution

### 1.3.1 Contribution

The aim of this project is firstly to take a journey over the system identification methods for temporal and spatio-temporal dataset, by presenting the beneficial properties and, simultaneously, revealing the flaws of the existing approaches. A detailed analysis and comparison over examples of different nature, which is not exhibited in literature up to now, will be given. In the

field of multiscale modelling and estimation over both temporal and spatio-temporal data, an analytical investigation of the existing reduced order state space identification methods will be addressed.

For temporal models, known algorithms such as Subspace and Multiscale Subspace Algorithms (SID, ms-SID) [38, 41, 42, 62] or SID with pre-filtering for noise reduction will be analysed, presenting the improved accuracy and reduced computational cost of the derived models. For spatial data, POD and DMD algorithms and their extensions in the multiscale field will be introduced. The thesis will present various examples and evaluates the performance of these methods, revealing the connections among the algorithms and their flaws that emerge.

The ultimate goal is to propose a novel approach over the multiscale high-dimensional datasets that lead on more accurate system representations. Detailed simulated examples will be presented revealing the gaps in the existing identification methods, where the novel approaches will be evaluated over them.

The first part of the thesis is concentrated on the stiff temporal systems. The subspace identification method using the tool of  $\delta$ -operator over stiff systems for capturing fast sampling data will be presented. The  $\delta$ -operator SID method will try to link the discrete model with their continuous counterparts, when the sample rate is near zero. Moreover, it will be evaluated and compared to the current SID methods in literature in an effort to create free of ill-conditioned and non-convergence phenomena. An extensive analysis of existing SID methods over specific examples with multiscale nature will be presented and applied to the new identification algorithm.

In the second part of the thesis, the research will expand to high dimensional spatio-temporal data where computational issues and estimation accuracy problems emerge. The POD, DMD and their expansion for higher order systems will be evaluated. Moreover, the existing DMD approaches will be analysed over models governed by multiresolution characteristics. Up to now, there is no detailed analysis over the recent Hankel DMD, where the conventional DMD methods fail. Hence, the Hankel-DMD will be applied over various examples, presenting its beneficial properties.

In the last step, the thesis will present a novel multiscale framework for high dimensional multiscale spatio-temporal data, called wavelet-DMD. The new algorithm is tailored to overcome the computational boundaries of the existing DMD and POD algorithms. Several examples will be discussed, where the standard DMD and its extensions lack the ability to reveal hidden structures and separate the system multiscale dynamics. Up to now, there is no approach that is developed in order to reveal the system multiscale characteristics and the aim of the novel wavelet-DMD algorithm is to reveal

the locations where:

- The multiscale temporal and spatial features are exhibited.
- The temporal multiscale features within the derived multiscale modes are exhibited.
- The DMD efficiency at revealing coherent structures is significantly increased.

### 1.3.2 Thesis Structure

The rest of the thesis consists of the following chapters.

- Chapter 2

This chapter provides the literature review over the state of art data driven system identification methods for temporal and spatio-temporal models with multiscale properties. A review of the different reduction order models towards that direction will be recorded presenting the gaps in the literature that will be evaluated and addressed in latter chapter.

- Chapter 3

The first topic of the research focuses on data driven estimation and reduced order model methods for temporal datasets. More precisely, the thesis is dealing with subspace system identification methods (SID). This chapter introduces the main principals and properties of the SID. This is followed by the application of the SID in a stiff system that demonstrates the example of the car suspension. More precisely, the behaviour of SID will be evaluated on high sample rate occasions and noisy measurements. The  $\delta$ -operator SID will be proposed for these occasions, in an effort to address the SID problems for fast sampled datasets and also to link the discrete SID interpretations with their continuous counterparts for high sampled dataset.

- Chapter 4

This chapter deals with the modern reduced order models for both temporal and spatio-temporal high dimensional data. A full analysis of the Principal Component Analysis (PCA), the Dynamic Mode

Decomposition (DMD) and their extensions will be presented. These methods reveal dominant structures that emerge and provide a state space realization with the minimum possible information for any given dataset. The main focus is on examples with multiscale characteristics and abrupt changes, where the advantages and disadvantages in the performance of these methods are revealed. Up to now, there is no detailed comparison of these methods reported in literature. Furthermore, the results of the simulations will constitute the basis of the novel multiscale DMD and POD methods that will be proposed in the next chapter.

- Chapter 5

In this chapter, the research expands over multiscale spatio-temporal data, dealing with models and mathematical representations of complex manufacturing processes. Models with multiscale characteristics are simulated and a novel method is proposed. The new Wavelet Decomposition DMD combines the properties of the wavelet decomposition, where the data are divided into different scales of resolution, and the DMD algorithm. The goal of this algorithm is to separate the multiscale system features and provide full insight of the system behaviour. Two multiscale examples are presented and the new method is compared with the existing DMD methods.

- Chapter 6

This chapter demonstrates the examples of Chapter 5 extensively. Detailed simulations are presented in order to provide full insight of the Wavelet-DMD advantages over the existing methods.

- Chapter 7

This chapter summarizes all the work accomplished in this thesis, following by a discussion about the contribution and the future work that can potentially improve all referred methods in the field of reduced order modelling.



## Chapter 2

# Background to Signal Decomposition and System Identification

This project is dealing with reduced order techniques in the field of data-driven models. The system identification aims at forming simple mathematical model representations for dynamical systems derived by observed data without making any assumptions about the system dynamics. The system identification methods are following different approaches depending on the model class. These classes are separated into two categories, parametric or Non-parametric and linear or non-linear.

The parametric methods estimate model dynamics where a fixed number of parameters is used whereas the non-parametric models are more flexible in the number of the selected parameters. Non-parametric models do not make any particular assumptions about the kind of model structure. Therefore, they offer flexibility and better accuracy compared to the parametric ones. The non-parametric data driven models will be of main interest in this thesis.

This chapter is focused on providing the essential background to the existing identification and model reduction approaches. This chapter will also highlight the gaps in existing literature and justifies the proposed contributions, described in the previous chapter.

The first section 2.1, illustrates the different reduced order data-driven system identification methods in time domain providing an extended analysis of the subspace identification algorithm. In Section 2.2, a detailed presentation of the reduced order modelling methods for spatio-temporal dataset is given, presenting the literature of modern approaches such as the Proper Orthogonal Decomposition (POD) and Dynamic Mode Decomposition (DMD). All these methods will be extensively investigated in latter chapters where

their flaws will be revealed and the necessity of alternative approaches will be shown.

## 2.1 Reduced Order Models

As in classical estimation, the system identification methods can be broken into three different classes of problems [57]. Parameter and state estimation models originate from the known physics of the processes which give rise to models with known model structure and known or unknown parameters [8, 38]. Alternatively, if the physics of the processes are unknown, then mathematical model classes are considered for choosing a model structure towards the identification of the system [38].

A brief outline of the class of estimation methods for single scale and multiple scale modelling are identified below with detailed information provided.

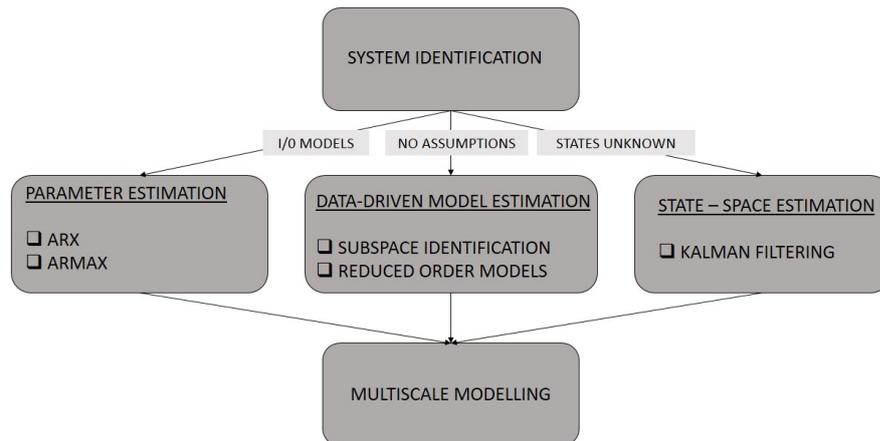


Figure 2.1.1: Single and Multiscale System Identification Method Categories

**Parameter estimation:** When the physics of the underlying process is known, but the parameters of the process are unknown, then a mathematical model structure of the process is selected but with the requirement of estimating its unknown parameters [38, 39]. While parameter estimation of stiff systems exists in the literature, models are either in continuous-time and estimated deterministically or in discrete-time in the difference equation form.

**State estimation:** Full knowledge of the underlying physics of the processes and the parameters combined with only partial observations of some

variables, means that state estimation has to be deployed to estimate the unobserved variables [47, 56, 67, 68]. Multiscale state estimation methods in the literature have made approximations that show performance loss when the number of scales, in which estimation is applied, increases [?, 1–3, 56].

**System identification:** In the absence of any known physics of an underlying process, system identification offers a systematic approach to construct mathematical models of its behaviour from observed input - output data [41, 42, 62].

One of the challenges even in classical system identification to accuracy in model parameter estimation is when the data are obtained by fast sampling or show complex characteristics that cannot be revealed into single scaled models. Hence, there is a need to incorporate methods that overcome fast sampling issues where possible and moreover reveal and separate multiscale structures among different scales of resolution.

Data-driven system identification models are quite attractive into this direction due to the fact that they do not require any priori information or knowledge of the underlying physics. This class of models are not iterative algorithms and their main objective is to utilize models by using the minimum possible amount of input data. These properties aim at computational efficiency, especially in occasions where the given process is governed by high dimensional data and multiscale characteristics. This class of models is called reduced-order models and have become quite popular due to the complexity of modern systems.

The introduction of reduced-order models (ROMs) for multiscale complex systems, concludes a wide range of engineering applications such as mechanism analysis, dynamic modelling, control law design and optimization [74]. Gaining a deep and accurate understanding of the behaviour over time and space of linear and non-linear structures requires a form of reduced order models or mode decomposition techniques due to the inefficient and high computational cost when dealing with complex and high dimensional data.

The determination of control laws for any system requires the knowledge of its dynamic model. The estimation of controllers for any system requires the derivation of a mathematical model that describes its dynamics. The majority of identification methods in literature, the so called predictor error methods are based on the approximate least absolute deviation criteria [40, 63]. These methods are iterative processes that aim to minimize the error between the actual and the estimated measures.

The disadvantage of these methods is that this iterative process can lead to problems such as numerical instability or lack of convergence and may get stuck into local minima [44, 64]. Moreover, in some applications, the computational cost of estimating the parameters of a mathematical model

using identification algorithms described above such as ARX, ARMAX is too expensive and complex, hence the use of these methods is improbable.

Reduced order models, such as the Subspace Identification Algorithm (SID), Eigenvalue Realisation Algorithm (ERA), Principal Component Analysis (PCA) and Dynamic Mode Decomposition (DMD) are data-driven models that appear to have common characteristics and are using the same mathematical tools described above.

### 2.1.1 Subspace Identification Method

The subspace identification algorithms (SID) is the first class of reduced order models that will be analysed. They have gained much attention due to their computational advantages over other estimation algorithms. Their simple structure and robustness in the parameter estimation of process models make them a efficient tool for estimating complex systems. The subspace identification methods are non-iterative algorithms that are based on mathematical tools such as SVD ( Singular Value Decomposition), QR decomposition and projection of matrices [38,41,42,62]. The purpose of subspace methods is the estimation of the model dynamics directly from input-output measurements without the prior knowledge of any system information. Hence, the identification problem becomes a linear least squares problem in the unknown system matrices. In contrast to the classical Prediction Error (PE) approaches that require a certain user-specified model parametrization, so-called canonical forms. The canonical forms can lead to numerically ill-conditioned problems, meaning that the parameters are extremely sensitive to small perturbations [42].

The use of mathematical tools such as QR decomposition and Singular Value Decomposition (SVD) do not increase computational costs and additionally increase numerical robustness and convergence. SVD is analysed in Appendix D and is shown that SVD is the main tools for detecting the optimum model order [38,62]. Thus, the term "subspace" is justified by the data compression step which can be referred as rank reduction, which is achieved by the SVD. The QR decomposition implementation is reassuring that ill-conditioning of the covariance matrices are concentrated in a certain triangular matrix. This ill-conditioning usually originates from ill-conditioned noise processes and from rounding errors [42].

The SID is stated for deterministic, pure stochastic models or most commonly for combined models due to the fact that most real-time models are corrupted with noise. The subspace algorithms contributed in a direction where the state sequences can be determined directly from the input-output observations in contrast to the classical approaches. Therefore, the use of the

Kalman filter states are also given directly from the input-output observations and the identification problem becomes a linear least squares problem in the unknown system matrices. The process and measurement noise covariance matrices follow from the least squares residuals [38, 62].

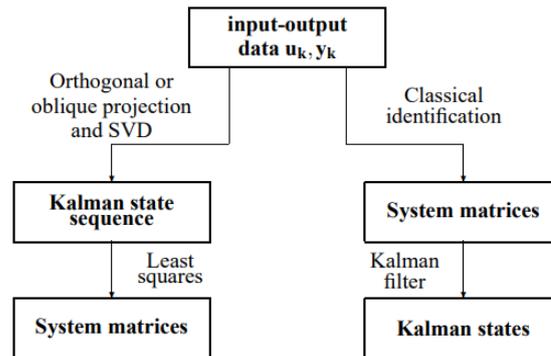


Figure 2.1.2: SID (Left hand side): The (Kalman filter) states are estimated directly (either implicitly or explicitly) from input-output data, then the system matrices can be obtained. Classical approach (Right hand side): first obtain the system matrices, then estimate the states.

The SID approaches originate from Ho and Kalman in 1966, where a scheme for recovering the system matrices from impulse response measurements is presented. In this approach, the SID stores the Markov parameters into a finite dimensional Hankel matrix where a state space realization of the LTI system is derived [33]. The first alterations of this approach were reported in 1974 and 1978 by McEwen and Kung respectively where the use of SVD was introduced as a tool to reduce the sensitivity to errors in the measured impulse response [34]. More recently, a number of algorithms further extending these ideas have been proposed, e.g. (Ljung 1991 [35, 36]), referred as 4SID methods, operating directly to input-output datasets.

All 4SID methods involve the extraction of the extended observability matrix from input-output data. All the approaches differ in the way the observability matrix is estimated, and also how the derived observability matrix is used for finding the system matrices (A, B, C and D). The most known subspace system identification algorithms are the numerical algorithms for subspace system identification (N4SID) and the multivariable output error state space model identification (MOESP). The N4SID approach was introduced by Van Overschee and De Moor and the MOESP approach by Verhaegen. The main difference between these two algorithms lie in the type of projection methods in the input-output matrices which are constituted of the

system input/output data. The N4SID is based on the oblique projection of the input data but the MOESP is particularly based on the orthogonal of the input data [64, 65]. Herein, these classes of 4SID methods, an estimate of an extended observability matrix is given. The latter matrix can then be used in various ways for obtaining a state-space model of the system under study.

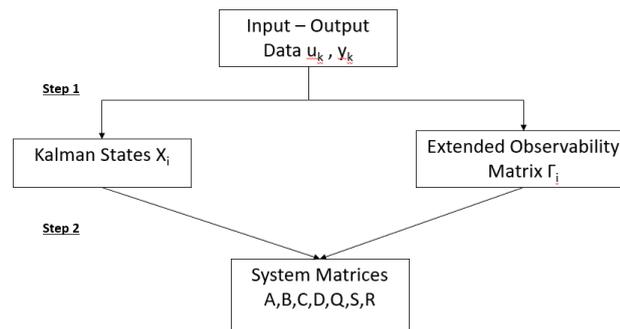


Figure 2.1.3: SID steps for state space matrices identification

There are several alternative approaches on ways to determine the order of a state-space model or revising existing approaches due to inaccuracies of the SID method for datasets with characteristics such as short samples or high dimensionality but will be not stated for the purpose of this thesis. The interest will be concentrated at revealing the flaws of the most known SID approaches for fast sampling data. There will be an attempt for the model fit of the SID to be optimized when it comes to fast-sampled data and moreover to link the discrete SID with the continuous counterpart as there is no stated connection of them in an attempt for a better interpretability.

### 2.1.2 Subspace Identification Method For Fast-Sampled Datasets

In the field of fast sampled datasets, the  $\delta$ -operator is introduced when the sampling time is close to zero. Up to date, these subspace alternatives are evaluated mostly on continuous models. In these methods, a method of identifying a continuous-time closed-loop system based on subspace methods is addressed. Herein, the closed-loop identification problem is transformed into an equivalent open-loop identification problem and then a  $\delta$ -operator subspace identification method for the equivalent open-loop one in order to obtain the estimate of open-loop plant model is developed [49, 50, 66].

The continuous-time systems are transformed to the  $\delta$ -operator state space models which converge to the original systems with the sampling period going to zero.

These approaches are focused on closed loop systems, as the majority of modern systems are of this nature. The identification of continuous-time systems is not always solved easily with the latest approach into this direction being addressed in 2021, in an attempt is based on fault detection [50].

These algorithms are based on the continuous subspace identification method which is not the direction of this thesis. On the other hand, there is a gap at addressing and comparing a discrete  $\delta$ -subspace identification method with the state of art subspace approaches, as a solution for fast sampled datasets. The application of the  $\delta$ -operator in the SID will be extensively addressed in the next chapter.

### 2.1.3 Spatio-temporal System Identification Methods

Several natural models are known to exhibit a spatio-temporal evolution pattern. The reduced order data-driven methods span the development of spatio-temporal models based on the compression of data into their most essential features that contain the maximum possible information. The addition of them in the identification methods entrains the derivation of accurate representations with the minimum loss of information and computational cost in favour of simplicity. Many techniques have been developed on this direction. The principal component analysis (PCA), Proper Orthogonal Decomposition (POD) and Dynamic Mode Decomposition (DMD) are most widely used approaches.

PCA is one of the oldest and its simple structure made it attractive for identification purposes. The PCA aims at reducing the dimensionality of a dataset, while preserving as much variability, by means of information, as possible. The term preservation of variability interprets into deriving new variables that are linear functions of those in the original dataset, that maximize variance and that are uncorrelated with each other. It aims at obtaining the system behaviour by extracting the dominant patterns or correlations among the given data. By finding the directions of the maximum variance of the input data, it provides a hierarchical coordinates system to represent the statistical variations in any data set  $\mathbf{X}$ .

It initiates by computing the covariance matrix of the input data which is a  $m \times m$  symmetric matrix (where  $m$  is the number of dimensions) that has as inputs the covariances associated with all possible pairs of the initial variables.

$$\mathbf{C} = \mathbf{X}^T \mathbf{X} / (n - 1) \quad (2.1.1)$$

The sign of the covariance indicates that if:

- $\mathbf{C} > 0$  : the two variables increase or decrease together (correlated)
- $\mathbf{C} < 0$  : one variable increases when the other decreases (inversely correlated)

The PCA computes the eigenvalues and eigenvectors of the computed covariance matrix in order to determine the dominant structures.

$$\mathbf{C} = \mathbf{V} \mathbf{\Lambda} \mathbf{V}^T \quad (2.1.2)$$

where  $\mathbf{V}$  is an eigenvector matrix (each column is an eigenvector) and  $\mathbf{\Lambda}$  is a diagonal matrix with eigenvalues  $\lambda_i$  which are arranged in a decreasing order.

The eigenvectors indicating where the most variance (most information concentration) is shown, are called Principal Components. The eigenvalues indicate the coefficients attached to eigenvectors, which give the amount of variance carried in each principal component.

By that process, the PCA is squeezing the data in order to derive the maximum possible information of the initial given data by using only the first components. Hence, the components with the maximum variance are determined, meaning the maximum information in order not to sacrifice the accuracy in favour of reducing the amount of information.

The equation 2.1.2 links the PCA and SVD. PCA is closely related to the SVD where the actual SVD is the PCA. SVD as mentioned in the previous section is equal to:

$$\mathbf{X} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T$$

Substituting the SVD equation to the covariance equation 2.1.1 of the PCA, it gives:

$$\mathbf{C} = \mathbf{V} \mathbf{S} \mathbf{U}^T \mathbf{U} \mathbf{S} \mathbf{V}^T / (n - 1) = \mathbf{V} \frac{\mathbf{S}^2}{n - 1} \mathbf{V}^T \quad (2.1.3)$$

Hence, the right singular vectors  $V$  are principal directions and that singular values are related to the eigenvalues of covariance matrix via  $\lambda_i = s_i^2/(n-1)$ .

PCA and SVD are similar but they are not comparable. SVD comprises a more general method and can be applied over non-square matrices, indicating that any arbitrary matrix can be decomposed to an orthogonal matrix (rotation), a diagonal matrix (stretch) and another orthogonal matrix [95]. The use of SVD to perform PCA, numerically, is more plausible than working over the covariance matrix that can cause loss of precision.

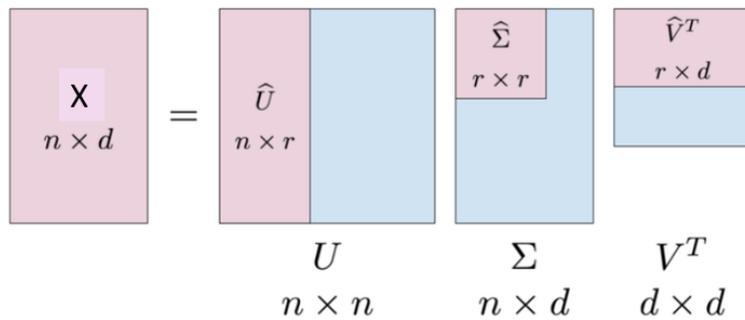


Figure 2.1.4: The Singular Value Decomposition (SVD) [79]

The first modal decomposition technique is called Proper Orthogonal Decomposition (POD) and was introduced in fluid flow dynamics in 1967 by Lumley as an attempt to decompose the random vector fields representing turbulent fluid motion into a set of functions that capture some portion of the total fluctuating kinetic energy in the flow [78]. The POD is also known as the Karhunen-Loeve decomposition (KLD) and for finite dimensional cases which are truncated after a few terms, the POD is equivalent to principal component analysis (PCA) [80].

The proper orthogonal decomposition (POD) is a multi-variate statistical method that aims at obtaining a compact representation of the given data. This method may serve two purposes, the order reduction by projecting high-dimensional data into a lower-dimensional space and feature extraction by revealing relevant structures hidden in the data.

The key idea of the POD is the reduction of a large number of variables to a much smaller amount of uncorrelated variables while retaining as much as possible of the variation-information in the original variables. An orthogonal transformation to the basis of the eigenvectors of the sample covariance

matrix is performed, and the data are projected onto the subspace spanned by the eigenvectors corresponding to the largest eigenvalues. This transformation decorrelates the signal components and maximizes variance [79].

Due to the large amount of computations required to derive the POD modes, the technique was unused until the middle of the 20th century. The POD has now gained popularity and is being used in numerous fields. The fundamental use of POD comes in the context of turbulences, extracting coherent structures [80]. POD was raised in the field of signal processing and was firstly suggested for the detection of the number of signals in a multichannel time-series [81]. Additionally, in the field of chemistry, POD modes were used to capture the reaction-diffusion of chemical processes [83]. POD also gained a lot of attention in the field of fluid flows, starting from 1995 in an attempt to describe quantitative changes in spatial complexity during extended episodes of ventricular fibrillation [84] and obtaining reduced-order models of unsteady viscous flows [85]. POD was also employed for forecasting in meteorology [86] and as a means of classifying speech data [87].

Additional applications of the POD are recorded in the 1990s in the field of structural dynamics for determining low-dimensional models of distributed systems [88], the control of self-excited vibrations of long torsional strings [89], creation of low-dimensional models of an overhung rotor [90], damage detection [91], finite element modelling [92], multi-body systems [93], stochastic structural dynamics [94] etc.

An extension of POD which was appeared in 2008, is called the Dynamic Mode Decomposition (DMD) [69]. The DMD is a data-driven model which, like the POD, does not rely on any prior assumptions or knowledge of the dynamics of the system. DMD has the ability to extract the coherent structures that dominate the observed spatio-temporal data and obtains linear reduced order models based on the eigendecomposition of the dominant modes [102].

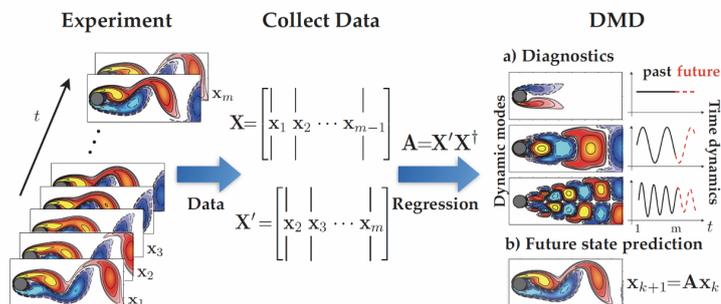


Figure 2.1.5: The Dynamic Mode Decomposition (DMD) [72]

In contrast to the POD, DMD modes are based on their dynamics rather than the energy content. Hence, linear and non-linear systems are determined by a superposition of modes whose dynamics are governed by the eigenvalues, describing a low - order dynamic model. Additionally, other dimensionality reduction methods such as Principal Component Analysis (PCA) or Proper Orthogonal Decomposition (POD), lack temporal representation and the fact that POD arranges modes in the order of energy contents, and not in the order of the dynamical importance which is accomplished through DMD. Therefore, DMD can be efficient in the identification of dominant frequencies. In occasions where the system contains relatively small energy, but is strongly connected with other structures sharing the same frequency, these structures are likely to be ignored by POD analysis but would be captured by the DMD [82].

The dynamic mode decomposition (DMD) is applied in both temporal or spatial domain, covering various aspects of data processing for experimental measurements, such as, the low-dimensional representation of a dynamic process, the filtering of data based on structural and dynamic coherence, and the recovery of data from gappy measurements or signals [106].

The DMD shows a vast range of applications, especially in several flow configurations. For instance, DMD has been applied in the study of the wake behind a flexible membrane [73], the flow around high-speed trains [103], instabilities in annular liquid sheets [104], shock wave-turbulent boundary layer interactions [105] and various jets [70, 102, 106–108]. There have also been a number of efforts regarding the numerics of the DMD algorithm, including the development of memory-efficient algorithms [110] and error analysis of DMD growth rates [111]. Variants of the DMD algorithm have also been proposed, including optimized DMD [112], optimal mode decomposition [113], multi-resolution dynamic mode decomposition [99] and higher order DMD [76, 77]. DMD is also been used extensively in modelling complex systems in the life sciences, such as brain neural activity [114], blood flow [115] and many more applications.

The DMD apart from the connection with POD and PCA, it is connected with other well-known reduced order methods, such as the ERA which is presented in Appendix. The ERA is a control-theoretic method for system identification of linear systems. In applications where the same datasets are used, the DMD eigenvalues reduce to poles of an ERA model [70].

In latter chapter, an extended analysis and comparison of the dominant DMD approaches will take place. In literature, there are many applications of these modal reduction methods by up to now, there is no straight comparison of the various approaches over examples that show linear dependencies and oscillatory patterns in the multiscale field. Modern applications and

processes are complex, governed by multiscale characteristic, hence the determination of the performance of these methods is of quite interest.

These data-driven algorithms are applied in complex models which often introduce multiscale characteristics or high-dimensional data. Approaches that take advantage of the multiresolution analysis (MRA) that was presented in Chapter 1 and the POD or DMD algorithms have come up recently. The first approach that was encountered in the multiresolution DMD (mr-DMD) by N.Kutz [99]. It was introduced in the field of video analysis and background noise removal by observing and separating the slow-moving and fast-moving modes. The mr-DMD is performing the DMD into segments of the given frames. In each step-scale, the length of the window over which DMD is applied is repeatedly halved, and the low frequency components of each iteration are separated from the rest of the dynamics. The process is repeated in an attempt to separate the slowest modes in terms of their frequency.

An extension of the mr-DMD is called windowed-DMD which is a generalization of the mr-DMD in the direction to deal with a broader variety of multiscale systems and reconstruct their isolated components more successfully. In contrast to the mr-DMD, this approach takes advantage of DMD sensitivity to the duration and sampling rate of the time series inputs [75]. The key difference is that the segments where the DMD is applied in the mr-DMD approach are limited to some base time span and power of-two subdivisions. This fixed segmentation can be problematic in systems whose frequency content does not follow that pattern. Therefore, the windowed DMD was introduced as an optimization of the mr-DMD, using sliding and overlapping windows that can cover the models described.

The most recent approach in the multiresolution analysis field is called multiresolution POD (mPOD) which combines the Multiresolution Analysis (MRA) with a standard POD. In this approach, the mPOD splits the correlation matrix of the input data into the contribution of different scales, retaining non-overlapping portions of the correlation spectra. Thus, by utilising the standard POD, the mPOD extracts the optimal basis from each scale [95]. The difference with the previous multiresolution algorithms, is that the actual multiresolution analysis is based on 1-D and 2-D filter banks via the discrete wavelet transform. The mPOD will be extensively analysed in Chapter 4 and is of main interest for the purpose of this thesis. The mPOD is the closest related algorithm to the novel Wavelet-DMD approach that will be introduced in this thesis.

## 2.2 Conclusion

This chapter has provided a background in the state of art reduced order identification methods for both temporal and spatio-temporal models. These models are the foundation of the derived ideas and algorithms that will be proposed and analysed in the next chapters. Moreover, the connection among all the different data driven identification methods is presented which are based on the same mathematical tools.

More precisely, in the temporal domain, the evolution and background of the subspace identification algorithms is provided, leading to their application in the modern complex systems. The subspace identification algorithms are shown to be robust and many alternatives are proposed. On the other hand, special attention requires to be given for fast sampled data where there is no extensive literature and comparison of the SID alternatives for this type of data with the existing SID methods.

In the field of modal analysis for temporal and spatio-temporal datasets, the most modern approaches of POD, DMD and their alternatives are introduced for single and multiscale models. These approaches will be the foundation of the extended analysis and comparisons in the latter chapters.



# Chapter 3

## Identification of Temporal Multiscale Systems

### 3.1 Introduction

The multiscale systems are inextricably linked to the stiff systems that are excessively appeared in industrial applications such as in chemical, biomedical and mechanical engineering. Various parameter and state estimation algorithms are established for identification purposes but the computation cost and complexity of stiff systems, makes the identification slow, inaccurate and in some cases even impossible. By definition, stiff systems combine slow and fast moving variables. Hence an additional problem of the fast sampling data arises. Ill-conditioned system dynamic matrices are estimated when defined for data sampled at sampling rates that are high relative to the dynamics of the underlying continuous-time processes being sampled. The practical difficulty in stiff systems is that the existing methods cannot resolve the microscopic-fast variables and thus model errors through various parametrizations are unavoidable when applied on the macroscopic-slow level [52, 55]. As the sampling rate increases, the covariance matrix of the observed data becomes increasingly stiff and therefore numerical conditioning becomes a serious problem [66]. A related difficulty occurs in the representation of sampled data by linear state-space models due to the increase of sampling rate.

In this chapter, a full analysis of the subspace identification (SID) method under the special occasions described above is presented. Full results of the SID approach are given over an example where the dynamics and behaviour are known. Moreover, an alternative approach introducing the  $\delta$ -operator to the input data to the SID approach is analysed and compared to the standard

SID over the same example. The aim to reveal whether the  $\delta$  algorithm can minimize the estimation error but moreover link accurately the discrete time approaches to the continuous counterparts which is not addressed up to date.

## 3.2 Subspace System Identification Algorithm

The main topic of this thesis is the investigation of existing methods and derivation of reduced order data-driven system identification methods with increased accuracy over stiff systems. In this chapter, the subspace identification method (SID) will be applied over stiff systems and will be the main topic of research. As mentioned in Chapter 2, the SID is a powerful identification method and has gained much attention due to its simple structure and the connection with fundamental methods such as ERA and the use of simple and powerful tools such as SVD and projection properties.

As stated, there are various approaches based on the nature of the input-output data. With the majority of modern system being complex, the SID method that was selected to be investigated for the purpose of this thesis is the combined deterministic- stochastic SID [62]. Therefore, the SID method will be exposed to an example with multiscale characteristics for both, noisy-free and corrupted with noise measurements. In order to add an extra level of difficulty, the simulations will take place for different sampling rates too. Thus, the performance of the SID will be tested under extreme conditions. The alternative  $\delta$ -SID will be compared with the existing discrete SID methods in order to address the necessity of an alternative SID approach for datasets when sampled time is close to zero. The  $\delta$ -SID will also be compared with the initial continuous model in an attempt to address if this approach can link the discrete model to the continuous counterpart.

The following algorithm illustrates the combined deterministic- stochastic SID which will be used for this example. As shown in Appendix E, the input-output data are formed into Hankel matrices, based on equations (E.5), (E.6) and (E.7). The order reduction is achieved based on the SVD over the oblique (MOESP) or orthogonal projection (N4SID) (see Equation (E.23)). As a next step, the extended observability matrix is derived, from where the state matrices A and C are estimated. Following that, the B and D matrices are determined based on least square regression and uncertainties based on Kalman filter when applicable.

**Algorithm 1** Robust combined algorithm [62]

1. Calculate the oblique and orthogonal projections:

$$\mathcal{O}_i = Y_f / U_f \mathbf{W}_p \quad , \quad \mathbf{Z}_i = Y_f / \begin{pmatrix} \mathbf{W}_p \\ U_f \end{pmatrix} \quad , \quad \mathbf{Z}_{i+1} = Y_f^- / \begin{pmatrix} \mathbf{W}_p^+ \\ U_f^- \end{pmatrix} .$$

2. Calculate the SVD of the weighted oblique projection (MOESP) or orthogonal projection (N4SID):

$$W_1 \mathcal{O}_i W_2 = USV^T .$$

3. Determine the order by inspecting the singular values in  $S$  and partition the SVD accordingly to obtain  $U_1$  and  $S_1$ . 4. Determine  $\Gamma_i$  and  $\Gamma_{i-1}$  as:

$$\Gamma_i = W_1 U_1 S_1^{1/2} \quad , \quad \Gamma_{i-1} = \underline{\Gamma}_i .$$

5. Solve the set of linear equations for  $A$  and  $C$  :

$$\begin{pmatrix} \Gamma_{i-1}^\dagger \cdot \mathbf{Z}_{i+1} \\ Y_{i|i} \end{pmatrix} = \begin{pmatrix} A \\ C \end{pmatrix} \Gamma_i^\dagger \cdot \mathbf{Z}_i + \mathcal{K} \cdot U_f + \begin{pmatrix} \rho_w \\ \rho_v \end{pmatrix} .$$

Recompute  $\Gamma_i$  and  $\Gamma_{i-1}$  from  $A$  and  $C$ . 6. Solve  $B$  and  $D$  from:

$$B, D = \arg \min_{B, D} \left\| \begin{pmatrix} \Gamma_{i-1}^\dagger \cdot \mathbf{Z}_{i+1} \\ Y_{i|i} \end{pmatrix} - \begin{pmatrix} A \\ C \end{pmatrix} \Gamma_i^\dagger \cdot \mathbf{Z}_i - \mathcal{K}(B, D), U_f \right\|_F^2$$

7. Finally, determine the covariance matrices  $Q, S$  and  $R$  as:

$$\begin{pmatrix} Q & S \\ S^T & R \end{pmatrix} = \mathbf{E}_j \left[ \begin{pmatrix} \rho_w \\ \rho_v \end{pmatrix} \cdot \begin{pmatrix} \rho_w^T & \rho_v^T \end{pmatrix} \right]$$

### 3.2.1 $\delta$ -operator Subspace System Identification Algorithm

Before the implementation of the SID algorithms, the  $\delta$ -SID will be analysed. As mentioned, it will be used when the datasets are governed by fast sampling rates which could lead in ill-conditioning in the covariance matrices of the state space representation. In that occasion, the signal can be reformulated via a delta, or divided-difference, dynamical operator and an alternative framework that ameliorates such ill-conditioning.

The  $\delta$  operator is formed in the SID method as follows. It is known that any discrete time state space model via zero-order hold is represented as follows:

$$qx(k) = A_q x(k) + B_q u(k) + w_\delta(k) \quad (3.2.1)$$

$$y(k) = C_q x(k) + D_q u(k) + e_\delta(k) \quad (3.2.2)$$

where the state space matrices are in the form :

$$A_q = e^{A\Delta T} \quad (3.2.3)$$

$$B_q = \int_0^{\Delta T} e^{A(\Delta T - \tau)} B d\tau \quad (3.2.4)$$

$$C_q = C \quad (3.2.5)$$

$$D_q = D \quad (3.2.6)$$

where A,B,C,D are the state matrices of the continuous model.

It is easily shown that when the sample time tends to be zero, meaning that the sampling frequency is high, the matrices  $A_q$  and  $B_q$  are in the following form:

$$\lim_{\Delta \rightarrow 0} A_q = I \quad (3.2.7)$$

$$\lim_{\Delta \rightarrow 0} B_q = 0 \quad (3.2.8)$$

This results to a discrete model that has no counterpart to the continuous model and the system could be unstable as the z-plane poles are all approximately unity.

On the other hand, by introducing the  $\delta$  operator which is a divided difference operator, it ameliorates the previous problem. This Delta form is a shifted and scaled version of the z-domain and is derived by transforming the discrete state space model according to the following equation:

$$q = \delta\Delta T + 1 \quad (3.2.9)$$

The  $\delta$ -operator discrete state space model is the following:

$$\delta x(k) = A_\delta x(k) + B_\delta u(k) + w_\delta(k) \quad (3.2.10)$$

$$y(k) = C_\delta x(k) + D_\delta u(k) + e_\delta(k) \quad (3.2.11)$$

where in this occasion, the state space matrices  $A_\delta$ ,  $B_\delta$ ,  $C_\delta$  and  $D_\delta$  have the following form:

$$\begin{aligned} A_\delta &= \frac{A_q - I}{\Delta} \\ B_\delta &= \frac{B_q}{\Delta} \\ C_\delta &= C_q = C \\ D_\delta &= D_q = D \\ w_\delta &= \frac{w_q}{\Delta} \\ e_\delta &= e_q \end{aligned}$$

In this occasion when the sampling time tends to zero, the delta-based discrete state space model converges to the continuous time system, meaning that:

$$\lim_{\Delta \rightarrow 0} A_\delta = A \quad (3.2.12)$$

$$\lim_{\Delta \rightarrow 0} B_\delta = B \quad (3.2.13)$$

Based on the previous, the subspace identification algorithm (SID) is modified in that case. Following the SID matrix derivation in Appendix C, the extended observability matrix which is used to derive the A and C matrices is transformed as follows:

$$\Gamma_i = \begin{bmatrix} C_\delta \\ C_\delta A_\delta \\ C_\delta A_\delta^2 \\ \vdots \\ C_\delta A_\delta^{i-1} \end{bmatrix} \in R^{li \times n} \quad (3.2.14)$$

Moreover, the Hankel formed input-output matrices are reconstructed in the following form.

The past input Hankel matrix is given by:

$$U_p = \begin{bmatrix} u(0) & u(1) & \dots & u(j-1) \\ \delta u(1) & \delta u(2) & \dots & \delta u(j) \\ \vdots & \vdots & \vdots & \vdots \\ \delta^{k-1} u(i-1) & \delta^{k-1} u(i) & \dots & \delta^{k-1} u(i+j-2) \end{bmatrix} \quad (3.2.15)$$

The future input Hankel matrix is given by:

$$U_f = \begin{bmatrix} \delta^k u(i) & \delta^k u(i+1) & \dots & \delta^k u(i+j-1) \\ \delta^{k+1} u(i+1) & \delta^{k+1} u(i+2) & \dots & \delta^{k+1} u(i+j) \\ \vdots & \vdots & \ddots & \vdots \\ \delta^{2k-1} u(2i-1) & \delta^{2k-1} u(2i) & \dots & \delta^{2k-1} u(2i+j-2) \end{bmatrix} \quad (3.2.16)$$

The past output Hankel matrix is given by:

$$Y_p = \begin{bmatrix} y(0) & y(1) & \dots & y(j-1) \\ \delta y(1) & \delta y(2) & \dots & \delta y(j) \\ \vdots & \vdots & \ddots & \vdots \\ \delta^{k-1} y(i-1) & \delta^{k-1} y(i) & \dots & \delta^{k-1} y(i+j-2) \end{bmatrix} \quad (3.2.17)$$

The future output Hankel matrix is given by:

$$Y_f = \begin{bmatrix} \delta^k y(i) & \delta^k y(i+1) & \dots & \delta^k y(i+j-1) \\ \delta^{k+1} y(i+1) & \delta^{k+1} y(i+2) & \dots & \delta^{k+1} y(i+j) \\ \vdots & \vdots & \ddots & \vdots \\ \delta^{2i-1} y(2i) & \delta^{2i} y(2i+1) & \dots & \delta^{2k-1} y(2i+j-2) \end{bmatrix} \quad (3.2.18)$$

where  $i$  is the number of block rows and  $j$  is the number of sampled data. The extended controllability matrix is defined as follows:

$$\Delta_i = [A_\delta^{i-1} B_\delta \quad A_\delta^{i-2} B_\delta \quad \dots \quad A_\delta B_\delta \quad B_\delta] \quad (3.2.19)$$

Accordingly, the lower triangular Toeplitz matrix is defined as follows:

$$H_i^d = \begin{bmatrix} D_\delta & 0 & 0 & \dots & 0 \\ C_\delta B_\delta & D_\delta & 0 & \dots & 0 \\ C_\delta A_\delta B_\delta & C_\delta B_\delta & D_\delta & \dots & 0 \\ \dots & \dots & \dots & \ddots & \dots \\ C_\delta A_\delta^{i-2} B_\delta & C_\delta A_\delta^{i-3} B_\delta & C_\delta A_\delta^{i-4} B_\delta & \dots & D_\delta \end{bmatrix} \quad (3.2.20)$$

The process of estimating the state space matrices  $A_\delta, B_\delta, C_\delta, D_\delta$  is the same as in the standard discrete time subspace system identification based on the MOESP or N4SID approaches.

In the following section, the effectiveness, the performance and the necessity of the  $\delta$ -operator subspace identification algorithm, will be evaluated and compared to the MOESP and N4SID algorithms described in the previous section for fast sampling data.

### 3.3 SID of Stiff Systems

For the purpose of this section, a stiff system that represents a quarter car model is presented. The active car suspension will be simulated and its state space model will be estimated using the three subspace system identification algorithms.

The vehicle suspension model is represented as a linear system. It consists of a single sprung mass (vehicle body- $m_1$ ) connected to two unsprung masses (front and rear wheel assembly masses) at each corner.

The unsprung mass represents the wishbone arms, wheel hub, knuckle, tire, rim, brake caliper, disc plate etc. it means that the components that are suspended (swinging, hanging ) from the frame are mentioned as unsprung mass. Each mass of the individual unsprung components are represented as a whole and it is named as unsprung mass. The same is applied for the sprung mass but the sprung mass represents the components which are mounted on the frame.

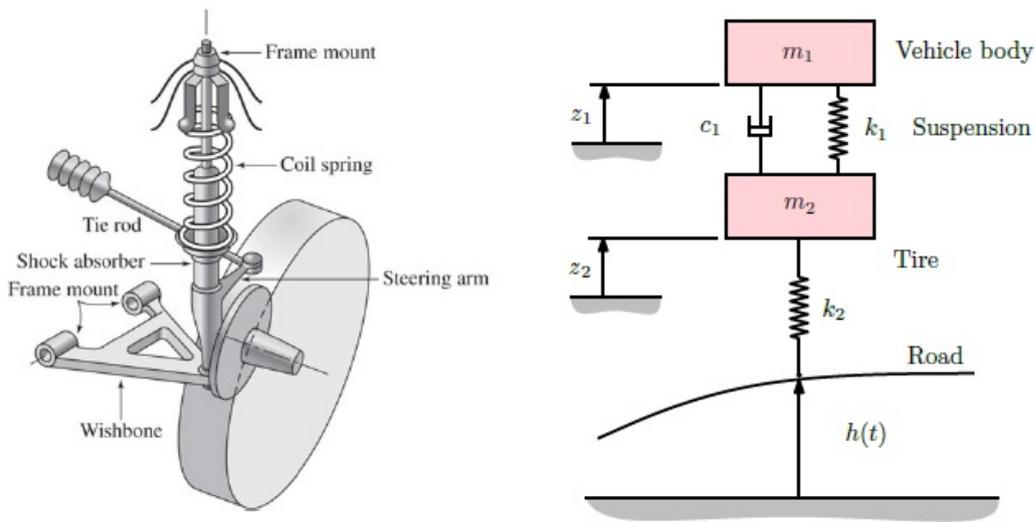


Figure 3.3.1: Vehicle suspension & Physics Representation

The sprung mass is free to heave and pitch, while the unsprung masses ( $m_2$ ) are free to bounce vertically with respect to the sprung mass. The suspensions between the sprung mass and unsprung masses are modelled as linear viscous dampers and spring elements, while the tyres are modelled as simple linear springs without damping components in half car model while in quarter car model damping is also included for tyre.

Given two masses  $m_1 \in \mathbb{R}$  and  $m_2 \in \mathbb{R}$ - representing the sprung and

the unsprung mass respectively- that are attached via a spring with spring constant  $k_1 \in \mathbb{R}$  and a second spring with spring constant  $k_2 \in \mathbb{R}$  attached mass  $m_2$  to a forcing plane with displacement from equilibrium  $f(t) \in \mathbb{R}$ . Damping  $d_1, d_2 \in \mathbb{R}$  is introduced and is the same for both springs.

The states  $x_1(t) \in \mathbb{R}$  is defined as the displacement of  $m_1$  from equilibrium and similarly  $x_2(t) \in \mathbb{R}$  as the displacement of  $m_2$  from equilibrium. The following equations describe the dynamics of the system:

$$\ddot{x}_1(t) + d_1\dot{x}_1(t) - \varepsilon d_1\dot{x}_2(t) + \bar{k}_1(x_1(t) - x_2(t)) = 0, \quad (3.3.1)$$

$$\varepsilon\ddot{x}_2(t) - d_1\dot{x}_1(t) + \varepsilon(d_1 + d_2)\dot{x}_2(t) + \bar{k}_1(x_2(t) - x_1(t)) + \bar{k}_2(x_2(t) - f(t)) = 0, \quad (3.3.2)$$

where  $\ddot{x} := d^2x/dt^2$ ,  $\varepsilon := m_2/m_1$ ,  $\bar{k}_1 := k_1/m_1$  and  $\bar{k}_2 := k_2/m_1$ .

The stiff parameter  $\varepsilon$  dictates the multi-time-scale behaviour of the system. As  $\varepsilon \rightarrow 0$  the disparity between the frequencies of signals  $x_1(t)$  and  $x_2(t)$  increases, with  $x_1(t)$  being the ‘‘slow’’ state (i.e. low frequency) and  $x_2(t)$  being the ‘‘fast’’ state (i.e. high frequency).

The continuous system can be represented as a state-space system such as:

$$x(t) = \begin{bmatrix} \dot{x}_1(t) & \varepsilon\dot{x}_2(t) & x_1(t) & x_2(t) \end{bmatrix}^\top,$$

$$A = \begin{pmatrix} -d_1 & d_1 & -\bar{k}_1 & \bar{k}_1 \\ d_1 & -(d_1 + d_2) & \bar{k}_1 & -(\bar{k}_1 + \bar{k}_2) \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

$$B = \begin{pmatrix} 0 \\ \bar{k}_2 \\ 0 \\ 0 \end{pmatrix}$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$D = (0 \ 0 \ 0 \ 0)$$

The discrete model is simulated for different samples rates and for random inputs with added white noise and the system outputs are computed. These

inputs and outputs are fed into the subspace methods and a new state space representation will be derived in every case. The new model will be compared to the initial state space model and their performance and robustness will be evaluated.

The subspace system identification algorithms are designed and simulated using the platform of MATLAB. The simulations were implemented and results are obtained for various sampling frequencies beginning from  $F_s = 2^5\text{Hz}$  up to  $F_s = 2^{20}\text{Hz}$ . Moreover, various stiff ratio  $\varepsilon$  values were tested, starting from 0.1 up to  $10^{-6}$ . The algorithm precision under extreme conditions of stiffness and fast sampling where ill-conditioned phenomena emerge, will be evaluated. As a next step, the outputs of the initial state space system will be corrupted with white noise and the performance of the SID will be presented too.

### 3.3.1 Experimental Results & Simulations

In this section, various simulations were implemented for the following occasions:

- Fixed sampling rate and varying  $\varepsilon$
- Varying sampling rate and fixed  $\varepsilon$
- Noise free measurements
- Noisy measurements

The behaviour and performance of subspace identification algorithms is investigated for various stiffness ratios combined with different sampling rates.

### 3.3.2 Noise Free Measurements

The first simulations are exploited for different sampling rates and stiffness ratio. The aim is to reveal the behaviour and the possibility of ill conditioned cases of the algorithm matrices that may emerge as the sample rate increases. Theoretically, fast sampling rates could lead in singular matrices making the accurate state space model estimation impossible.

The tables 3.1, 3.2, 3.3 & 3.4 illustrate the norm of the difference between the given discrete model outputs and the estimated one based on the different SID algorithms for occasions where no added measurement noise on the output Hankel matrices is introduced.

More specifically, the table 3.1 demonstrates the norm of the difference between the given model and the 3 different subspace models where  $sys_{SID}$  is the subspace MOESP model,  $sys_{\delta SID}$  is the subspace methods based on  $\delta$ -operator and  $sys_{N4SID}$  is based on N4SID, for different sampling rates.

Sample Rate	$ sys_d - sys_{SID} $	$ sys_d - sys_{\delta SID} $	$ sys_d - sys_{N4SID} $
$2^6$	$6.36 \times 10^{-12}$	$3.39 \times 10^{-12}$	$3.87 \times 10^{-13}$
$2^8$	$8.88 \times 10^{-10}$	$2.8 \times 10^{-10}$	$9.15 \times 10^{-12}$
$2^{10}$	$5.86 \times 10^{-9}$	$3.5 \times 10^{-8}$	$1.14 \times 10^{-11}$
$2^{11}$	$2.41 \times 10^{-9}$	$2.04 \times 10^{-7}$	$2.74 \times 10^{-11}$
$2^{12}$	$9.33 \times 10^{-8}$	$7.78 \times 10^{-11}$	$1.75 \times 10^{-8}$
$2^{14}$	$2.62 \times 10^{-6}$	$2.34 \times 10^{-5}$	$2.48 \times 10^{-9}$
$2^{16}$	$3.04 \times 10^{-4}$	0.03	$5.08 \times 10^{-8}$
$2^{18}$	$6.46 \times 10^{-4}$	0.0338	$2.82 \times 10^{-5}$

Table 3.1: Difference of given discrete state space model & subspace models for fixed stiffness ratio  $\epsilon=0.1$

Based on the results above, it is shown that all SID algorithms manage to identify an accurate model representation for all fast sampling data occasions. The sample rate is increased up to  $2^{19}$  for fixed  $\epsilon = 0.1$  and even in these occasions, the algorithms are robust, estimating accurate models.

As a second example, the performance of the SID algorithms is evaluated for fixed sample rate and varying  $\varepsilon$  values. The aim is to evaluate the accuracy of these algorithms when the stiffness ratio is increased and the system is governed by multiscale characteristics.

Table 3.2 presents the results of a stiff system with a high sampling frequency. Simulations are provided for varying  $\varepsilon$  and sampling frequency  $2^{18}$  Hz.

According to tables 3.2, 3.3 and 3.4, for the extreme occasions where the sampling rate is increased up to  $2^{19}$  Hz and the stiffness ratio is high, the Delta subspace identification method deals with difficulties at identifying a representative and additionally stable dynamic model in contrast with the MOESP and N4SID that manage to deal with the occasions. This indicates that the fundamental SID algorithms are efficient under extreme occasions where the model presents multiscale characteristics.

As the sampling rate decreases, the subspace identification methods are becoming more efficient and manage to identify more accurate models. The

$\epsilon$	$ sys_d - sys_{SID} $	$ sys_d - sys_{\delta SID} $	$ sys_d - sys_{N4SID} $
$10^{-3}$	$1.63 \times 10^{-4}$	0.0234	0.9130
$10^{-4}$	$4.41 \times 10^{-4}$	0.0071	0.0051
$10^{-5}$	$8.34 \times 10^{-4}$	0.0137	0.0835
$10^{-6}$	0.0668	0.0243	0.1164

Table 3.2: Difference of given discrete state space model & subspace models for fixed sampling rate  $F_s=2^{18}$  Hz

$\epsilon$	$ sys_d - sys_{SID} $	$ sys_d - sys_{\delta SID} $	$ sys_d - sys_{N4SID} $
$10^{-3}$	$1.6742 \times 10^{-4}$	0.0025	$4.9804 \times 10^{-6}$
$10^{-4}$	$1.2346 \times 10^{-4}$	0.0015	$4.2390 \times 10^{-8}$
$10^{-5}$	$1.5866 \times 10^{-4}$	0.0017	$4.0263 \times 10^{-8}$

Table 3.3: Difference of given discrete state space model & subspace models for fixed sampling rate  $F_s=2^{16}$  Hz

$\epsilon$	$ sys_d - sys_{SID} $	$ sys_d - sys_{\delta SID} $	$ sys_d - sys_{N4SID} $
$10^{-3}$	$1.6742 \times 10^{-4}$	$6.23 \times 10^{-5}$	$1.58 \times 10^{-9}$
$10^{-4}$	$1.41 \times 10^{-5}$	$3.65 \times 10^{-4}$	$8.69 \times 10^{-10}$
$10^{-5}$	$8.51 \times 10^{-8}$	$1.5866 \times 10^{-4}$	$4.049 \times 10^{-10}$

Table 3.4: Difference of given discrete state space model & subspace models for fixed sampling rate  $F_s=2^{14}$  Hz

robustness of these methods can be seen in extreme occasions where the sampling frequency is as high as  $F = 2^{14}$  Hz and the stiffness ratio as low as  $\epsilon = 10^{-5}$ , estimating accurate models and manage to reveal the multiscale nature of initial dynamic model.

Following the tables above, it is of interest to present the system response of the three different identification methods. The plots below demonstrate the system outputs for every different occasion and are compared to the initial model. Additionally, the eigendecomposition of the derived models

are presented and compared with initial discrete model.

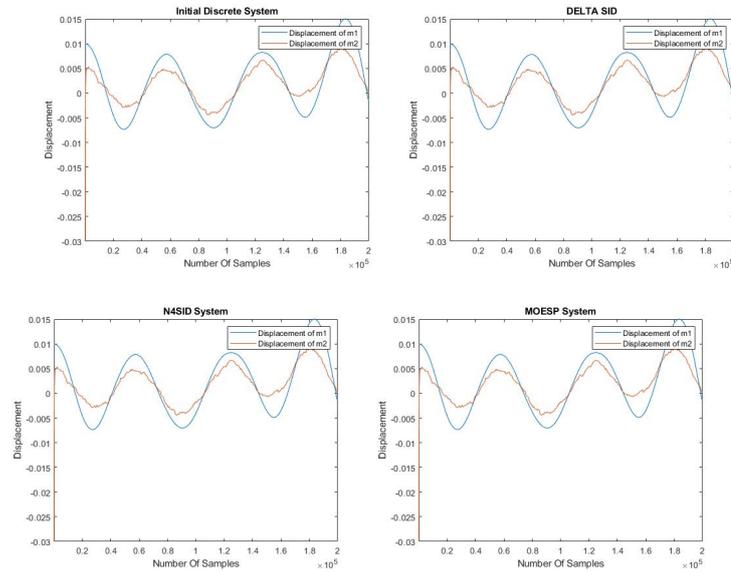


Figure 3.3.2: System response of the derived state space model using subspace system identification methods for  $\mathbf{F_s} = 2^{16} \mathbf{Hz}$  and  $\epsilon = 10^{-5}$

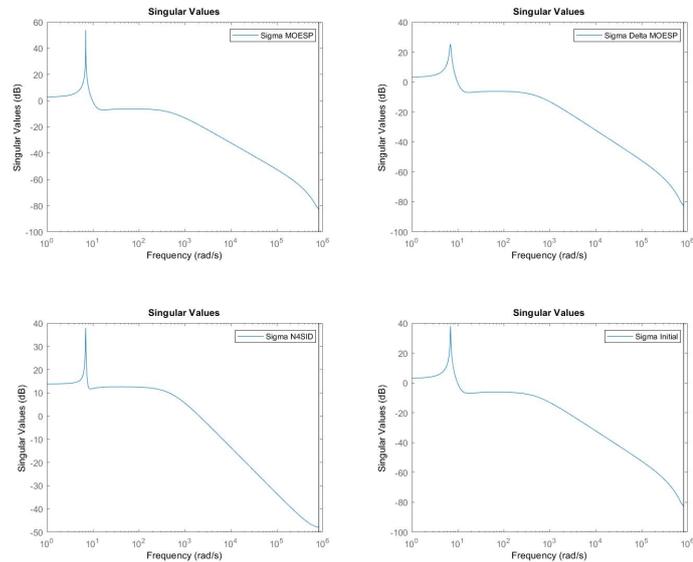


Figure 3.3.3: Poles of the derived state space model using subspace system identification methods for  $\mathbf{F_s} = 2^{16} \mathbf{Hz}$  and  $\epsilon = 10^{-5}$

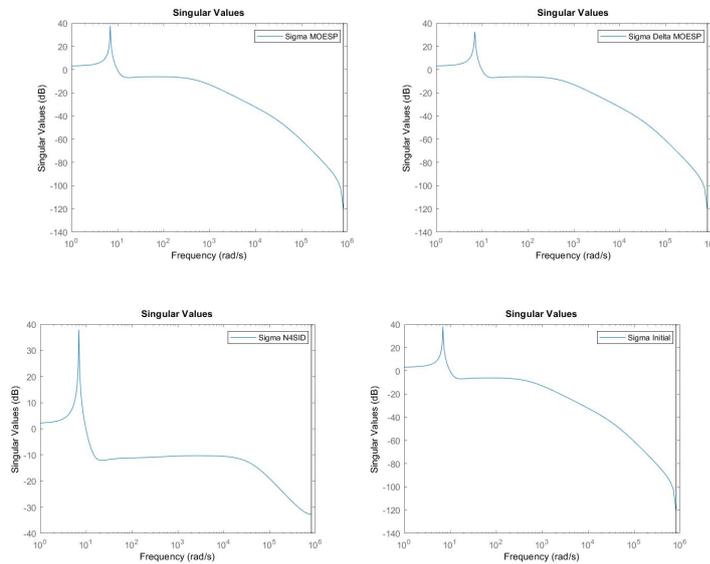


Figure 3.3.4: Poles of the derived state space model using subspace system identification methods for  $\mathbf{F}s = 2^{16} \text{ Hz}$  and  $\epsilon = 10^{-5}$

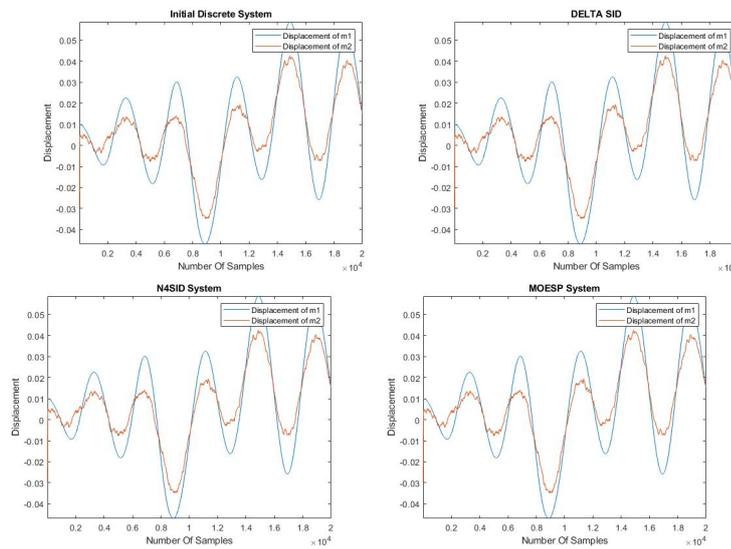


Figure 3.3.5: System response of the derived state space model using subspace system identification methods for  $\mathbf{F}s = 2^{12} \text{ Hz}$  and  $\epsilon = 10^{-4}$

The simulations above justify the assumption that the SID algorithms manage to identify the system dynamics successfully. Moreover, according to the simulations, the  $\delta$ -SID did not manage to improve the algorithm performance as was initially surmised. The following images demonstrate a comparison between the output measurements of the initial continuous model and the ones from the derived model from the  $\delta$ -SID method.

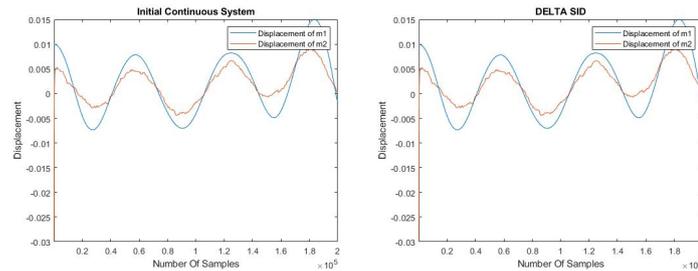


Figure 3.3.6: System response comparison of the derived state space model using  $\delta$  subspace system identification methods for  $\mathbf{F}s = 2^{16} \mathbf{Hz}$  and  $\varepsilon = 10^{-5}$  & initial continuous system

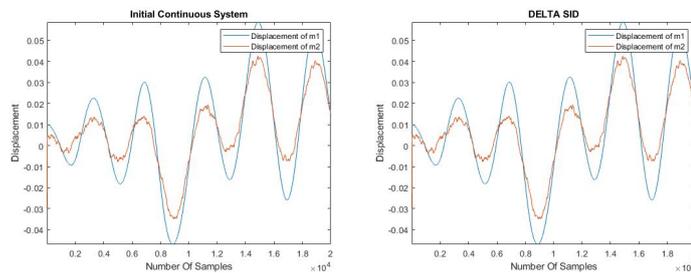


Figure 3.3.7: System response comparison of the derived state space model using  $\delta$  subspace system identification methods for  $\mathbf{F}s = 2^{12} \mathbf{Hz}$  and  $\varepsilon = 10^{-4}$  & initial continuous system

### 3.3.3 Measurements With Added Noise

In the previous section, the identification algorithms presented very high accuracy, even for exceptionally high sampling rates and stiffness ratios. In this section, the behaviour and performance of the three identification algorithms will be studied for data with additive white noise, for varying Signal-to-Noise Ratios (SNRs) and number of samples.

It is known that the noise is square root proportional to the noise spectral width. And the noise spectral width is related to the sampling rate by the time-bandwidth relation. As a result, as the sampling rate increases, the noise spectrum becomes wider, resulting to an increased measurement noise. Hence, as the SNR decreases, the subspace algorithms have to deal with the effects of increased noise.

The accuracy of the algorithms is affected disproportionately by factors such as the amount of samples, the noise margin and the sampling rate. Different simulations took place where the SID algorithms performance is investigated in order to validate their performance. The simulations are shown below for various combinations of the parameters described previously.

<b>Samples, SNR</b>	$ sys_d - sys_{SID} $	$ sys_d - sys_{DELSID} $	$ sys_d - sys_{N4SID} $
500, 40dB	Inf	Inf	0.2204
4000, 40dB	Inf	Inf	0.0161
10000, 40dB	Inf	2.5534	0.0043
20000, 40dB	2.2413	2.2526	0.0073
40000, 40dB	2.2461	2.2447	0.0108
60000, 40dB	2.2468	2.2507	0.1995

Table 3.5: Difference of given discrete state space model and subspace models for sampling rate  $F_s = 2^6$  Hz and  $\varepsilon = 0.1$

As illustrated in the previous tables, Table 3.5 demonstrates the norm of the difference between the given system response and the response of the three different models derived by the SID for different SNRs and number of input data. It is important to note, that for high number of samples ( $\geq 80.000$ ) and high sampling frequencies ( $\geq 2^{10}$ ), all identification method fail to identify a representative model of the input data.

The table 3.8, presents a conclusive table of the 3 different methods accuracy for four different amount of samples and three different SNRs, for higher sampling frequency  $F = 2^{10}$  Hz, confirming the difficulties of the subspace methods to estimate representative models compared to the given one. Especially in simulations, where the SNR is lower than 60dB, the Delta subspace identification method estimates an unstable dynamic model and for lower than 40dB, all the methods estimate unstable dynamic models too.

Following the results presented above, it is shown that for low SNRs and as the sampling rate and the number of the available samples increases, the identification methods deal with difficulties in identifying representative

<b>Samples, SNR</b>	$ sys_d - sys_{SID} $	$ sys_d - sys_{DELSID} $	$ sys_d - sys_{N4SID} $
500, 60dB	Inf	Inf	0.0186
4000, 60dB	2.1785	2.1959	0.0019
10000, 60dB	Inf	2.1760	0.0036
20000, 60dB	0.2831	1.9876	0.0163
40000, 60dB	2.0869	2.0831	0.0011
60000, 60dB	2.0875	Inf	0.0194

Table 3.6: Difference of given discrete state space model & subspace models for sampling rate  $F_s = 2^6$  and  $\varepsilon = 0.1$

<b>Samples, SNR</b>	$ sys_d - sys_{SID} $	$ sys_d - sys_{DELSID} $	$ sys_d - sys_{N4SID} $
500, 60dB	0.0360	0.3441	0.0106
4000, 60dB	0.1047	0.9493	0.0010
10000, 60dB	0.0469	0.0776	0.0031
20000, 60dB	0.0246	0.0707	0.0010
40000, 60dB	0.0089	0.1135	0.0001
60000, 60dB	0.0171	0.1374	0.0012

Table 3.7: Difference of given discrete state space model & subspace models for sampling rate  $F_s = 2^5$  Hz and  $\varepsilon = 0.1$

dynamic models to the initial one. In all occasions though, as the SNR increases, the algorithms manage to deal with the additive noise indicating that the use of any possible data pre-filtering method that improves the SNR, can lead to accurate model representations.

For sampling frequencies greater than  $2^{13}$  Hz, the algorithms performance is deteriorated, hence the SNR had to be increased significantly ( $SNR \geq 70dB$ ) in order to obtain satisfactory results.

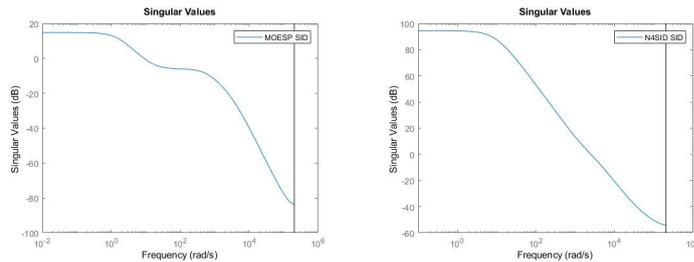
The following plots demonstrate the Bode plots of the singular values response of the derived dynamic systems in a transfer function form. These plots present the damped natural frequencies within the accuracy of the frequency-resolution and corresponding non-scaled mode shapes and solutions.

The plots will confirm the problem of the derived methods at identifying

<b>Samples, SNR</b>	$ sys_d - sys_{SID} $	$ sys_d - sys_{DELSID} $	$ sys_d - sys_{NASID} $
5000,20dB	Inf	0.6517	0.571
10000,20dB	Inf	0.9196	Inf
15000,20dB	Inf	Inf	Inf
20000,20dB	3.436	Inf	0.5174
5000,40dB	127.65	0.6151	117.9995
10000,40dB	0.6067	Inf	0.3353
15000,40dB	Inf	Inf	Inf
20000,40dB	20.81	0.61	23.81
5000,60dB	15.11	Inf	0.096
10000,60dB	0.7198	0.633	0.024
15000,60dB	21.52	Inf	0.095
20000,60dB	0.1551	Inf	0.1551

Table 3.8: Difference of given discrete state space model & subspace models for sampling rate  $F_s = 2^{10}$  Hz and  $\varepsilon = 0.1$

and estimating the behaviour and system response when noise is added in different occasions, such as increased samples or increased sampling ratio and stiffness ratio.



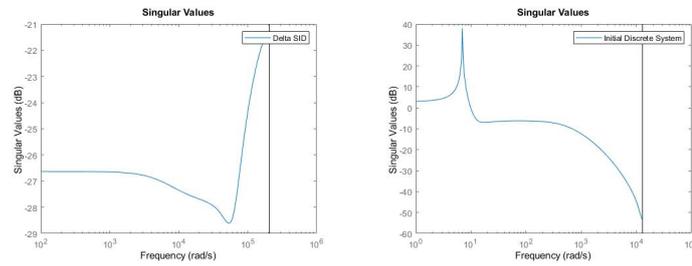


Figure 3.3.8: Singular values of the derived state space models for :  $F_s = 2^{16}$  Hz,  $N=200.000$  samples and  $SNR = 70$ db

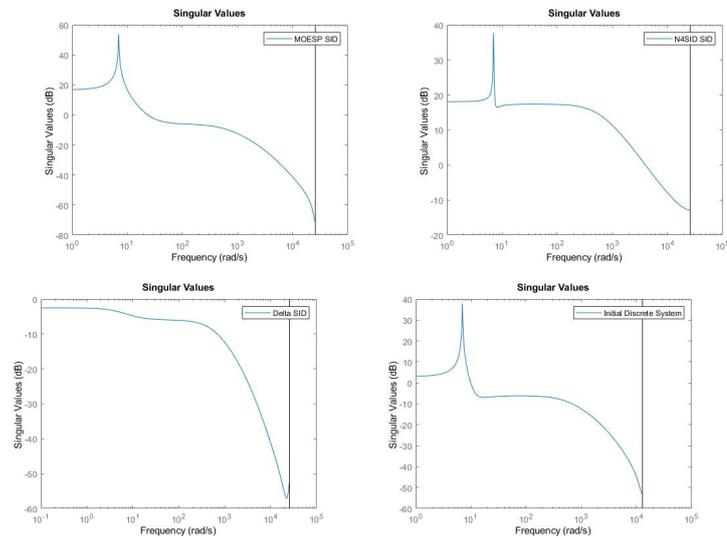
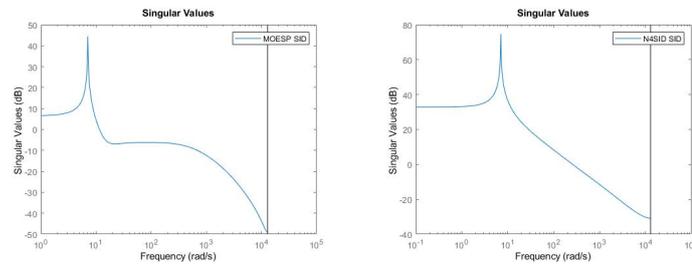


Figure 3.3.9: Singular values of the derived state space model using subspace system identification methods for  $F_s = 2^{13}$  Hz,  $N=20000$  Samples and  $SNR = 70$ db



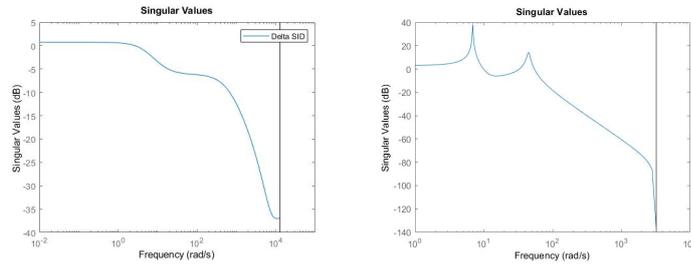


Figure 3.3.10: Singular values of the derived state space model using subspace system identification methods for  $F_s = 2^{12}$ ,  $\varepsilon = 10^{-4}$  and SNR = 70dB

The following images demonstrate a comparison between the output measurements of the initial continuous model and the ones from the derived model from the  $\delta$ -SID method for different SNR.

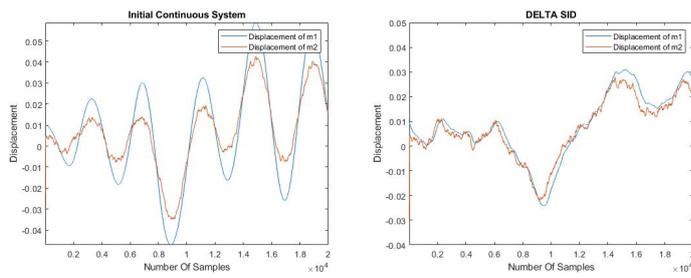


Figure 3.3.11: System response comparison of the derived state space model using  $\delta$  subspace system identification methods for  $\mathbf{F}s = 2^{12}$  Hz and  $\varepsilon = 10^{-4}$  & initial continuous system

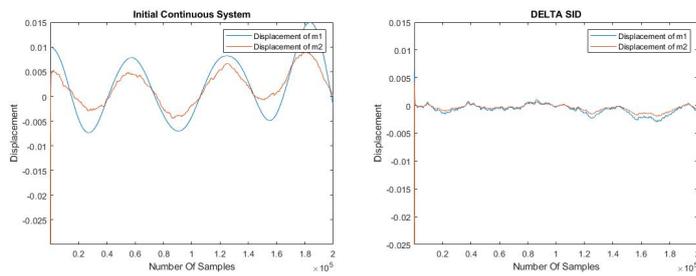


Figure 3.3.12: System response comparison of the derived state space model using  $\delta$  subspace system identification methods for  $\mathbf{F}s = 2^{16}$  Hz and  $\varepsilon = 10^{-5}$  & initial continuous system

### 3.4 Summary

In conclusion, in this chapter the SID method was applied over a system with multiscale characteristics and high sampling frequency. The aim was to identify the threshold where the SID could potentially struggle at revealing all system structures for both noise-free and noisy input-output data and enhance its performance by introducing the  $\delta$ -operator SID.

By simulating both methods for the noise free measurements, all identification algorithms estimate accurate dynamic models which are identical to the initial discrete dynamic model as shown from their frequency responses and singular value plots. This indicates that no SID alternative with single-scale or multiscale structure needs to be obtained.

In the occasions, where white noise is added to the measurements, it is shown that as the sampling frequency increases and as the SNR reduces, the more inaccurate the algorithms are in terms of estimating a representative model to the initial one which is justified by the results. As the number of samples corrupted with noise increases, the more difficult becomes for the algorithms to obtain accurate estimations.

The subspace algorithms deal with these difficulties in most of occasions when the SNR = 20dB and various number of samples. In occasions where the SNR is higher such as SNR = 60dB or even higher, all algorithms manage to estimate accurate models and especially N4SID is estimating state space models that are almost identical to the initial discrete one. On the other hand, as the number of samples increases, the  $\delta$  subspace identification algorithm deals with problems due to noise. Consequently, for the occasions where the SNR is low, the measurements could be filtered using methods such as Kalman Smoothing, accomplishing higher SNRs where the identification methods are efficient.

Additionally, all algorithms are immune to added noise to the inputs, confirming the theory presented in chapter 3 where it was proven that through projection of data, the input noise is filtered in every occasion.

As a conclusion, the proposed  $\delta$ -operator state space model does not offer any higher accuracy in the temporal model estimation when the sampling frequency is very high, indicating the robustness of the conventional subspace identification method. The main use of the  $\delta$ -operator could be that of linking the continuous counterpart of the derived discrete model when the sampling time tends to zero.

## Chapter 4

# Multiscale Spatio-temporal Signal Modelling

The development of reduced-order models (ROMs) for high dimensional complex systems, concludes a wide range of engineering applications including mechanism analysis, dynamic modelling, control law design and optimization. Reduced-order models based on input-output data, provide satisfactory results as described in previous chapters for modelling dynamic systems for forecast and control purposes.

Gaining a deep and accurate understanding of the behaviour over time of linear and non-linear systems requires a form of reduced order models or mode decomposition techniques due to the inefficient and high computational cost when dealing with high dimensional data.

Proper Orthogonal Decomposition (POD), Dynamic Mode Decomposition (DMD) and their expansions are fundamental and powerful methods in analysing linear and non-linear spatio-temporal systems. They are designed to extract the system dominant coherent structures, without requiring the explicit prior knowledge of the governing equations. These methods were initially introduced in Computational Fluid Dynamics (CFD) for gaining a better understanding of complex fluid dynamics overall behaviour over time. Fast though, they gained much attention in many more fields due to their simple structure and computational power.

These methods are referred as modal decomposition methods and integrate mathematical techniques that extract energetically and dynamically important features of the introduced process. The spatial extracted features are called (spatial) modes and they are followed by characteristic values that correspond to the energy concentration levels or growth rates and frequencies of each mode [71].

In this chapter, an extended analysis of these modal decomposition meth-

ods will be presented. The POD, DMD algorithm and their expansions will be analytically presented and applied over different examples. Up to date, there is no analysis in literature revealing the advantages and drawbacks of the different alternatives. Hence, a detailed discussion over various examples with periodical behaviour and multi-scale characteristics will be given. This analysis will reveal the necessity of alternative approaches considering systems with multiscale dynamics and mixed frequency effects.

## 4.1 Proper Orthogonal Decomposition

As introduced in Chapter 2, the Proper Orthogonal Decomposition (POD) is a post-processing technique. It obtains a given set of data and extracts basis functions, that contain as much information as possible. The POD method creates an orthogonal basis for representing a given set of data in a certain least-squares optimal sense. In fact, it offers ways to find optimal lower-dimensional approximations for the given data set and that justifies the term "proper" in the orthogonal decomposition process [79].

These approximations are also referred by the term "energy" [79, 80]. The energy of a dynamical system through POD is typically expressed as a Singular Value Decomposition. The SVD extracts spatio-temporal coherent structures arranged in descending order according to their energy contribution to the dynamic system. These coherent structures are called modes. The SVD is a decomposition technique for rectangular matrices and POD can be seen as a decomposition formalism where SVD has the advantage of being robust against round error [80].

POD is a multi-variate statistical method, known for handling high dimensional data sets. The POD modes and the corresponding eigenvalues are thus characterized by the eigensolutions of the sample covariance matrix  $\mathbf{C}$ .

$$\mathbf{C} = \frac{1}{n} \mathbf{X} \mathbf{X}^T$$

Given that the SVD computes the matrices:

$$\begin{aligned} \mathbf{X} \mathbf{X}^T &= \mathbf{U} \mathbf{S}^2 \mathbf{U}^T \\ \mathbf{X}^T \mathbf{X} &= \mathbf{V} \mathbf{S}^2 \mathbf{V}^T \end{aligned} \tag{4.1.1}$$

It is known that the singular values of  $\mathbf{X}$  are equal to the square roots of the eigenvalues of  $\mathbf{X} \mathbf{X}^T$  or  $\mathbf{X}^T \mathbf{X}$ . The left and right singular vectors of  $\mathbf{X}$  are the eigenvectors of  $\mathbf{X} \mathbf{X}^T$  and  $\mathbf{X}^T \mathbf{X}$  respectively. The POD modes, which

are determined as the eigenvectors of the covariance matrix  $\mathbf{C}$ , are equal to the left singular vectors of  $\mathbf{X}$ . The proper orthogonal values, which designate the eigenvalues of matrix  $\mathbf{C}$ , are the square of the singular values divided by the number of samples.

Hence, POD is simply given by the computation of the reduced SVD of the  $\mathbf{X}$  matrix :

$$\mathbf{X} = U_r S_r V_r^* \quad (4.1.2)$$

where  $U_r$  has a size of  $mn \times r$  representing the dominant coherent structures or else the POD Modes

$S_r$  has a size of  $r \times r$

$V_r$  has a size of  $mn \times r$ , with  $r$  being the number of modes selected.

The POD modes are organized from most important to least important in terms of capturing the variance of  $\mathbf{X}$ . The order of the modes indicates which modes are dominating in the data structure in terms of the energy concentration. The number of selected modes differs and depends on the application demands based on criteria such as accuracy and amount of information needed.

## 4.2 Dynamic Mode Decomposition (DMD)

The Dynamic mode decomposition (DMD) is an expansion of POD which was initially introduced in Fluid Mechanics, forming a powerful tool for analysing linear and non-linear dynamic models described by high-dimensional data models [70]. As presented in Chapter 2, the DMD algorithm is arranging the data based on their dynamical importance and not simply by the energy content, thus dynamics that contain small energy can be captured in contrast to the POD methods.

The DMD intends to represent the linear or non-linear dynamics with a simple linear dynamic model :

$$\mathbf{X}' = A\mathbf{X} \quad (4.2.1)$$

The ultimate goal is to estimate the best linear fit operator  $\tilde{A}$  that can approximate the dynamics of the high dimensional data. The DMD modes and eigenvalues are computed in order to represent the system with the dominant eigenvalues and eigenvectors.

The DMD algorithm is summarized in the following steps.

- The first step of the DMD is the data collection and their resizing into high-dimensional column matrices. Let  $x_i$  be the  $i^{th}$  video frame or snapshot, of size  $m \times n$ , arranged in a column vector of size  $mn \times 1$ . The data are arranged into two matrices of size  $mn \times (n - 1)$ .

$$\mathbf{X} = [\bar{\mathbf{x}}_1, \bar{\mathbf{x}}_2, \bar{\mathbf{x}}_3, \dots, \bar{\mathbf{x}}_{N-1}] = \begin{pmatrix} x_1^1 & x_2^1 & \dots & x_{N-1}^1 \\ \vdots & \vdots & \vdots & \vdots \\ x_1^{mn} & x_2^{mn} & \dots & x_{N-1}^{mn} \end{pmatrix} \quad (4.2.2)$$

and its time shifted counterpart :

$$\mathbf{X}' = [\bar{\mathbf{x}}_2, \bar{\mathbf{x}}_3, \bar{\mathbf{x}}_4, \dots, \bar{\mathbf{x}}_N] = \begin{pmatrix} x_2^1 & x_3^1 & \dots & x_N^1 \\ \vdots & \vdots & \vdots & \vdots \\ x_2^{mn} & x_3^{mn} & \dots & x_N^{mn} \end{pmatrix} \quad (4.2.3)$$

where each snapshot  $\bar{\mathbf{x}}_N$  has a size of  $mn \times 1$  and  $X'$  are one step time advanced data. Aim of the DMD is to compute a matrix  $A$  that satisfies the equation  $\mathbf{X}' = A\mathbf{X}$ . Thus

$$A = \mathbf{X}'\mathbf{X}^\dagger \quad (4.2.4)$$

The  $\mathbf{X}^\dagger$  is a least square regression algorithm, called Moore-Penrose algorithm.

- There is a dimensionally problem in the equation 4.2.4. The two matrices are high dimensional, hence their outer product will be a massive matrix causing computational problems. As a result, the second step includes the computation of the reduced SVD of the  $X$  matrix and the extraction of the dominant coherent structures.

$$\mathbf{X} = U_r \Sigma_r V_r^* \quad (4.2.5)$$

where  $U_r$  has a size of  $mn \times r$  representing the dominant coherent structures or else the POD Modes

$\Sigma_r$  has a size of  $r \times r$  and

$V_r$  has a size of  $mn \times r$ , with  $r$  being the number of modes selected.

Hence using equation 4.2.4 and 4.2.5:

$$\begin{aligned} \mathbf{X}' &= AU_r \Sigma_r V_r^* \\ A &= \mathbf{X}' V_r \Sigma_r^{-1} U_r^* \end{aligned} \quad (4.2.6)$$

There is no specific selection criteria for the amount of the dominant modes. The selection is mainly based on the energy concentration ratio of the first  $r$  modes and the total system energy.

- The third step evaluates the projection of the high dimensional matrix  $A$  to the  $U_r$  matrix that includes the dominant modes. The objective of DMD is not to work directly with the matrix  $A$ , due to the size of  $A$  which is massive due to the high dimensional data  $\mathbf{X}'$ . Hence, DMD will be work with a similarity transform of  $A$ , called  $\tilde{A}$  with a nice set of variables,  $U_r$ .

Thus, a truncated matrix  $\tilde{A}$  is derived and described as:

$$\tilde{A} = U_r^* A U_r \quad (4.2.7)$$

The combination of equation 4.2.6 and 4.2.7 gives :

$$\begin{aligned} \tilde{A} &= U_r^* [\mathbf{X}' V_r \Sigma_r^{-1} U_r^*] U_r \\ \tilde{A} &= U_r^* [\mathbf{X}' V_r \Sigma_r^{-1}] \end{aligned} \quad (4.2.8)$$

where  $\tilde{A}$  is of size  $r \times r$ .

- Solving the eigenvalue problem of  $\tilde{A}$  is easier than solving the one for  $A$  directly. The eigenvalues  $\lambda$  and the eigenvectors  $\omega$  of the  $\tilde{A}$  that approximate some of the eigenvalues of the full system  $A$  are given by:

$$\begin{aligned} A U_r &\approx U_r \tilde{A} \\ A U_r &\approx U_r \omega \lambda \omega^{-1} \\ A(U_r \omega) &\approx (U_r \omega) \lambda \end{aligned} \quad (4.2.9)$$

- The DMD modes that represent the high dimensional data are given by:

$$\begin{aligned}\Phi &= U\omega \\ \Phi &= \mathbf{X}'V\Sigma^{-1}\omega\end{aligned}\quad (4.2.10)$$

The dynamic modes do not only contain information about dynamic structures, but also about the temporal evolution of patterns within a data sequence.

- After deriving the truncated linear model using the  $\tilde{A}$  matrix, the DMD modes can be used to reconstruct and predict the future response of the dynamic system. This is obtained by using the known linear solution equation of the discrete linear system:

$$\hat{X}(t) = \Phi\lambda^t b_0 \quad (4.2.11)$$

where  $b_0 \approx \Phi^\dagger \bar{\mathbf{x}}_1$ ,  $\Phi^\dagger$  denotes the Moore-Penrose pseudo-inverse.

Taking the  $\lambda_i$  as the diagonal elements of the eigenvalues of the dynamic modes, the growth-decay and frequencies of the corresponding DMD modes are contained. Hence the temporal evolution is then formed via the Vandermonde matrix  $V(f)$ , which raises its column vector to the appropriate power.  $V(f)$  has a size of  $(L-1) \times (f+1)$  where  $f$  is the number of future vectors that are to be forecast and is defined as:

$$V(f) = \begin{pmatrix} 1 & \lambda_1^1 & \lambda_1^2 & \dots & \lambda_1^f \\ 1 & \lambda_2^1 & \lambda_2^2 & \dots & \lambda_2^f \\ \dots & \dots & \dots & \dots & \dots \\ 1 & \lambda_{L-1}^1 & \lambda_{L-1}^2 & \dots & \lambda_{L-1}^f \end{pmatrix} \quad (4.2.12)$$

If  $f > L$ , the Vandermonde matrix is estimating a forecast.

The continuous counterpart of the linear solution is :

$$x(t) = \sum_{j=1}^n b_j \phi_j \exp(\mu_j t) = \Phi \text{diag}(\exp(\mu_j t)) b_0 \quad (4.2.13)$$

Each of DMD modes consists of the corresponding frequency  $\mu_j$  that is defined by:

$$\mu_i = \frac{\log(\lambda_j)}{dt}$$

where  $dt$  is the lag between the vectors  $\bar{x}_i$  and  $\bar{x}_{i+1}$ . The real part of  $\mu_i$  indicates the growth or decay of the DMD modes, while the imaginary part of  $\mu_i$  drives oscillations in the DMD modes.

- The DMD reconstruction of the data  $\mathbf{X}$  at any instance  $f$  after the initial vector  $\bar{X}$  is given by :

$$\mathbf{X} = \Phi(V(f) \circ b_0) \quad (4.2.14)$$

Where  $b_0 \approx \Phi^{-1}\mathbf{X}$  and contains the initial amplitudes of the dynamic modes and the  $\circ$  indicates the element-wise multiplication of  $b_0$  with every column of  $V(f)$ .

### 4.2.1 Hankel DMD

The Hankel DMD or else HAVOK (Hankel Alternative Of Koopman) is an alternative approach of the traditional DMD algorithm. It has gained lots of attention due to its ability of revealing hidden structures of the given data, where the DMD fails to [76].

Although DMD was originally designed for analysing and predicting fluid flow spatial-temporal data, it is also used for temporal series. DMD analyses data-driven spatial-temporal or temporal data, exploiting the low order structure of the data using the minimum amount of computational resources.

DMD generates a linear time-invariant state space system that tries to span the time-series principal modes of the given data set. While approximating systems with oscillatory or quasi-periodic behaviour, the DMD fails to identify a representative model. Hidden structures that are not revealed in the Singular Value Decomposition step of the DMD, render the DMD incapable of identifying the system dynamics [76].

The Hankel matrix DMD is an alternative approach that enables the estimation of linear models that can capture the oscillatory or periodic behaviour of non linear systems. The Hankel introduces new block series of the input data by introducing time delayed block series of them [77]. Hence, it increases artificially the system's order. As a consequence, the amount of available information for model prediction is increased without the need of inserting new data. Hence, DMD manages to capture modes that could not be available in a first order system.

The Hankel matrix of the input data has the following structure:

$$\mathbf{X}_1 = \begin{bmatrix} x_1 & x_2 & \cdots & x_j \\ x_2 & x_3 & \cdots & x_{j+1} \\ \vdots & \vdots & \vdots & \vdots \\ x_i & x_{i+1} & \cdots & x_{j+i-1} \end{bmatrix} \quad (4.2.15)$$

Re-arranging the input data into the Hankel form, new state variables are created. Hence, the number of modes is also increased, maintaining the system order as low as possible. Furthermore, in the reconstruction step of the linear system, the initial condition  $b_0$  includes more information due to the increased model size, leading to more accurate forecast model.

The drawback of the Hankel matrix form DMD is that the introduction of the new variables reduces the number of samples of the training data set as shown in equation 4.2.15. Hence, the number of these new variables (number of rows the Hankel matrix) has to be a balance between the ability to detect dominant modes and the accuracy of the estimated model.

The behaviour of DMD for long term forecasting time-series with a periodical or quasi-periodical behaviour will be addressed below and a straight comparison with the HAVOK will be evaluated. Moreover, these methods will be compared over objects moving on a periodic fashion and systems with linear dependencies, revealing the advantages and flaws of these approaches.

### 4.3 Multiresolution Model Representations

The dynamic mode decomposition is used to decompose big set of data into their first principal components, investigating the behaviour and also predicting the future behaviour of the given high dimensional dynamic system with the minimum computational cost. Forecasting is a broad and important concept in many fields such as disease modelling, robotics, neuroscience, finance and many more and the DMD is extensively used in these applications.

As discussed in Chapter 1, modern systems in engineering are governed by multiscale characteristics. Therefore, different multiscale POD and DMD approaches are developed into this direction. Hence, before the demonstration of the various examples, two DMD algorithms in the field of multiresolution analysis will be addressed, the multiscale POD and the multiresolution DMD.

### 4.3.1 Multiscale POD

The multiscale POD (mPOD) combines the Multi-resolution Analysis (MRA) and the standard Proper Orthogonal Decomposition (POD) in order to allow for the optimal compromise between decomposition convergence and distinct separation of its modes [95–97]. The fundamental idea behind the mPOD is that POD is applied at different scales of resolution, each retaining non-overlapping fraction of the frequency spectra. Hence, the temporal basis is computed via a combination of the Multi-resolution analysis and the eigenvalue decomposition.

The mPOD applies the multiresolution analysis MRA over the temporal correlation matrix  $K = \mathbf{X}^T \mathbf{X}$  aiming at reducing the computational cost of this operation. The multiresolution analysis (MRA) in the frequency domain is introduced via the 2D Fourier transform of the correlation matrix [97]. The mPOD links three Fourier pairs which are related to the time evolution of the data ( $\mathbf{X}$ , row-wise), the temporal structures of the POD modes ( $V_{\mathcal{P}}$ , column-wise) and the temporal correlation matrix ( $K$ , over both columns and rows). These equations are written as follows:

$$\widehat{\mathbf{X}} = \mathbf{X} \bar{V}_{\mathcal{F}} \iff \mathbf{X} = \widehat{\mathbf{X}} V_{\mathcal{F}} \quad (4.3.1)$$

$$\widehat{V}_{\mathcal{P}} = \bar{V}_{\mathcal{F}} V_{\mathcal{P}} \iff V_{\mathcal{P}} = V_{\mathcal{F}} \widehat{V}_{\mathcal{P}} \quad (4.3.2)$$

$$\widehat{K} = \bar{V}_{\mathcal{F}} K \bar{V}_{\mathcal{F}} \iff K = V_{\mathcal{F}} \widehat{K} V_{\mathcal{F}}. \quad (4.3.3)$$

The correlation matrix  $K$  can be described in the frequency domain as follows:

$$K_{\mathcal{F}} = \widehat{\mathbf{X}}^\dagger \widehat{\mathbf{X}} = V_{\mathcal{F}} [\mathbf{X}^\dagger \mathbf{X}] \bar{V}_{\mathcal{F}} = V_{\mathcal{F}} K \bar{V}_{\mathcal{F}} \iff K = \bar{V}_{\mathcal{F}} K_{\mathcal{F}} V_{\mathcal{F}} \quad (4.3.4)$$

The mPOD is based on the MRA, meaning that the correlation matrix is decomposed into different scales based on filter banks, designed to separate fractions of the frequency spectra. The idea is to split the spectra of the Fourier transform of the temporal evolution (row-wise) of the data  $\widehat{\mathbf{X}}_p = \mathbf{X}_p \bar{V}_{\mathcal{F}}$ , for a given location  $\mathbf{i}_p$ , that is  $\mathbf{x}_p[k] = \mathbf{X}[\mathbf{i}_p, k]$  into  $M$  scales, each retaining a fraction with imperceptible overlapping as illustrated in figure 4.3.1. The frequency bandwidths of these scales  $\Delta f_m = f_{m+1}^c - f_m^c$ , with  $m \in [1, \dots, M-1]$ , are defined by a frequency vector  $F_V = [f_1^c, f_0^c, \dots, f_{M-1}^c]$ .

As figure 4.3.1 shows, a low-pass filter with cut off frequency  $f_1^c$ , a high-pass filter with cut off  $f_{M-1}^c$ , and  $M-2$  band-pass filters between these, are required. These filters are constructed from the set of  $m$  low-pass filters with

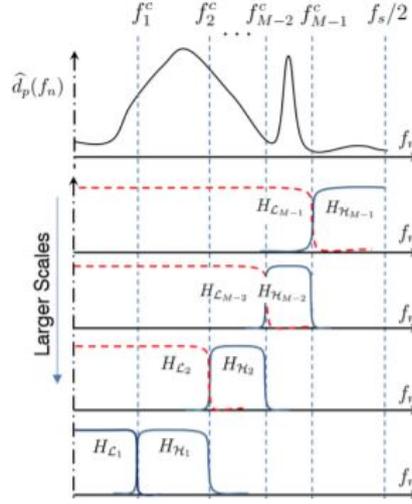


Figure 4.3.1: Multiscale POD  
[98]

transfer function  $\mathcal{H}_{\mathcal{L}_m}$ . Only the first low pass filter ( $\mathcal{H}_{\mathcal{L}_1}$ ) is recovered and the remaining are only used to build the bandpass filters as complementary differences  $H_{\mathcal{H}_M} = H_{\mathcal{L}_{M+1}} - H_{\mathcal{L}_M}$ . The finest one is expressed by  $H_{\mathcal{H}_M} = 1 - \mathcal{H}_{\mathcal{L}_M}$  [98].

The set of the derived filter transfer functions are by construction such that  $H_{\mathcal{L}_1} + H_{\mathcal{H}_1} + \dots + H_{\mathcal{H}_{M-1}} = \underline{1}$ . As a result the entire spectra of the dataset is reconstructed and is expressed as follows.

$$\mathbf{X}_m = \underbrace{\left[ (\mathbf{X} \bar{V}_{\mathcal{F}}) \odot H'_m \right]}_{\hat{\mathbf{X}}_m} V_{\mathcal{F}}. \quad (4.3.5)$$

Extending the equation 4.3.5, the correlation matrix  $K_m = \mathbf{X}_m^\dagger \mathbf{X}_m$  from each scale contribution is written as:

$$K_m = \bar{V}_{\mathcal{F}} \left[ \left( \hat{\mathbf{X}} \odot H'_m \right)^\dagger \left( \hat{\mathbf{X}} \odot H'_m \right) \right] V_{\mathcal{F}} = \bar{V}_{\mathcal{F}} \left[ \left( \hat{\mathbf{X}}^\dagger \hat{\mathbf{X}} \right) \odot \left( (H'_m)^\dagger \odot H'_m \right) \right] V_{\mathcal{F}} \quad (4.3.6)$$

Introducing the cross-spectral density matrix  $K_{\mathcal{F}}$  in equation 4.3.4 and expanding it in the 2D spectrum by using the 2D transfer function of the filter banks as  $\underline{\mathcal{H}}_m = (H'_m)^\dagger \odot H'_m$ , it yields:

$$K_m = \bar{V}_{\mathcal{F}} \underbrace{[K_{\mathcal{F}} \odot \underline{\mathcal{H}}_m]}_{K_{\mathcal{F}m}} V_{\mathcal{F}} = V_{\mathcal{F}} \underbrace{[\widehat{K} \odot \underline{\mathcal{H}}_m]}_{\widehat{K}_m} V_{\mathcal{F}} \quad (4.3.7)$$

The equation 4.3.7 reveals the foundation of the mPOD. The filtered cross-spectral density  $K_{\mathcal{F}m} = K_{\mathcal{F}} \odot \underline{\mathcal{H}}_m$  shares the same eigenvalues of the correlation of filtered data  $K_m = \mathbf{X}_m^\dagger \mathbf{X}_m$ .

Secondly, the eigenvectors of  $K_{\mathcal{F}}$  are the DFT of the POD modes of  $\mathbf{X}_m$  :

$$K_{\mathcal{F}m} = V_{\mathcal{F}} K_m \bar{V}_{\mathcal{F}} = V_{\mathcal{F}} \left[ V_{\mathcal{P}m} \Sigma_{\mathcal{P}m}^2 \Psi_{\mathcal{P}m}^\dagger \right] \bar{V}_{\mathcal{F}} = \left( P_\pi \widehat{V}_{\mathcal{P}m} \right) \Sigma_{\mathcal{P}m}^2 \left( P_\pi \widehat{V}_{\mathcal{P}m} \right)^\dagger \quad (4.3.8)$$

where the permutation matrix  $P_\pi$  can be recovered by applying the DFT operator twice:

$$P_\pi = \Psi_{\mathcal{F}} \Psi_{\mathcal{F}} = \bar{\Psi}_{\mathcal{F}} \bar{\Psi}_{\mathcal{F}} = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & & 0 & 1 \\ 0 & 0 & & 1 & \vdots \\ \vdots & \vdots & . & & \vdots \\ 0 & 1 & 0 & \dots & 0 \end{bmatrix} \quad (4.3.9)$$

The impact of the filter on the POD modes is revealed by the diagonal entries of  $K_{\mathcal{F}m}$  and  $K_{\mathcal{F}}$ . Finally using equation 4.3.8:

$$K_{\mathcal{F}m}[i, j] = K_{\mathcal{F}}[i, j] \odot \underline{\mathcal{H}}_m[i, j] = \sum_{r=1}^{n_t} \sigma_{\mathcal{P}mr}^2 \widehat{\psi}_{\mathcal{P}mr}[i] \widehat{\psi}_{\mathcal{P}mr}^\dagger[j] \quad (4.3.10)$$

In occasions where there is no frequency-overlapping, the eigenspaces of the approximation terms are orthogonal complements. This property is the main characteristic that consists the mPOD different to other multi-scale methods such as Continuous Wavelet Transform (CWT). In CWT, the temporal basis is constructed by shifting and dilating a 'mother' function. It is different to the occasion of the multi-resolution DMD (mr-DMD), where the temporal basis is constructed by performing DMD on different portions of the datasets [98].

These decompositions potentially produce high redundancy and poor convergence since the basis is larger than  $n_t$ . On the other hand, each of the

basis elements in the mPOD exist over the entire time domain, while CWT or mr-DMD produce different bases for different portions of the time domain. This leads to decompositions more complex but could be of better use on more complicated models [98].

The mPOD will be evaluated into various examples and will be compared with all relevant multiscale reduced order models.

In a nutshell, the mPOD algorithm is summarized as follows:

### 4.3.2 Multiresolution DMD

This method implements a Fourier decomposition of correlated spatial activity of the video frames in time. Multiresolution DMD (mr-DMD) has the ability to distinguish the stationary background from the dynamic foreground by differentiating between the near-zero Fourier modes and the remaining modes bounded away from the origin, respectively [95].

The innovation of the multi-resolution DMD is that it allows for further separation of dynamic content in the video or in the selected snapshot series. As a result, it allows the separation of components that are happening on different time scales. Hence, the multi-resolution framework reveals the multiscale characteristics of the given data-driven system.

Initially, multiresolution DMD was introduced for implementing the background/foreground separation in video frames sequences, removing low frequency contents but can be also used for multiscale systems, connecting macroscale and microscale structures. By recursively refining the sampling time of the data of interest, this method removes temporal or spatial features in every scale [97]. Each DMD mode has a corresponding frequency (Fourier mode). By removing the slow modes in every scale, a full separation of the slow (low-rank) and fast (sparse) information is accomplished.

The main concept of every step in multiresolution DMD is summarized in the following equation, where  $\omega_j \leq \epsilon$ , where  $\epsilon$  is a low frequency threshold.

$$X = X_{\text{DMD}}^{\text{Low-Rank}} + X_{\text{DMD}}^{\text{Sparse}} \quad (4.3.14)$$

$$X_{\text{DMD}} = \underbrace{b_p \phi_p e^{\mu_p t}}_{\text{Background Video}} + \underbrace{\sum_{j \neq p} b_j \phi_j e^{\mu_j t}}_{\text{Foreground Video}} \quad (4.3.15)$$

Based on Fig.4.3.2, in the initial pass, the slowest  $m_1$  modes are removed and DMD is once again performed with now only  $\frac{M}{2}$  snapshots and the process is continued.

---

**Algorithm 2** Multiscale POD Algorithm
 

---

1. Assembly of the data matrix
2. Computation of the temporal correlation matrix  $K$  and its Fourier transform  $\widehat{K}$
3. Computation of the frequency splitting vector  $F_V$  and construction of the set of filter banks transfer functions
4. Computation of the approximation and diagonal detail terms of the 2D transfer functions, that are  $\mathcal{H}_{\mathcal{L}_1} = (H'_{\mathcal{L}_1})^\dagger \odot H'_{\mathcal{L}_1}$  and  $\mathcal{H}_{\mathcal{H}_m} = (H'_{\mathcal{H}_m})^\dagger \odot H'_{\mathcal{H}_m}$ . Subtracting the horizontal and vertical details in each scale, the correlation matrix is approximated as:

$$K \approx V_{\mathcal{F}} \left[ \widehat{K} \odot \mathcal{H}_{\mathcal{L}_1} \right] V_{\mathcal{F}} + \sum_{m=1}^M V_{\mathcal{F}} \left[ \widehat{K} \odot \mathcal{H}_{\mathcal{H}_m} \right] V_{\mathcal{F}} \approx K_{\mathcal{L}_1} + \sum_{m=1}^{M-1} K_{\mathcal{H}_m} \quad (4.3.11)$$

Each of these contributions is a symmetric, real and positive definite matrix:

$$K \approx V_{\mathcal{L}_1} \Sigma_{\mathcal{L}_1}^2 V_{\mathcal{L}_1}^T + \sum_{m=1}^{M-1} V_{\mathcal{H}_m} \Sigma_{\mathcal{H}_m}^2 V_{\mathcal{H}_m}^T \quad (4.3.12)$$

5. Computation of all the contributions of all the scales into  $V_{\mathcal{M}}^0$  as:

$$V_{\mathcal{M}}^0 = [V_{\mathcal{L}_1}, V_{\mathcal{H}_1}, V_{\mathcal{H}_2} \dots V_{\mathcal{H}_{M-1}}] P_{\Sigma} \quad (4.3.13)$$

6. Computation of the spatial basis  $V_{\mathcal{M}} = DV_{\mathcal{M}} \Sigma_{\mathcal{M}}^{-1}$  and sort  $V_{\mathcal{M}}$  in descending order of energy contribution
-

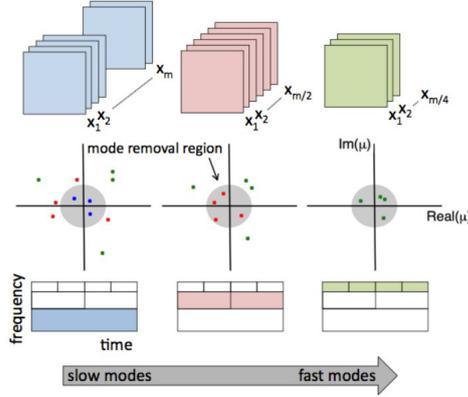


Figure 4.3.2: Multiresolution DMD  
[99]

In the first step, mathematically, DMD is represented as:

$$\begin{aligned} \mathbf{x}_{\text{mr-DMD}}(t) &= \sum_{k=1}^M b_k \phi_k^{(1)}(t) \exp(\mu_k t) \\ &= \sum_{k=1}^{m_1} b_k \phi_k^{(1)}(t) \exp(\mu_k t) + \sum_{k=m_1+1}^M b_k \phi_k^{(1)}(t) \exp(\mu_k t) \end{aligned} \quad (4.3.16)$$

where the  $\phi_k^{(1)}$  represent the DMD modes computed from the full  $M$  snapshots. The first part of the sum in equation 4.3.16 indicates the slow modes and the second part of the sum indicates the fast modes.

In the next step, DMD is performed on the  $\frac{M}{2}$  snapshots, meaning the fast modes of the previous scale. This fragment is separated into a new of matrices:

$$\mathbf{X}_{M/2} = \mathbf{X}_{M/2}^{(1)} + \mathbf{X}_{M/2}^{(2)} \quad (4.3.17)$$

where the first matrix contains the first  $M/2$  snapshots and the second matrix contains the remaining  $\frac{M}{2}$  snapshots.

This iterative algorithm functions in a recursively pattern by removing slow frequency dynamics for the remaining  $\frac{M}{4}, \frac{M}{8}$  etc dataset, till a representative multiresolution model is achieved according to the needs of each application.

The full approximate solution of the multiresolution DMD in  $n$  scales is presented as:

$$\begin{aligned} x_{\text{DMD}}(t) &= \sum_{k=1}^{m_1} b_k^{(1)} \phi_{\mathbf{k}}^{(1)}(\mathbf{x}) \exp\left(\mu_k^{(1)} t\right) \\ &+ \sum_{k=1}^{m_2} b_k^{(2)} \phi_{\mathbf{k}}^{(2)}(\mathbf{x}) \exp\left(\mu_k^{(2)} t\right) \\ &+ \sum_{k=1}^{m_3} b_k^{(3)} \phi_{\mathbf{k}}^{(3)}(\mathbf{x}) \exp\left(\mu_k^{(3)} t\right) + \dots \end{aligned} \quad (4.3.18)$$

where the  $\phi_{\mathbf{k}}^{(k)}$  and  $\mu_k^{(k)}$  are the DMD modes and DMD eigenvalues at the  $k^{\text{th}}$  level of decomposition, the  $b_k^{(k)}$  are the initial amplitudes of the DMD modes, and the  $m_k$  are the number of slow-modes retained at each level of decomposition. This decomposition method reveals different spatial-temporal DMD modes that are used to represent key multi-scale features. Thus, there is not a single set of modes that dominates the SVD decomposition and potentially marginalizes features at other time scales [99].

## 4.4 Comparison of DMD Methods

In this section, the different DMD methods that described above, are applied into several temporal and spatial examples. The aim is to present the occasions where vanilla DMD is incapable of creating representative dynamic models. Moreover, it will be shown that there are occasions where the multiresolution DMD and POD methods are superior than the vanilla DMD but they are yet dealing with problems when it comes to dataset with abrupt changes among the selected segments of data. Hence, different examples that will compare all existing DMD algorithms will be illustrated and reveal the occasions where more accurate multiresolution algorithms need to be derived.

In a nutshell, the problems of the DMD alternatives to forecast long term time-series with a periodical or quasi-periodical behaviour will be addressed. Approaches such as Multiresolution DMD, Wavelet DMD and the Hankel DMD or higher order dynamic mode decomposition (HODMD) or HAVOK (Hankel Alternative View Of Koopman) will be exhibited. The HAVOK will be addressed analytically and will be shown that outperforms all methods mentioned above when it comes to capture the periodic or almost periodic dynamics of non-linear systems in a nearly perfect way.

Different examples of periodic or almost periodic dynamics will be illustrated, as well as examples with hidden structures where the vanilla DMD fails to identify a representative model. A complete analysis will reveal which methods outperform vanilla DMD and which method is the most consistent among all DMD alternatives.

#### 4.4.1 Periodical time series

In this first example, a straight comparison of the standard DMD and Hankel formed DMD will be addressed over some simple oscillatory temporal series. It will be shown that the Hankel DMD reveals hidden structures and outperforms the original DMD.

The first example that will be investigated is the time series evolution of a square wave. The square wave is a special case of a pulse wave which allows variable durations at minimum and maximum and is extensively used in engineering and signal processing. The ratio between the total period of the pulse and the high period is called duty cycle and in the case of the square wave, it is fixed to 50%. The square wave can be expressed mathematically as a sum of sinusoidal terms.

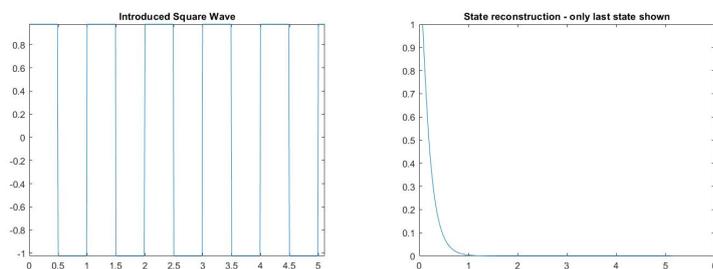
The following equation represents a zero mean square wave expanding in time.

$$x(t) = \text{sgn} \left( \sin \frac{2\pi t}{T} \right) = \text{sgn}(\sin 2\pi ft) \quad (4.4.1)$$

In this example, the simulated data will be collected and will be feed into a DMD model that will try to identify a reduced order model. The validity and accuracy of the derived dynamical model will be investigated by reconstructing the data and comparing them with the input data.

The figure 4.4.1 illustrates the behaviour of the DMD when it comes on reconstructing the given dataset. It is shown that DMD fails to identify the oscillatory behaviour of the introduced data. A simple way to identify the source of this issue, is by checking the eigenvalues of the derived linear model in the DMD algorithm. It is shown that the DMD model includes only one real eigenvalue, hence the derived model is losing insight of the oscillatory behaviour of the introduced data.

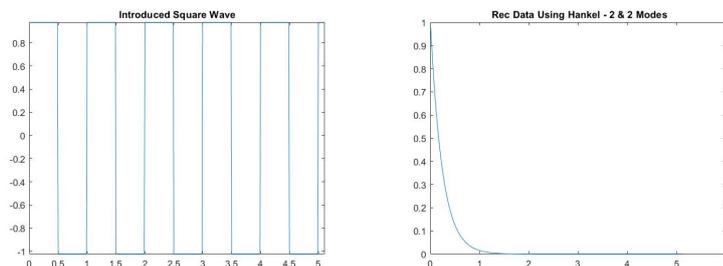
The weakness of DMD to reconstruct the input data will be attempted to be resolved by using Hankel DMD. In the following images, different sizes of Hankel blocks are used in the DMD input data. As described in the previous chapter, the introduction of Hankel form block rows of the input data leads to an artificial increase of the system order, forming higher order models.



(a) Simulation of zero mean square wave over time internal 0-5s  
 (b) Reconstructed data of the DMD model

Figure 4.4.1: Comparison of the given dataset and the output series of the derived dynamic model after applying the conventional DMD

Figure 4.4.1 demonstrates and compares the reconstructed data after the derivation of second order model using DMD with modified input data using two block rows.



(a) Simulation of zero mean square wave over time internal 0-5s  
 (b) Reconstructed data of the DMD model with a two block row Hankel input matrix

Figure 4.4.2: Comparison of the given dataset and the output series of the derived dynamic model after applying the Hankel-2 DMD

It is shown that the augmented DMD method still fails to reconstruct the given data. Hence, the size of the Hankel matrix will be increased gradually, identifying the threshold where the order of the introduced data is high enough, in order to fully reconstruct the given data.

It will be shown that each additional block row introduces an extra odd sinusoidal term to the derived dynamic system. Hence, after a specific number of block rows, the number of sinusoidal terms is adequate enough to reveal the oscillatory behaviour of the input data. Each additional block row, adds an extra complex conjugate pair of eigenvalues. Hence, after a

specific number of block rows, the amount of sinusoidal terms is adequate enough to identify the oscillatory behaviour of the input data.

The figures below, demonstrate the reconstructed data from the derived DMD dynamic model, using 16 and 32 block rows for the input data respectively. Figures indicate that the model starts to identify the oscillatory behaviour, but it lacks accuracy.

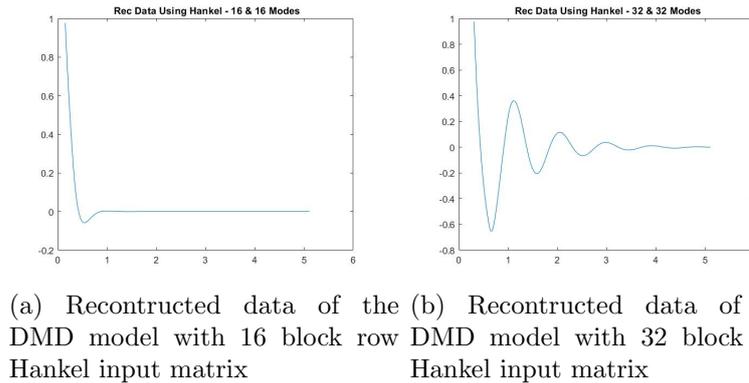


Figure 4.4.3: Comparison of the output series of the derived dynamic model after applying Hankel-16 and Hankel-32 DMD

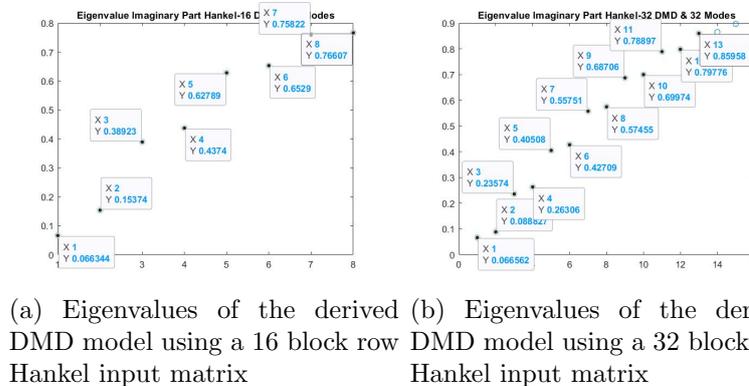
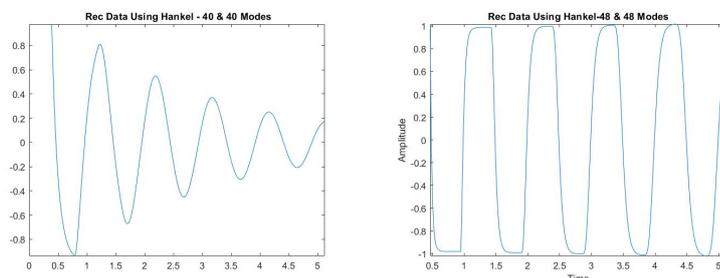


Figure 4.4.4: Eigenvalues presentation of the derived dynamic model after applying Hankel-16 and Hankel-32 DMD

Even by using 40 Hankel block rows, DMD still fails to approximate the introduced data set. A fully representative data reconstruction comes after using 50 Hankel block row input data.

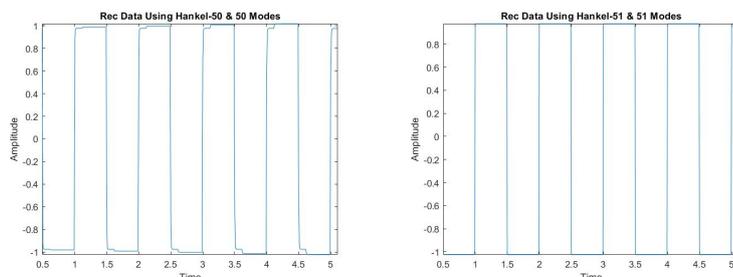
In order to justify that, the Singular Value Decomposition step of the augmented DMD is presented. The SVD step identifies the amount of dominant modes of the introduced system. In order to make it more clear, we



(a) Reconstructed data of the DMD model with 40 block row Hankel input matrix

(b) Reconstructed data of the DMD model with 48 block row Hankel input matrix

Figure 4.4.5: Comparison of the output series of the derived dynamic model after applying Hankel-40 and Hankel-48 DMD



(a) Reconstructed data of the DMD model with 50 block row Hankel input matrix

(b) Reconstructed data of the DMD model with 51 block row Hankel input matrix

created a 100 block row Hankel input data and we applied the SVD step of the DMD on this amount of data. It is clearly shown that there are 51 dominant modes in this system, so in order to fully reconstruct the initial system, we need 51 modes. Even with 50, it is shown that there is some error when the data makes a jump from the negative to the positive values.

This example shows the importance of the Hankel DMD and its capability of identifying hidden structures and system characteristics over the conventional DMD when it comes to oscillatory or higher order data.

#### 4.4.2 Sinusoidal time series

A second example where Hankel DMD outperforms the conventional DMD will be analysed in this section. We will analyse DMD and Hankel DMD behaviour over a sinusoidal temporal series.

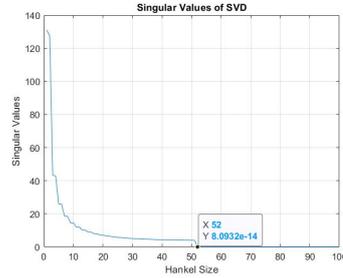


Figure 4.4.6: Singular Value Decomposition plot of the given dataset

A sinusoidal signal is described under the following equation:

$$x(t) = A \cos(\omega_0 t) = \frac{A}{2} \exp(j\omega_0 t) + \frac{A}{2} \exp(-j\omega_0 t) \quad (4.4.2)$$

Analysing equation 4.4.2, every sinusoidal signal contains two modes, the  $\exp(j\omega_0 t)$  and  $\exp(-j\omega_0 t)$  and this is the reason why DMD fails to identify a representative model due to the fact that it extracts a first order model with contains only one mode. The following figure verifies this speculation.

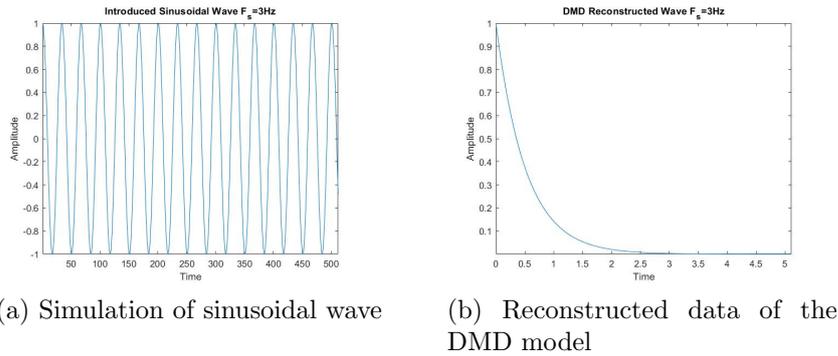


Figure 4.4.7: Comparison of the given dataset and the output series of the derived dynamic model after applying the conventional DMD

In this example, the DMD requires at least two modes in order to identify the oscillatory behaviour of the system. Using the conventional DMD, for this example of a sinusoidal signal with frequency  $f_s = 3\text{Hz}$ , there is only one eigenvalue  $\lambda_1 = 0.9807$  in contrast to the Hankel-2 DMD where there are two complex conjugate eigenvalues  $\lambda_1 = 0.9823 + 0.1874i$  and  $\lambda_2 = 0.9823 - 0.1874i$  revealing the oscillatory characteristics of the introduced dynamics.

In conclusion, Hankel DMD is a powerful that can increase the amount of the available data and the system order without the need of additional data.

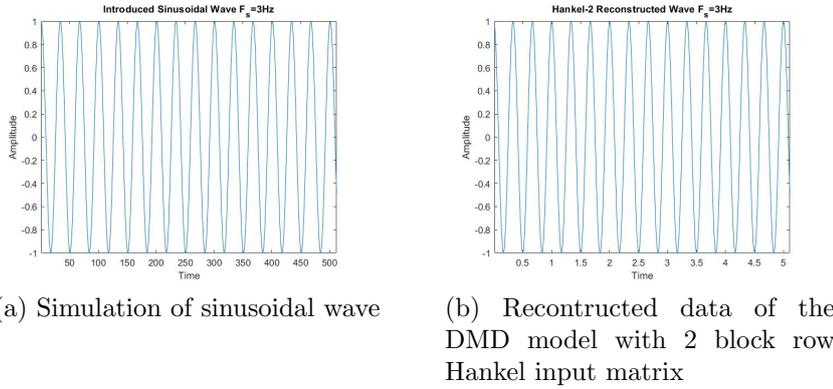


Figure 4.4.8: Comparison of the given dataset and the output series of the derived dynamic model after applying the Hankel-2 DMD

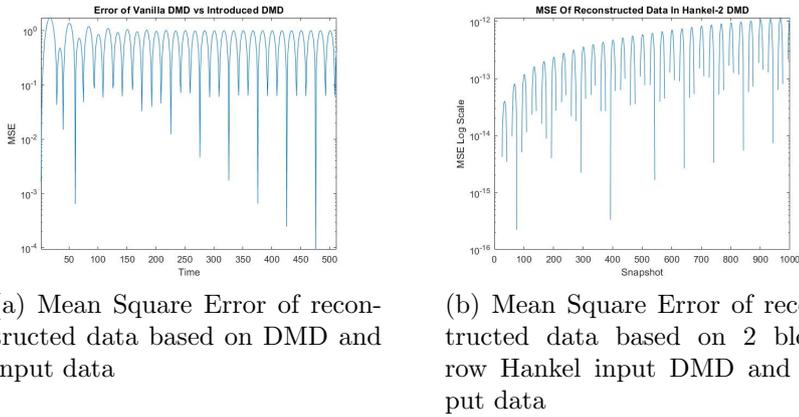


Figure 4.4.9: Comparison of the MSE of output series between the derived dynamic models after applying the conventional DMD and the Hankel-2 DMD

This property makes the estimation of a dynamic model more accurate and precise compared to the conventional DMD.

### 4.4.3 Diffusion Equation Modelling

The DMD fails when it comes to periodical temporal data series. In this section, the behaviour of DMD and Hankel DMD over spatio-temporal data will be analysed. In these datasets, there is more information available per data set, consisting DMD a more suitable tool.

The two methods will be further analysed and their performance will be evaluated over the diffusion equation. The diffusion equation is given by the following equation:

$$\frac{\partial \phi(\mathbf{r}, t)}{\partial t} = \nabla \cdot [D(\phi, \mathbf{r}) \nabla \phi(\mathbf{r}, t)] \quad (4.4.3)$$

where,  $D(\phi, \mathbf{r})$  is the collective diffusion coefficient for density  $\phi$  at location  $\mathbf{r}$ . When  $D$  is constant, then the equation is identical to the heat equation.

The eigenvalues of the conventional DMD and the Hankel-2 DMD are shown in the following figure. In this example, based on the SVD graph, the standard DMD demonstrates 11 dominant modes. Additionally, it is shown that there are complex conjugate eigenvalues, indicating that the DMD captures the oscillatory behaviour of the given system. On the other hand, the Hankel DMD, by introducing extra variables, the amount of available modes is higher, 20 in this example. The biggest difference is that the eigenvalues of the Hankel-DMD are not concentrated entirely on the unity circle. This is an indication that the Hankel-DMD reveals different dynamics inside the system. In order to check the importance and necessity of these extra modes, the Mean Square Error (MSE) in both approaches is examined.

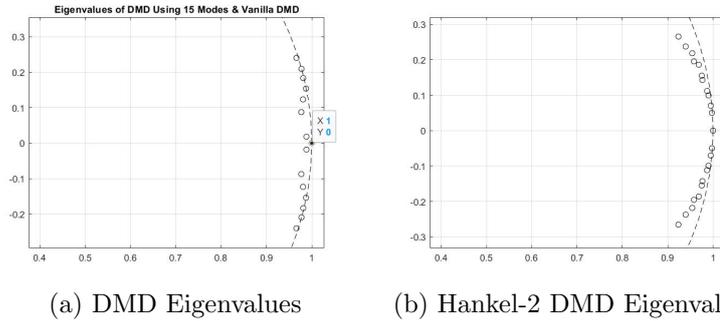


Figure 4.4.10: Eigendecomposition of the two derived dynamic models based on conventional DMD and the Hankel-2 DMD

By observing the Mean Square Error both in time and space for the DMD and the Hankel DMD, it is clearly shown that the Hankel-DMD significantly improves the performance of the derived dynamic model over the one of the standard DMD.

The artificial increase of the number of introduced data and the arrangement into Hankel block rows, turns DMD into a very efficient tool for forecasting the temporal evolution of the given data. In all cases, the Mean Square Error is significantly decreased and even in the last example where the standard DMD provides better results compared to the previous example, the Hankel DMD outperforms significantly the standard DMD.

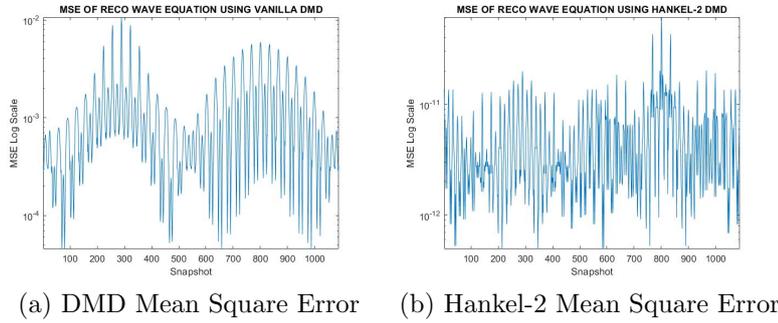
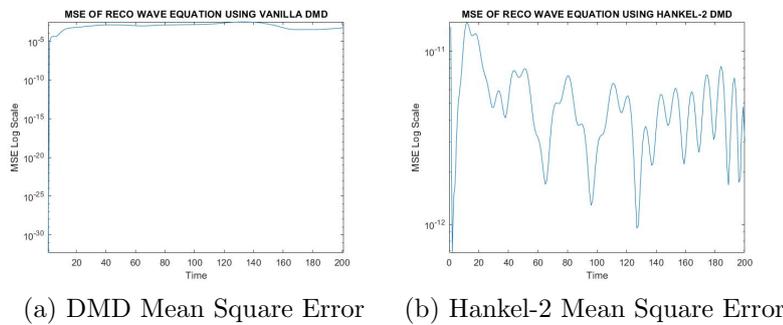


Figure 4.4.11: Comparison of the output MSE between the dynamic models derived from conventional DMD and Hankel-2 DMD



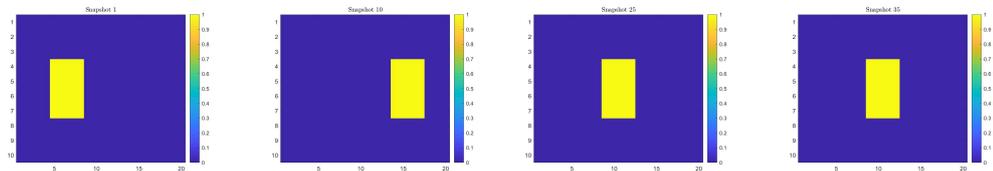
#### 4.4.4 Moving Image Data

The final example where the performance of the two different DMD methods will be investigated, concerns the motion of a image inside another image in a periodical fashion. A  $10 \times 20$  pixels steady image will be introduced where another smaller image  $3 \times 4$  pixels is moving vertically one pixel per time-step for a specific time period  $t$ .

The images below illustrate this example in different time steps, presenting the motion of the smaller image inside the bigger blue image.

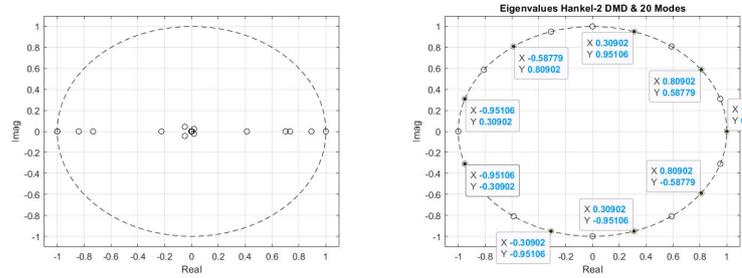
The oscillatory behaviour and the ability of the DMD to capture the periodical and recurrent behaviour of the introduced data, can be easily seen by the eigendecomposition of the derived linear model. Plotting the eigenvalues of each method, it is shown that the DMD presents only real eigenvalues that will result in a poor forecasting performance.

On the other hand, the eigendecomposition of the derived dynamic model based on the Hankel-DMD using two block rows, introduces eigenvalues that react as complex conjugate pairs. This denotes that the system presents a completely different behaviour to the conventional DMD model, which will be illustrated below.



(a) 1<sup>st</sup> Snapshot of (b) 10<sup>th</sup> Snapshot of (c) 25<sup>th</sup> Snapshot of (d) 35<sup>th</sup> Snapshot of Simulation Of Moving Simulation Of Moving Simulation Of Moving Simulation Of Moving Image Image Image Image

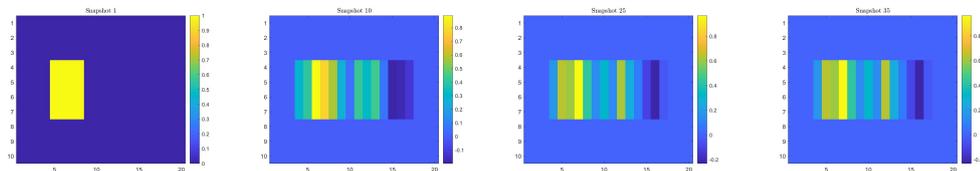
Figure 4.4.12: Simulation of the model of small  $3 \times 3$  image moving inside a  $10 \times 10$  image



(a) Eigenvalues of DMD (b) Eigenvalues of Hankel-2 DMD

Figure 4.4.13: Eigendecomposition plot of the conventional DMD & the Hankel-2 DMD methods

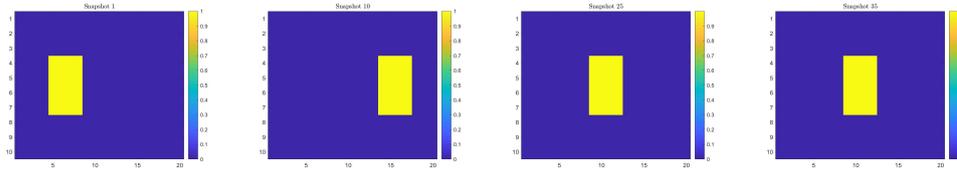
Plotting the system solutions for the standard DMD and the Hankel-2 DMD respectively, it is clearly shown that the standard DMD shows poor approximation and fails to identify the one step movement of the image per time step.



(a) 1<sup>st</sup> Snapshot of (b) 10<sup>th</sup> Snapshot of (c) 25<sup>th</sup> Snapshot of (d) 35<sup>th</sup> Snapshot of Simulation Of Recon-Simulation Of Recon-Simulation Of Recon- Simulation Of Recon- Reconstructed Moving Im-structured Moving Im-structured Moving Im-structured Moving Im-structured Moving Im-structured Moving Im-structured Moving Image Using DMD Image Using DMD Image Using DMD Image Using DMD

Figure 4.4.14: Simulation of specific reconstructed snapshots of the moving image based on the standard DMD method

On the other hand, the Hankel-2 DMD manages to reconstruct the given dataset perfectly as was predicted from the eigenvalue decomposition plot.



(a) 1<sup>st</sup> Snapshot of (b) 10<sup>th</sup> Snapshot of (c) 25<sup>th</sup> Snapshot of (d) 35<sup>th</sup> Snapshot of Simulation Of Recon-Simulation Of Recon-Simulation Of Recon-Simulation Of Reconstructed Moving Im-structured Moving Image Using Hankel-2age DMD Using Hankel-2age DMD Using Hankel-2age DMD Using Hankel-2age DMD

Figure 4.4.15: Simulation of specific reconstructed snapshots of the moving image based on Hankel-2 DMD method

The Hankel DMD by using only 2 block rows, shows great results and it is also enhanced by the Mean Square Error (MSE) plot. The MSE of the Hankel-DMD is of order  $10^{-10}$  compared to the introduced data which is close to 1.

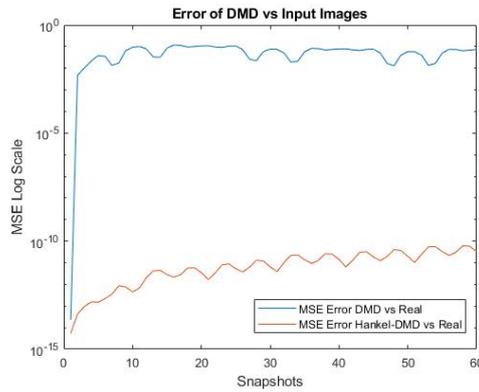


Figure 4.4.16: Comparison of the Mean Square Error in the reconstruction step between standard DMD and Hankel - 2 DMD

In all examples where the nature of data is or tends to be periodical and recurrent, the Hankel DMD outperforms the standard DMD in all cases. The dimension increase of the signal achieves to contain all principal modes making DMD a very effective tool.

### 4.4.5 Multiresolution POD and DMD over moving image

The same example will be addressed using the multiresolution POD and DMD algorithms in order to evaluate the performance of the existing multiresolution algorithms.

#### 4.4.5.1 Multiresolution DMD

In these example, the multiresolution DMD (mr-DMD), as described, separates the slow and fast moving dynamics according to the user needs and as a next step, it divides into half the remaining data and applies the method again. This process is repeated till the need of every example is achieved. It will be shown that the multiresolution DMD deals with issues when it comes to abrupt changes. Multiresolution DMD separates the data into sections and the fact that there is a small amount of available data, the DMD method is sensitive to abrupt changes and cannot reconstruct effectively the dataset. Data with abrupt changes as shown in the previous examples, can cause the DMD to fail deriving a representative model.

The same problems are revealed with the multiresolution DMD. Even in the occasion where the mr-DMD uses the Hankel formation on the input data, the algorithm is dealing with the same problems. The simulations took place with even 8 Hankel block rows that had a big computational cost and the mr-DMD could not solve the previous problem too.

In the following images, the Mean square error (MSE) between the initial data and the two multiresolution DMD methods (mr-DMD), the conventional mr-DMD and the Hankel-6 mr-DMD is presented.

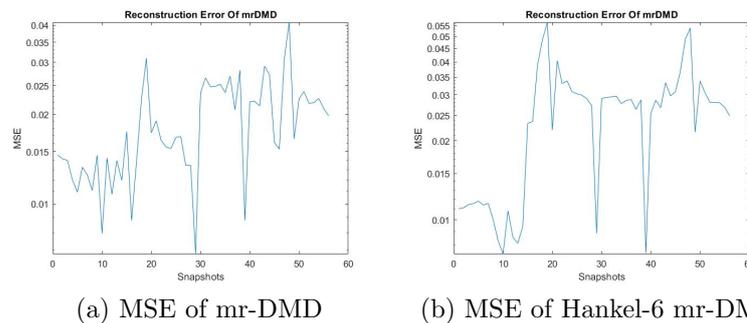


Figure 4.4.17: Comparison of the Mean Square Error in the reconstruction step of the mr-DMD using respectively

The abrupt changes into the different data segments render the mr-DMD

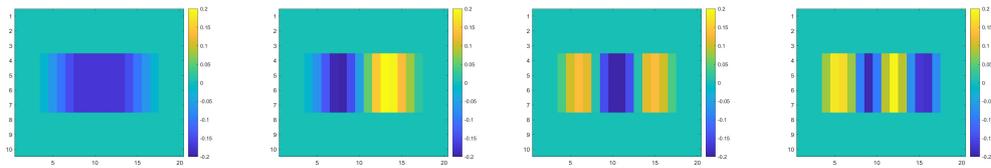
incapable of reconstructing the given data. The division of data in small segments hold the DMD performance back, which requires high amount of data in order to be efficient. The same example was repeated using more blocks of data into the Hankel matrices without any improvements in performance.

Hence, the mr-DMD does not serve its purpose in these occasions, hence an alternative multiresolution approach needs to be considered when it comes to data with abrupt changes.

#### 4.4.5.2 Multiresolution POD

The multiresolution POD (mr-POD) will be encountered too. The mr-POD algorithm will be tested in the same example as the mr-DMD. As presented above, the reconstructed images will be presented directly revealing if the segmentation of the data puts a toll on the loss of information.

Performing the mr-POD, it is shown that the algorithm cannot reveal the correct image motion. The MSE remains in the same levels as the one in the mr-DMD.



(a) 1<sup>st</sup> Snapshot of Simulation Of Reconstructed Moving Image Using mr-POD (b) 10<sup>th</sup> Snapshot of Simulation Of Reconstructed Moving Image Using mr-POD (c) 25<sup>th</sup> Snapshot of Simulation Of Reconstructed Moving Image Using mr-POD (d) 35<sup>th</sup> Snapshot of Simulation Of Reconstructed Moving Image Using mr-POD

Figure 4.4.18: Simulation of specific snapshots of the moving image by using the vanilla DMD method

Hence, in occasions where the data needs to be divided into different levels of resolution, a more efficient approach needs to be considered. In chapter 5, a novel multiresolution POD and DMD algorithm will be derived, overcoming the problems presented in this section.

## 4.5 Conclusions

In this chapter, the dominant reduced order identification methods were presented. POD and DMD are applied over various examples and the weaknesses of these methods were revealed. The key finding is that the standard DMD

---

and POD methods fail to derive accurate representations when it comes to data with abrupt changes and oscillatory behaviours. Hence, the importance of alternative solutions such the Hankel-DMD was addressed and presented over the same examples. Moreover, the multiresolution counterparts of these methods were addressed and applied in the same examples which dealt with the same issues.

Therefore, alternative solutions, especially for the multiresolution DMD and POD that manage to overcome these problems need to be encountered. Moreover, the performance of the Hankel-DMD needs to be evaluated in more complex datasets.

In the following chapter, a novel multiresolution DMD algorithm will be derived that will attempt to address the issues shown above. Moreover, the novel approach will be compared with the existing DMD algorithms in complex real time applications where their performance will be evaluated.

# Chapter 5

## Spatial Multiscale DMD

In the previous chapter, the dominance of the DMD method using Hankel input matrices over examples with abrupt changes, oscillatory behaviour and objects moving in periodic fashion was shown.

In this chapter, various examples comprising multiscale characteristics, mixed frequency effects and linear dependencies will be considered. It will be shown that the existing methods lack of accuracy and interpretability of all effects taking place. Therefore, the institution of multiscale reduced order approaches is essential.

A novel multiscale DMD and POD algorithm will be introduced. These approaches are based on wavelets which have the ability to isolate and present localized structures. By using wavelets, the DMD algorithm is capable of capturing basic features and details in different scale of resolution simultaneously. The wavelet decomposition method is modified in order to encapsulate as much information as possible and minimize the computational resources in terms of the number of DMD algorithms used simultaneously. Hence, they will give rise to characteristics and effects that cannot be revealed using single-scaled approaches or even the existing multiscale algorithms.

In this chapter, the Wavelet decomposition mathematical derivation will be introduced and modified for the purpose of DMD algorithms. It will be proven that the novel approach is consistent to the conventional and Hankel-DMD properties, hence it can substitute them. The proposed Wavelet-DMD algorithm will create modes that include only specific patterns in each scale of resolution and can separate mixed frequency effects.

Two complex examples will be considered. The first example simulates a system of two waves oscillating in different frequencies and amplitudes creating a system with mixed effects and dynamics. The target is to apply and compare the different DMD approaches, investigating their ability to give full insight of the different system dynamics. The second example introduces a

real problem in manufacturing. The data were received from the university of Leeds. During the production of the powder used in 3D printers, turbulences are observed, resulting in poor quality. The aim is to reveal the area where turbulences occur, meaning the concentration of powder particles in small areas, guiding the user to alter the direction and speed of the nozzles used to spray air and molten material.

## 5.1 2-D Wavelet Decomposition

The wavelet decomposition is applied over time-series signals but it is also applied over images under the multiresolution formulation with broad applications. It can reveal hidden characteristics in images for filtering purposes such as smoothing, sharpening, removing noise, and edge detection. The wavelet decomposition method applies a low pass and a high pass filter over the image coefficients. Therefore, in every scale of resolution, it derives a series of low and high frequency coefficients. Typically, the mathematical representation of a 2D wavelet decomposition of the low frequency coefficients  $W_\phi(j, m, n)$  is represented by:

$$W_\phi(j_0, m, n) = \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(x, y) \phi_{j_0, m, n}(x, y) \quad (5.1.1)$$

The corresponding wavelet function of the high frequency representation is given from the following equation:

$$W_\psi(j_0, m, n) = \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(x, y) \psi_{j_0, m, n}(x, y) \quad (5.1.2)$$

where  $m = (0, 1, \dots, M-1)$ ,  $n = (0, 1, \dots, N-1)$ ,  $j$  is equal to the number of selected scales  $j = 1, 2, \dots, J$ , where  $J$  is the number of scales and  $M, N$  are the rotations of  $x, y$  in  $\mathbb{Z}$  domain.

The high frequency representations are applied in both  $x$  and  $y$  axis, so three sub-band images are derived, the horizontal, vertical and diagonal representation of the images.

The low pass filters are fundamental tools for the majority of smoothing methods. An image is smoothed by decreasing the discrepancy between pixel values by averaging neighbouring pixels [27, 32]. Therefore, the use of low pass filters leads to retain the low frequency information while reducing the

high frequency information. It is used for computing the approximation coefficients and the process is called blurring.

The high pass filters are used for computing the detail coefficients. The high pass filters are fundamental tools for sharpening methods. An image is sharpened when the image contrast is enhanced between neighbouring areas with little variation in brightness or darkness [27, 32].



Figure 5.1.1: Simulation of a DWT of an image [28]

In every scale, the low and high pass filters are applied along both x and y axis. Thus, four sub-band images are generated and called as  $LL_1$ ,  $LH_1$ ,  $HL_1$ , and  $HH_1$ . The images  $LH_1$ ,  $HL_1$  and  $HH_1$  contain the horizontal, vertical, and diagonal information of the image after applying the high pass filter respectively.  $LL_1$  represents the approximation coefficients and contains the maximum information of the image obtained by the convolution of the signal with the low pass filter. These are described in a more detail:

$LL^{(i)}$ : The upper left quadrant consists of all coefficients, which were filtered by the analysis low pass filter along the rows and then filtered along the corresponding columns with the analysis low pass filter again. This sub-block is denoted by  $LL^{(i)}$  and represents the approximated version of the original at half the resolution.

$HL^{(i)}/LH^{(i)}$ : The lower left and the upper right blocks were filtered along the rows and columns with low pass filter and high pass filter, alternatively. The  $LH^{(i)}$  block contains vertical edges, mostly. In contrast, the  $HL^{(i)}$  blocks shows horizontal edges very clearly.

$HH^{(i)}$ : The lower right quadrant was derived analogously to the upper left quadrant but with the use of the analysis high pass filter which belongs to the given wavelet. It interprets and allocates edges of the original image in the diagonal direction.

In the 2-D Wavelet decomposition, a one-dimensional multiresolution analysis as defined for the 1-D wavelet decomposition is considered, where:

$$\{0\} \subset \cdots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \cdots \subset L^2(R)$$

and defines the spaces  $\mathbf{V}_j, j \in \mathbb{Z}$ , by

$$\mathbf{V}_j = V_j \otimes V_j = \{F(x, y) \mid F(x, y) = f(x)g(y), f, g \in V_j\}$$

The subspaces  $\mathbf{V}_j$  form a separable multiresolution analysis of  $L^2(\mathbb{R}^2)$ , with an increasing sequence of linear subspaces of  $L^2(\mathbb{R}^2)$ .

$$\{0\} \subset \cdots \subset \mathbf{V}_{-2} \subset \mathbf{V}_{-1} \subset \mathbf{V}_0 \subset \mathbf{V}_1 \subset \mathbf{V}_2 \subset \cdots \subset L^2(\mathbb{R}^2)$$

satisfying

- (i)  $\bigcap_{j \in \mathbb{Z}} \mathbf{V}_j = \{0\}, \overline{\bigcup_{j \in \mathbb{Z}} \mathbf{V}_j} = L^2(\mathbb{R}^2)$
- (ii)  $f(x, y) \in \mathbf{V}_j \iff f(2x, 2y) \in \mathbf{V}_{j+1}$
- (iii)  $f(x, y) \in \mathbf{V}_j \iff f(2^j m - x, 2^j n - y) \in \mathbf{V}_j$ , for all  $m, n \in \mathbb{Z}$

The scaling function associated with this  $L^2(\mathbb{R}^2)$  multiresolution analysis is then given by

$$\phi_j(x, y) = \phi(x)\phi(y) = \frac{1}{2^j} \varphi\left(\frac{2^j x - m}{2^j}\right) \varphi\left(\frac{2^j y - n}{2^j}\right) \quad (5.1.3)$$

where  $\phi(x)$  is the scaling function as opposed in the 1D wavelet. For each  $j \in \mathbb{Z}$ , the set  $\{\phi_{j,k}(x) = 2^{j/2} \phi(2^j x - k), k \in \mathbb{Z}\}$  is an orthonormal basis for  $V_j$ , hence it follows that the set

$$\phi_{j,m,n}(x, y) = \phi_{j,m}(x)\phi_{j,n}(y) = 2^j \phi(2^j x - m) \phi(2^j y - n), \quad m, n \in \mathbb{Z} \quad (5.1.4)$$

is an orthonormal basis for  $\mathbf{V}_j$ . For each  $j \in \mathbb{Z}$ , denoted by  $\mathbf{W}_j$ , the orthogonal complement of  $\mathbf{V}_j$  in  $\mathbf{V}_{j+1}$ . Then, we have the wavelet spaces  $\mathbf{W}_j$  given by:

$$\mathbf{W}_j = (W_j \otimes W_j) \oplus (V_j \otimes W_j) \oplus (W_j \otimes V_j), \quad j \in \mathbb{Z} \quad (5.1.5)$$

Hence four different filters are required in order to define the orthogonal complement of  $V_j$  in  $V_{j+1}$ :

$$\begin{aligned} LL^{(j)} &= \phi_{j_0,m,n}(x, y) = \phi_{j_0,m}(x)\phi_{j_0,n}(y) \\ LH^{(j)} &= \psi_{j_m,n}^H(x, y) = \psi_{j,m}(x)\phi_{j,n}(y) \\ HL^{(j)} &= \psi_{j_m,n}^V(x, y) = \phi_{j,m}(x)\psi_{j,n}(y) \\ HH^{(j)} &= \psi_{j_m,n}^D(x, y) = \psi_{j,m}(x)\psi_{j,n}(y) \end{aligned} \quad (5.1.6)$$

The corresponding wavelet function of the horizontal, vertical and diagonal representation of the images are as follows:

1. Horizontal sub-band representation

$$W_{\psi}^H(j, m, n) = \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(x, y) \psi_{j,m,n}^H(x, y)$$

2. Vertical sub-band representation

$$W_{\psi}^V(j, m, n) = \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(x, y) \psi_{j,m,n}^V(x, y)$$

3. Diagonal sub-band representation

$$W_{\psi}^D(j, m, n) = \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(x, y) \psi_{j,m,n}^D(x, y)$$

On the other hand, any image can be reconstructed and for that purpose the Inverse Wavelet Transform is used which is the transpose of the forward transform matrix and is given by :

$$\begin{aligned} f(x, y) &= \frac{1}{\sqrt{MN}} \sum_m \sum_n W_{\phi}(j_0, m, n) \phi_{j_0, m, n}(x, y) \\ &+ \frac{1}{\sqrt{MN}} \sum_{I=H,V,D} \sum_{j=j_0}^{\infty} \sum_m \sum_n W_{\psi}^I(j, m, n) \psi_{j,m,n}^I(x, y) \end{aligned} \quad (5.1.7)$$

The 2-D wavelet decomposition can be represented in matrix multiplication form as in the matrix multiplication form of the 1-D wavelet decomposition, with the difference that the 1-D wavelet is applied in all image columns and rows.

For the purpose of this application, V and W define two vector spaces of functions defined on the intervals  $[0, M)$  and  $[0, N)$ , respectively, and suppose that  $f_1 \in V$  and  $f_2 \in W$ . The tensor product of  $f_1$  and  $f_2$ , denoted  $f_1 \otimes f_2$ , denotes the function in two variables defined on  $[0, M) \times [0, N)$  given by  $f_1(x_1) f_2(x_2)$ .

From equation 5.1.5, if the spaces  $V_j$  and  $W_j$  can be used to approximate functions in one variable as stated above, then  $V_j \otimes W_j$  can be used to approximate functions in two variables space.

Given the equation 5.1.3 and 5.1.5:

$$\phi_j \otimes \phi_j = \{\phi_{j,m,n} \otimes \phi_{j,m,n}\}$$

and

$$\begin{aligned} & (\phi_{j-1} \oplus \psi_{j-1}) \otimes (\phi_{j-1} \oplus \psi_{j-1}) \\ &= \{\phi_{j-1,m,n} \otimes \phi_{j-1,m,n} \\ & \quad \phi_{j-1,m,n} \otimes \psi_{j-1,m,n} \\ & \quad \psi_{j-1,m,n} \otimes \phi_{j-1,m,n} \\ & \quad \psi_{j-1,m,n} \otimes \psi_{j-1,m,n}\} \end{aligned}$$

The space  $\psi_{j,m,n}^H$  defined in equation 5.1.6 is spanned by  $\{\phi_{j,m} \otimes \psi_{j,n}\}_{m,n}$ , which is called LH<sup>j</sup>-sub-band.

The space  $\psi_{j,m,n}^V$  spanned by  $\{\psi_{j,m} \otimes \phi_{j,n}\}_{m,n}$ , which is called the HL<sup>j</sup>-sub-band.

The space  $\psi_{j,m,n}^D$  spanned by  $\{\psi_{j,m} \otimes \psi_{j,n}\}_{m,n}$ , which is called the HH<sup>j</sup>-sub-band.

Hence, the equations above in each level of wavelet reconstruction can be summarized as :

$$\begin{aligned} V_j \otimes V_j &= (V_{j+1} \otimes V_{j+1}) \oplus W_{j+1}^H \oplus W_{j+1}^V \oplus W_{j+1}^D = \\ & \sum_{m,n} W_{\phi,j+1,m,n} \phi_{j+1,m,n}(x,y) + \sum_{m,n} W_{\psi,j+1,m,n}^H \psi_{j+1,m,n}^H(x,y) + \\ & \sum_{m,n} W_{\psi,j+1,m,n}^V \psi_{j+1,m,n}^V(x,y) + \sum_{m,n} W_{\psi,j+1,m,n}^D \psi_{j+1,m,n}^D(x,y) \end{aligned} \quad (5.1.8)$$

The coefficients  $W_{j,m,n}^H$ ,  $W_{j,m,n}^V$ , and  $W_{j,m,n}^D$  are as mentioned the horizontal, vertical, and diagonal detail coefficients respectively, and I=H,V,D. The first level detail coefficients are computed by the inner product of the given image and the corresponding wavelet function  $\psi_{j+1,m,n}^I$ .

$$W_{\psi,j,m,n}^I = \langle f, \psi_{j+1,m,n}^I \rangle = \sum f(x,y) \psi_{j+1,m,n}^I(x,y) \quad (5.1.9)$$

Every next scale j, the wavelet coefficients are derived by the inner product of wavelet function of the previous level of decomposition and the approximation coefficients derived in the previous step.

$$W_{\psi,j,m,n}^I = \langle W_{\phi,j+1,m,n}, \psi_{j+1,m,n}^I \rangle = \sum W_{\phi,j+1,m,n} \psi_{j+1,m,n}^I(x,y) \quad (5.1.10)$$

Each space vector  $V_j$  is spanned on a interval of the form  $[k2^{-j}, (k+1)2^{-j})$ ,  $V_j \otimes V_j$  is the vector space of functions in two variables which are constant on any square of the form  $[m2^{-j}, (m+1)2^{-j}) \times [n2^{-j}, (n+1)2^{-j})$ .

This relationship can be further expanded recursively for all  $j$  scales:

$$V_j = V_{j-1} \oplus W_{j-1} = V_{j-2} \oplus W_{j-2} \oplus W_{j-1} = \dots = V_0 \oplus \bigoplus_{l=0}^{j-1} \mathbf{W}_l \quad (5.1.11)$$

Hence, from the Inverse Wavelet Decomposition equation given in 5.1.7 based on equation 5.1.4, 5.1.9 and 5.1.10 is given by:

$$\begin{aligned} f(x, y) &= \sum_{m,n} W_{\phi,j_0,m,n} \phi_{j_0,m}(x) \phi_{j_0,n}(y)^T + \sum_{m,n} W_{\psi,j_0,m,n}^H \psi_{j_0,m}(x) \phi_{j_0,n}(y)^T \\ &+ \sum_{m,n} W_{\psi,j_0,m,n}^V \psi_{j_0,m}(x) \psi_{j_0,n}(y)^T + \sum_{m,n} W_{\psi,j_0,m,n}^D \phi_{j_0,m}(x) \psi_{j_0,n}(y)^T + \\ &+ \sum_{m,n} W_{\psi,j_0+1,m,n}^H \psi_{j_0+1,m}(x) \phi_{j_0+1,n}(y)^T + \sum_{m,n} W_{\psi,j_0+1,m,n}^V \psi_{j_0+1,m}(x) \phi_{j_0+1,n}(y)^T \\ &+ \dots \end{aligned} \quad (5.1.12)$$

Rephrasing equation 5.1.12 in matrix multiplication form:

$$\begin{aligned} F &= \Phi_{j_0,M} W_{\Phi,j_0} \Phi_{j_0,N}^T + \Psi_{j_0,M} W_{\Psi,j_0}^H \Phi_{j_0,N}^T + \Psi_{j_0,M} W_{\Psi,j_0}^V \Psi_{j_0,N}^T \\ &\quad + \Phi_{j_0,M} W_{\Psi,j_0}^D \Psi_{j_0,N}^T + \Psi_{j_0+1,M} W_{\Psi,j_0+1}^H \Phi_{j_0+1,N}^T \\ &\quad + \Psi_{j_0+1,M} W_{\Psi,j_0+1}^V \Psi_{j_0+1,N}^T + \Phi_{j_0+1,M} W_{\Psi,j_0+1}^D \Psi_{j_0+1,N}^T + \dots \end{aligned} \quad (5.1.13)$$

In other words, any function can be considered as a combination of infinitely many approximations of different levels of details.

The equation 5.1.13 is of main attention and is used in the Wavelet DMD algorithm, where the input image data are transformed in single column form. Hence, it is required that all data to be in vector form and this is applied by the introduction of Kronecker product. The Kronecker equation of the scaling function will be introduced, where the wavelet functions for I=H,V,D are following the same pattern. Based on the equations 5.1.13 and 5.1.11 :

$$a_{\Phi_{j_0}} = \text{vec}(\Phi_{j_0,M} W_{\Phi,j_0} \Phi_{j_0,N}^T) = (\Phi_{j_0,N} \otimes \Phi_{j_0,M}) \text{vec}(W_{\Phi,j_0}) \quad (5.1.14)$$

$$a_{\Psi_{j_0}^H} = \text{vec}(\Psi_{j_0,M} W_{\Psi,j_0}^H \Phi_{j_0,N}^T) = (\Phi_{j_0,N} \otimes \Psi_{j_0,M}) \text{vec}(W_{\Psi,j_0}^H) \quad (5.1.15)$$



$$a_k = \begin{bmatrix} \text{vec}(W_{\Phi, j_0}^k) \\ \text{vec}(W_{\Psi, j_0}^H) \\ \text{vec}(W_{\Psi, j_0}^V) \\ \text{vec}(W_{\Psi, j_0}^D) \\ \vdots \end{bmatrix} \quad (5.1.21)$$

which is summarized :

$$\text{vec}(F) = \Phi_{tot} a_k \quad (5.1.22)$$

Any block of the  $\Phi_{tot}$  matrix has the following properties as described in chapter 2 for the one dimensional wavelets:

$$\begin{aligned} & (\Phi_{j_0, N} \otimes \Phi_{j_0, M})(\Phi_{j_0, N} \otimes \Phi_{j_0, M})^T = \\ & (\Phi_{j_0, N} \Phi_{j_0, N}^T) \otimes (\Phi_{j_0, M} \Phi_{j_0, M}^T) = \begin{cases} I \otimes I = I, m = n \\ 0, \text{ otherwise} \end{cases} \end{aligned} \quad (5.1.23)$$

$$\begin{aligned} & (\Phi_{j_0, N} \otimes \Psi_{j_0, M})(\Phi_{j_0, N} \otimes \Psi_{j_0, M})^T = \\ & (\Phi_{j_0, N} \Phi_{j_0, N}^T) \otimes (\Psi_{j_0, M} \Psi_{j_0, M}^T) = \begin{cases} I \otimes I = I, m = n \\ 0, \text{ otherwise} \end{cases} \end{aligned} \quad (5.1.24)$$

The equations 5.1.12, 5.1.20, 5.2.1 and 5.1.22 will be the reference equations for the theorem derivation of the novel wavelet DMD algorithm.

## 5.2 Wavelet Decomposition DMD

The principal idea of introducing the wavelets in the dynamic mode decomposition, is to reveal hidden structures that emerge in higher order complex systems. It aims at revealing the multiscale nature of the input data by means of deriving modes that separate the mixed frequency dynamic effects where other DMD algorithms are unable to identify. It will be shown that the use of wavelets, will force the dynamic mode decomposition to identify and separate the multiscale features using the minimum possible amount of dominant modes. Therefore, the computational efficiency will be increased.

A novel alternative approach called Wavelet Decomposition DMD is derived and analysed in this section. In the proposed algorithm, the wavelet

decomposition is applied over the input images, creating  $j$  scales of approximation and detail image coefficients. The first alteration to the standard wavelet reconstruction steps is that the detail coefficients in each level are merged together instead of taking every sub-band image  $LH_j, HL_j$  and  $HH_j$  separately. Hence, the reconstruction of images that include information from all the high frequency coefficients is accomplished in each level.

For instance, the wavelet reconstruction from the detail coefficients in level  $j$  is given by:

$$W_{\psi_j^{HVD}}(j, m, n) = [W_{\psi_j^H}(j, m, n) \quad W_{\psi_j^V}(j, m, n) \quad W_{\psi_j^D}(j, m, n)] \quad (5.2.1)$$

The inverse wavelet transform algorithm reconstructs images based only on the selected coefficients. Therefore, images are reconstructed by the approximation coefficients  $W_{\phi_{j_0}}(j_0, m, n)$  and the merged coefficients  $W_{\psi_j^{HVD}}(j, m, n)$  separately instead of the complete reconstruction from all coefficients as the original Inverse Wavelet transform claims. The ultimate goal is to create images that include only specific patterns in each scale of resolution, which are fed into the DMD algorithm separately. Due to this separation, the derived DMD models reveal hidden structures in specific areas that may emerge, which cannot be obtained from the existing reduced order models. Moreover, based on this novel method, the DMD is applied over smaller amount of datasets, hence the computation cost is also reduced.

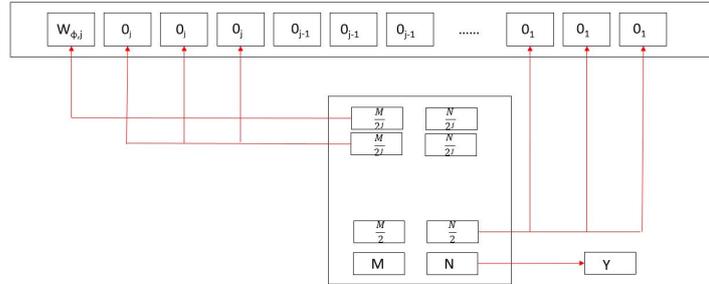


Figure 5.2.1: Example of a reconstructed Image based only on the Approximation Coefficients of the coarsest scale

The wavelet-based DMD will be analysed in order to verify if it is computationally more efficient than the other DMD methods such as Hankel DMD in terms of the number of modes needed to reveal the system behaviour. The aim is to evaluate if Wavelet-DMD is capable of revealing and also separating multiscale features that the standard DMD methods are incapable of.

The algorithm is summarized in the following steps.

---

**Algorithm 3** Wavelet DMD Algorithm
 

---

1. **Initialization:** Input data are inserted in the Wavelet Transform algorithm.

2. **Decomposition:** The data are decomposed in  $j$  scales based on the selected wavelet function according to the application needs.  $j + 1$  datasets are created based on the equations for the approximation and detail coefficients.

$$W_{\psi,j,m,n}^I = \Psi_{j,M} W_{\Psi,j}^H \Phi_{j,N}^T + \Psi_{j,M} W_{\Psi,j}^V \Psi_{j,N}^T + \Phi_{j,M} W_{\Psi,j}^D \Psi_{j,N}^T$$

$$W_{\phi,j_0,m,n} = \Phi_{j_0,M} W_{\Phi,j_0} \Phi_{j_0,N}^T$$

3. **Reconstruction:** Each of the  $W_{\psi,j,m,n}^I$  and  $W_{\phi,j_0,m,n}$  are reconstructed separately to create images that contain low and high frequency dynamics separately. As a pre-step, the details coefficient in each level are merged as shown in equation 5.2.1.  $j + 1$  image sets are reconstructed based on the equations:

$$a_{\Phi_{j_0}} = \Phi_{j_0,M} W_{\Phi,j_0} \Phi_{j_0,N}^T$$

$$a_{\Psi_{j_0}} = \Psi_{j_0,M} W_{\Psi,j_0}^H \Phi_{j_0,N}^T + \Psi_{j_0,M} W_{\Psi,j_0}^V \Psi_{j_0,N}^T + \Phi_{j_0,M} W_{\Psi,j_0}^D \Psi_{j_0,N}^T$$

and for every remaining  $j$  scale :

$$a_{\Psi_j} = \Psi_{j,M} W_{\Psi,j}^H \Phi_{j,N}^T + \Psi_{j,M} W_{\Psi,j}^V \Psi_{j,N}^T + \Phi_{j,M} W_{\Psi,j}^D \Psi_{j,N}^T$$

In order to preserve the size of initial images, for instance, when the image is reconstructed from the approximation coefficients, the remaining coefficients are set to zero (see Fig.5.2.1).

4. **Modes extraction:** DMD is applied separately to the different image sets. The amount of selected modes is equal among the different models.  $j + 1$  state space models are created based on equations (4.2.2 - 4.2.10).

---

### 5.2.1 Analysis of the similarity between Hankel-DMD & Wavelet DMD

The purpose of the proposed method is to substitute the standard DMD and help the user understand the system dynamics in a better way compared to the existing models derived by the DMD algorithms. Hence, it is important to find the connection between each scale DMD model after the wavelet decomposition and the DMD model derived from the standard DMD.

The following proof will demonstrate that the Wavelet-DMD and standard-DMD method share the same eigenvalues, hence the derived DMD modes are not changing. Therefore, the Wavelet-DMD can be used in any stage without the loss or alteration of information.

**Theorem 1** *Let  $A$  be the estimated dynamic matrix of the standard DMD algorithm and let  $B$  be the estimated dynamic matrix of the wavelet-DMD algorithm. For every given wavelet basis, the matrix  $A$  is a linear transformation of the matrix  $B$ ,  $A = \Phi_{tot} B \Phi_{tot}^{-1}$ . Therefore, the standard DMD and its extensions share the same eigenvalues with the Wavelet-DMD, meaning that the Wavelet-DMD is consistent with the standard DMD approaches, yet revealing additional characteristics.*

#### Proof :

The dynamic mode decomposition is an equation-free data driven model that approximates the best linear dynamic model of a given data set  $\mathbf{X}$ . It is known that a linear system derived by the DMD is expressed by :

$$\mathbf{X}' = A\mathbf{X} \quad (5.2.2)$$

where,  $\mathbf{X}$  contains the snapshots  $X_k$  in vectorized form. Each of the snapshots  $\mathbf{X}$  are derived from the vectorization of the  $F$  matrix in equation 5.1.22, meaning that  $X_k = \text{vec}(F)$ .

The idea of Wavelet-DMD is to estimate a set of linear systems, estimating one dynamical model per scale  $j$  for each of the approximation and merged detail coefficients, that can substitutes the dynamic model of equation 5.2.2.

$$a' = Ba \quad (5.2.3)$$

The vectorization of equations 5.2.2 and 5.2.3 for a single snapshot is defined as :

$$X_{k+1} = AX_k \quad (5.2.4)$$

$$a_{k+1} = Ba_k \quad (5.2.5)$$

By multiplying both legs of the equation (5.2.5) by  $\Phi_{tot}$ , it is given that:

$$\Phi_{tot}a_{k+1} = \Phi_{tot}Ba_k \quad (5.2.6)$$

The first leg of the equation 5.2.6, based on equations 5.1.20 and 5.1.21 is equal to:

$$X_{k+1} = \Phi_{tot}a_{k+1} \quad (5.2.7)$$

Hence, the first leg of equation (5.2.6) and the second leg of equation (5.2.7) are the same, thus:

$$X_{k+1} = \Phi_{tot}Ba_k \quad (5.2.8)$$

Moreover, combining equation 5.2.4 and equation 5.2.7, one gets:

$$X_{k+1} = A\Phi_{tot}a_k \quad (5.2.9)$$

Finally, the combination of the equations 5.2.8 and 5.2.9, derives that :

$$\begin{aligned} \Phi_{tot}Ba_k &= A\Phi_{tot}a_k \\ \Phi_{tot}B &= A\Phi_{tot} \\ B &= \Phi_{tot}^{-1}A\Phi_{tot} \end{aligned} \quad (5.2.10)$$

or

$$A = \Phi_{tot}B\Phi_{tot}^{-1} \quad (5.2.11)$$

The equation 5.2.11 indicates that the state matrix of the standard DMD is a linear transformation of the state matrix of the Wavelet-DMD. It is

proven that the new system based on the wavelet transform and the initial dynamic system have the same eigenvalues.

It is known that the eigenvalues are the roots of the characteristic polynomial. Hence, the characteristic polynomial is given as:

$$\chi_A(\lambda) = \det(A - \lambda I) = \det(\Phi_{tot} B \Phi_{tot}^{-1} - \lambda I)$$

The argument of the determinant above is equal to :

$$\begin{aligned} \Phi_{tot} B \Phi_{tot}^{-1} - \lambda I &= \\ \Phi_{tot} B \Phi_{tot}^{-1} - \lambda \Phi_{tot} I \Phi_{tot}^{-1} &= \\ \Phi_{tot} (B - \lambda I) \Phi_{tot}^{-1} & \end{aligned} \quad (5.2.12)$$

Thus, the characteristic polynomial can be written as:

$$\chi_A(\lambda) = \det(\Phi_{tot} (B - \lambda I) \Phi_{tot}^{-1}) \quad (5.2.13)$$

Based on the orthonormality property proven in section 2.3.2

$$\begin{aligned} \chi_A(\lambda) &= \det(B - \lambda I) \\ &= \chi_B(\lambda) \end{aligned} \quad (5.2.14)$$

Since similar matrices A and B have the same characteristic polynomial, they also have the same eigenvalues. Based on that, the eigenvector relationship between the two matrices is given by:

If  $A = \Phi_{tot} B \Phi_{tot}^{-1}$  and  $v \neq 0$  is an eigenvector of B (say  $Bv = \lambda v$ ) then consider:

$$\begin{aligned} A(\Phi_{tot} v) &= \Phi_{tot} B \Phi_{tot}^{-1} (\Phi_{tot} v) \\ &= \Phi_{tot} B (\Phi_{tot}^{-1} \Phi_{tot} v) \\ &= \Phi_{tot} B v \\ &= \lambda (\Phi_{tot} v) \end{aligned} \quad (5.2.15)$$

Thus  $\Phi_{tot} v$  (which is non-zero since  $\Phi_{tot}$  is invertible) is an eigenvector for B with eigenvalue  $\lambda$ . Similarly, if  $u$  is an eigenvector for B then  $\Phi_{tot}^{-1} u$  is an eigenvector for A.

It can also be shown that it can be related to the inverse part. If  $A = \Phi_{tot}B\Phi_{tot}^{-1}$  and as proven above  $\lambda_A = \lambda_B = \lambda$ , then :

Suppose that the eigenvectors of A is a transformation of the eigenvectors of B, meaning that  $v_a = \Phi v_b$ .

$$\begin{aligned}
 Av_a &= A\Phi_{tot}v_b \\
 Av_a &= \Phi_{tot}Bv_b \\
 Av_a &= \Phi_{tot}\lambda v_b \\
 Av_a &= \lambda\Phi_{tot}v_b \\
 Av_a &= \lambda v_a
 \end{aligned} \tag{5.2.16}$$

In conclusion, this proof demonstrates that the temporal modes remain the same in contrast to the spatial modes that differ between the two approaches where the modes of the standard DMD are actually a transformation of the Wavelet-DMD modes. Hence, the two methods display the same behaviour and evolution, leading to the fact that the Wavelet-DMD can be chosen over the Hankel-DMD when needed.

Moreover, it is proven that the spatial modes of the one approach based on the eigenvalues in every model, is a transformation of the second based on the mother wavelet. This indicates that the spatial modes among the different scales are able to obtain features of the initial images in different sampling frequencies, revealing hidden structures.

In the following examples, it will be shown that the Wavelet-DMD method outperforms the standard Hankel DMD and multiscale POD and DMD in terms of revealing hidden structures and separating the mixed frequency effects of the given system. The spatial modes of the wavelet DMD methods will be able provide full insight of the different dynamics of any given system, making this algorithm a powerful tool for complex systems.

### 5.3 Wavelet-DMD Examples

Following the proof that was derived in the previous chapter, the novel multiscale DMD method based on the modified Wavelet transform will be applied into two different examples and its performance will be evaluated and compared with all existing single scale and multiscale DMD methods. Key feature will be to investigate the ability of the new method to reveal hidden structures and behaviour of the system that cannot be revealed while using

the existing methods. In this section, the key results and comparisons will be demonstrated, followed by analytical simulations in Chapter 6.

For the purpose of this project, the chosen wavelet basis is based on the family of Symlet wavelets. The Symlet wavelets are an extension of the Daubechies wavelet family which are orthogonal and have compact support abilities for feature analysis. There are smoother than the simple Haar wavelets and are more capable at identifying abrupt changes [27]. The Symlet wavelets are nearly symmetrical wavelets in contrast to the Daubechies and they are preferred for the purpose of this project [29].

### 5.3.1 Wave Diffusion Example

The first example where the performance of the DMD methods will be compared is on the wave diffusion model. Two independent waves oscillating in different frequency and amplitude will be created and combined into one dynamic model, simulating an oil and a water drop. The frequency ratio and amplitude of the two drops will be such that a model with multiscale characteristics will be obtained.

The general equation of the wave diffusion is described as:

$$\frac{\partial \phi(\mathbf{r}, t)}{\partial t} = \nabla \cdot [D(\phi, \mathbf{r}) \nabla \phi(\mathbf{r}, t)] \quad (5.3.1)$$

where  $\phi(r, t)$  is the density of the diffusing material at location  $\mathbf{r}$  and time  $t$  and  $D(\phi, \mathbf{r})$  is the collective diffusion coefficient for density  $\phi$  at location  $\mathbf{r}$ ; and  $\nabla$  represents the vector differential operator.

The parameter  $D$  is constant, in order for the equation to reduce to the following linear differential equation:

$$\frac{\partial \phi(\mathbf{r}, t)}{\partial t} = D \nabla^2 \phi(\mathbf{r}, t) \quad (5.3.2)$$

$$\frac{\partial^2 \delta}{\partial t^2} = c^2 \frac{\partial^2 \delta}{\partial x^2} + c^2 \frac{\partial^2 \delta}{\partial y^2} + f$$

which can be discretized as

$$\frac{\delta_{ij}^{(n+1)} - 2\delta_{ij}^{(n)} + \delta_{ij}^{(n-1)}}{\Delta t^2} = c^2 \frac{\delta_{i+1,j}^{(n)} - 2\delta_{ij}^{(n)} + \delta_{i-1,j}^{(n)}}{\Delta x^2} + c^2 \frac{\delta_{i,j+1}^{(n)} - 2\delta_{ij}^{(n)} + \delta_{i,j-1}^{(n)}}{\Delta y^2} + f_i^{(n)} \quad (5.3.3)$$

or letting  $\Delta x = \Delta y$

$$\delta_{ij}^{(n+1)} = 2\delta_{ij}^{(n)} - \delta_{ij}^{(n-1)} + \left[ c \frac{\Delta t}{\Delta \tau} \right]^2 \left( \delta_{i+1,j}^{(n)} + \delta_{i-1,j}^{(n)} - 4\delta_{ij}^{(n)} + \delta_{ij+1}^{(n)} + \delta_{ij-1}^{(n)} \right) + \Delta t^2 f_i^{(n)} \quad (5.3.4)$$

The purpose of this example is to compare the new wavelet DMD algorithm with the existing DMD methods over a system with multiscale characteristics. The dominant modes and the estimated dynamic model in all cases will be derived and a straight comparison of the mode characteristics will be performed among all different algorithms.

The wavelet-DMD will derive  $(j+1)$  different dynamic models, one for each of the merged detail coefficients and one for the approximation coefficients, where  $j$  is the number of scales. For the sake of a fair comparison between the two methods, the amount of dominant modes per wavelet DMD dynamic model will be  $(j+1)$  times smaller than the standard DMD methods. Hence, the total amount of modes between the two methods will be the same. Following the SVD plot, the minimum number of dominant modes  $k$  is selected and will be used in this example.

As discussed above, in order to determine a model with multiscale characteristics, the wave that simulates the water drop amplitude and frequency, is 4 and 10 times greater than the oil drop parameters respectively. The figures below simulate the two waves in a total of 150 time-steps, hence the DMD will be performed over a total of 150 input images. The simulations below, demonstrate the 2 wave system at different time steps.

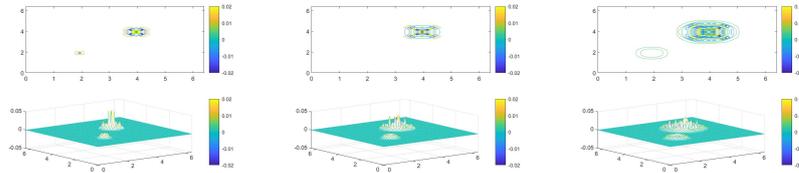
From the images above it can be seen that two different waves are simulated, where the amplitude of the fast moving one is noticeably greater than the second wave and after the 50<sup>th</sup> time-step, the two waves are merged. Additionally, the frequency spectrum of the fast moving wave is much greater than the slow one, so the system can be entitled as a multiscale system.

The following reduced order approaches will be compared and the Wavelet methods advantages will be addressed over the following methods :

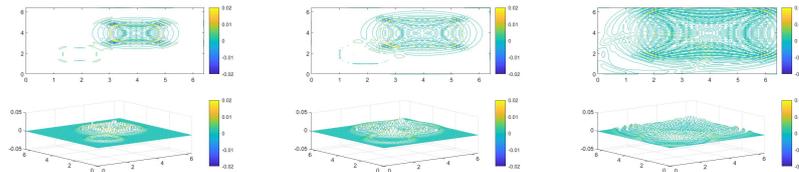
- Proper Orthogonal Decomposition POD
- Dynamic Mode Decomposition DMD ( Hankel Form )
- Wavelet Proper Orthogonal Decomposition W-POD

### 5.3.1.1 Hankel - DMD

The simulation will start by applying the Hankel-DMD algorithm over the simulated waves. A full analysis of the eigendecomposition and DMD spatial



(a) Wave Diffusion Simulation of oil and water drop in step  $k=1$       (b) Wave Diffusion Simulation of oil and water drop in step  $k=10$       (c) Wave Diffusion Simulation of oil and water drop in step  $k=20$



(d) Wave Diffusion Simulation of oil and water drop in step  $k=40$       (e) Wave Diffusion Simulation of oil and water drop in step  $k=60$       (f) Wave Diffusion Simulation of oil and water drop in step  $k=100$

modes will be presented addressing the performance of the Hankel-DMD algorithm. Purpose of this example, is to investigate if the DMD can separate and reveal in discrete modes, the fast and slow moving dynamics and additionally to investigate the total amount of modes needed in order to reveal all dynamics separately.

The example will start by plotting the singular values in the SVD step in order for the amount of dominant modes to be determined. A total of 48 dominant modes is selected for this example out of the total 150 modes. The following figures present the computed DMD modes after applying the DMD algorithm.

It is shown that for the first 39 modes, the DMD cannot separate the slow and fast dynamics and the system is dominated by high frequency effects. The DMD performance can majorly be affected in cases where the user selects a smaller amount of modes in the SVD step, leading into a model that is incapable of revealing both slow and fast dynamics. Following the images below, the most dominant modes ( see mode 1 and mode 3), are showing only high frequency or mixed data of both waves. Hence, it is not clear where the different dynamics are located.

The slow dynamics (second wave) are picked up only past the 40<sup>th</sup> dominant mode. Figure 5.5 shows Modes 40 and 42 where only the slow moving wave is shown. The same pattern continues for the remaining modes, where only the low frequency dynamics are shown.

Although the DMD manages to separate the slow and fast dynamics in

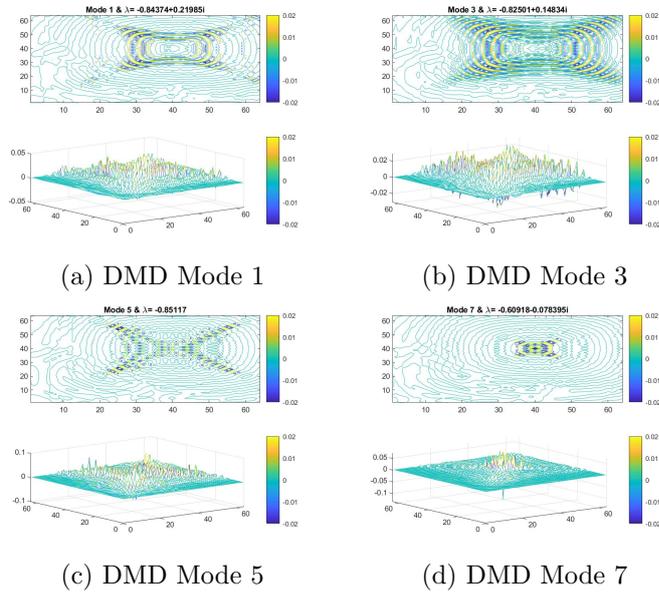


Figure 5.3.1: Presentation of the first 7 DMD mode after applying the Hankel-2 DMD

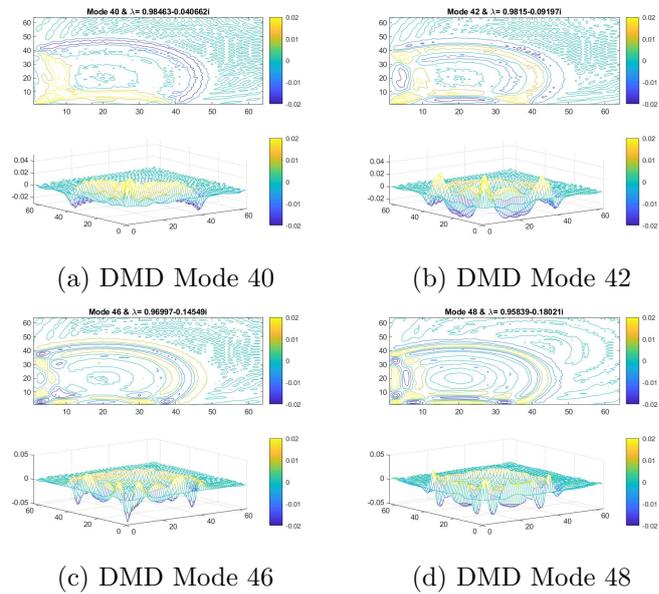


Figure 5.3.2: Presentation of the DMD modes of the low frequency effects after applying the Hankel-2 DMD

this example, the number of modes needed in order to reveal the slow dynamics is appeared after the 40th mode. The high number of modes required

in order to reveal the various system dynamics is causing an increase in the computational cost. Moreover, it can mislead the user about the system behaviour in cases where a smaller amount of modes is selected.

The introduction of the Wavelet-DMD will try to address these problems in the following section.

### 5.3.1.2 Wavelet Decomposition DMD

The Hankel-DMD manages to separate the slow and fast dynamics but the amount of modes needed for this purpose is high ( approximately 30% of total data required ), putting a toll in the computational cost and time. Moreover, it is not clear if the dominant modes contain mixed or only high frequency dynamics. The aim of the wavelet decomposition DMD is to minimize these issues by minimizing the amount of modes needed in order for the DMD to separate the different system dynamics and separate all mixed dynamics successfully.

For the sake of this example, three (3) scales of decomposition are selected. Hence, based on the proposed theory, the Wavelet DMD will create 4 different dynamic models, one for each scale of the merged detail coefficients and one for the approximation coefficients. Thus, 12 modes per scale are selected for a fair comparison with the previous DMD algorithm. The Wavelet Decomposition is applied over the input data. The reconstructed images per scale are given as an input on the DMD algorithm separately and the DMD derives a dynamic model for each one of them.

After determining the DMD modes, the first two images below demonstrate the two most dominant modes of the DMD in the first scale of decomposition.

The second mode is not presented due to being the same as the first mode with its eigenvalue being the complex conjugate of the eigenvalue of the first mode. These modes contain information of the reconstructed images based on the detail coefficients in scale one. By observing the remaining modes, it is shown that only the high frequency system dynamics are revealed indicating that the system is governed by high frequency dynamics. The key point is to investigate if the wavelet decomposition manages to separate the slow and fast system dynamics successfully and faster than the conventional DMD.

Applying the DMD method over the reconstructed images on the second scale of decomposition, it is shown that after the 5th dominant mode, the slow frequency dynamics are revealed and are separated from the fast dynamics.

By observing the dynamic model on the second scale, the user can identify and locate the different system dynamics. The mode No5 reveals only dynamics from the slow moving wave. The high frequency dynamics are

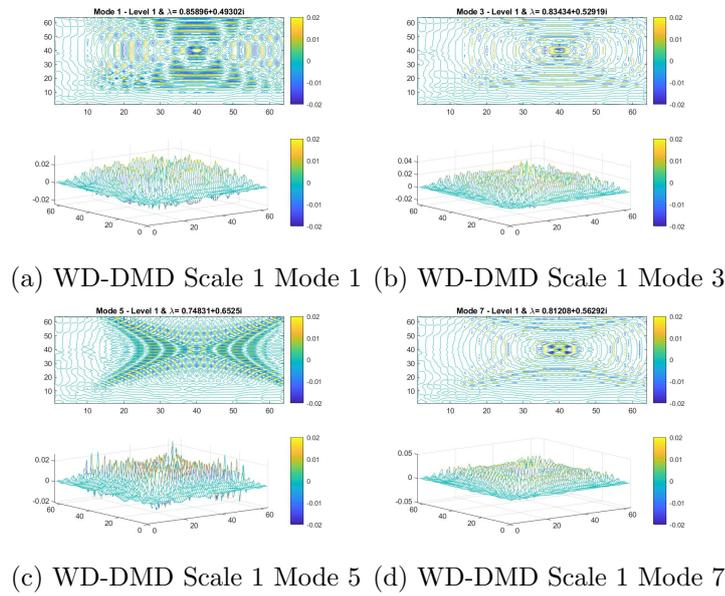


Figure 5.3.3: Presentation of the first 7 DMD mode in the first scale of decomposition using the Symlet mother wavelet, revealing only the fast dynamics

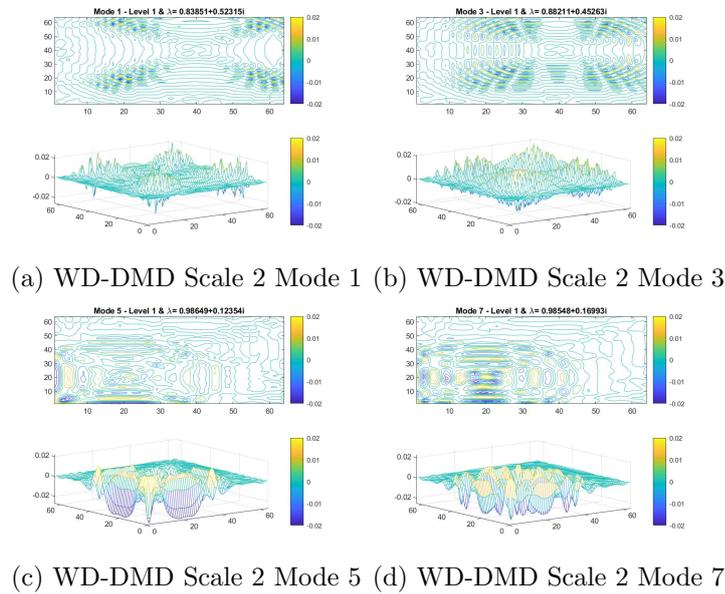


Figure 5.3.4: Presentation of the first 7 DMD mode in the second scale of decomposition using the Symlet mother wavelet, revealing the slow dynamics

showed only the first 4 modes. Hence, the use of the wavelets manages to separate the mixed dynamics successfully.

The beneficial properties of the pre-application of the wavelet decomposition are even more obvious in the third scale of decomposition, where only the slow dynamics are shown, even from the first modes.

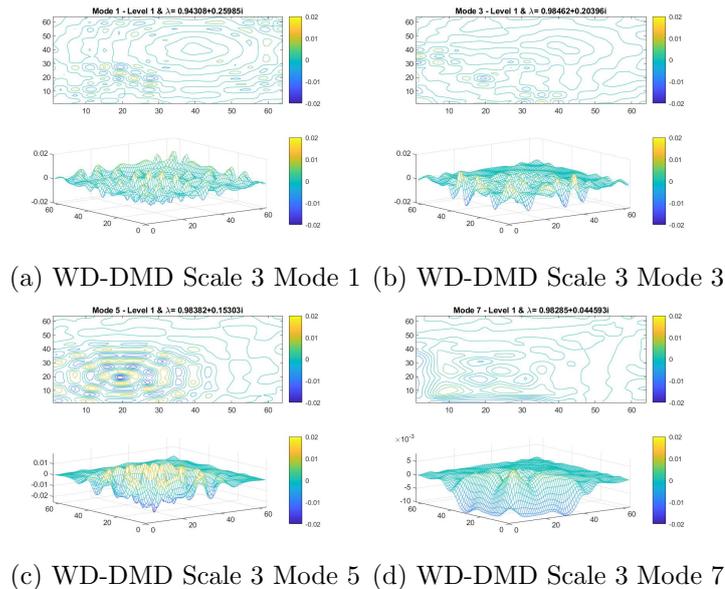


Figure 5.3.5: Presentation of the first 7 DMD mode in the third scale of decomposition using the Symlet mother wavelet, revealing immediately the slow dynamics of these modes

The first mode includes some fast moving dynamics that are based on the fast moving wave but the first mode reveals dynamics only in the position where the slow moving wave is located. The phenomenon is even more clear in the next modes. The wavelet decomposition DMD manages to separate successfully the slow and fast dynamics of a given multiscale system. Moreover, the derived method manages to use the minimum amount of dominant modes in order to reveal the different system dynamics increasing the accuracy and minimizing the computational cost in contrast to the Hankel-DMD method.

Depending on the application purposes, the user can use wavelets in deeper scales. For instance, by applying the Wavelet DMD in deeper scales for this example, the beneficial properties of the wavelet decomposition are even more realistic.

Comparing the modes of the third and fourth scale, it can be seen that the fourth scale contains low frequency dynamics that are completely separated

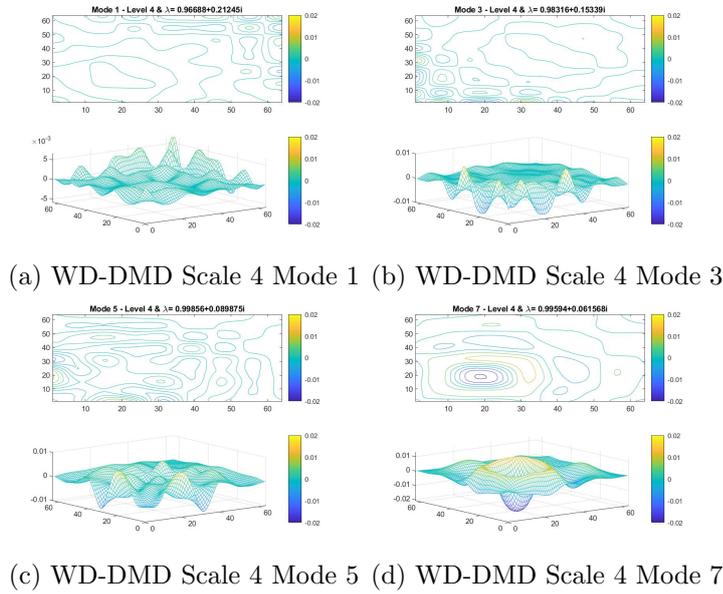


Figure 5.3.6: Presentation of the first 7 DMD mode in the fourth scale of decomposition using the Symlet mother wavelet, revealing immediately the slow dynamics of these modes

from the high frequency ones.

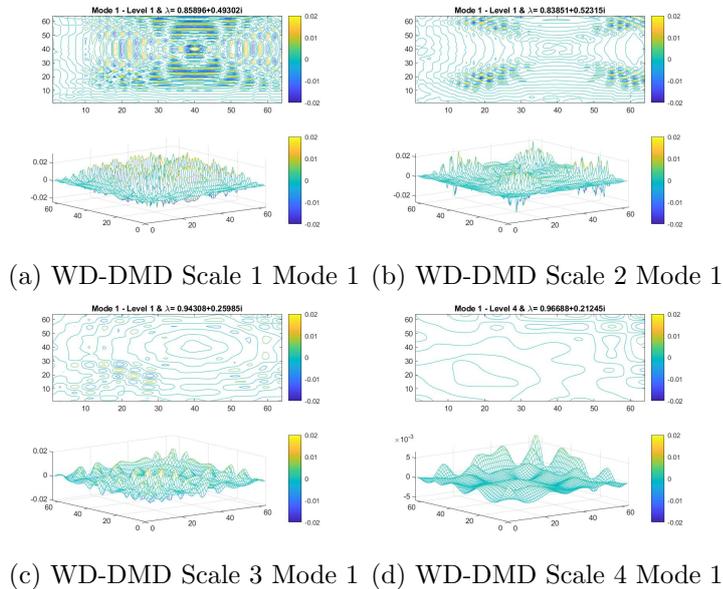


Figure 5.3.7: Conclusive presentation of the first DMD mode at the different scales of resolution

In conclusion, the wavelet decomposition manages to separate the high frequency and low frequency dynamics in each mode successfully. By observing each mode separately, in the first scale of decomposition, the modes are governed by information with high frequency characteristics, where when a deeper level of decomposition is achieved, the same modes are governed by the slow frequency dynamics. Hence, the user can identify and reveal all system patterns at all times.

### 5.3.1.3 Proper Orthogonal Decomposition POD

As described in the previous chapter, POD is designed to decompose the vectorized data into a set of deterministic spatial modes, that give an insight about the fluctuation and the time evolution of the given data, followed by their time coefficients showing each mode time evolution.

The following figures illustrate the spatial modes and their corresponding time evolution after applying the POD algorithm over the input images. As described in the previous chapter, in order for the fair comparison with the identification methods presented above, the selection of the number of dominant modes are exactly the same as the ones selected for the Hankel DMD.

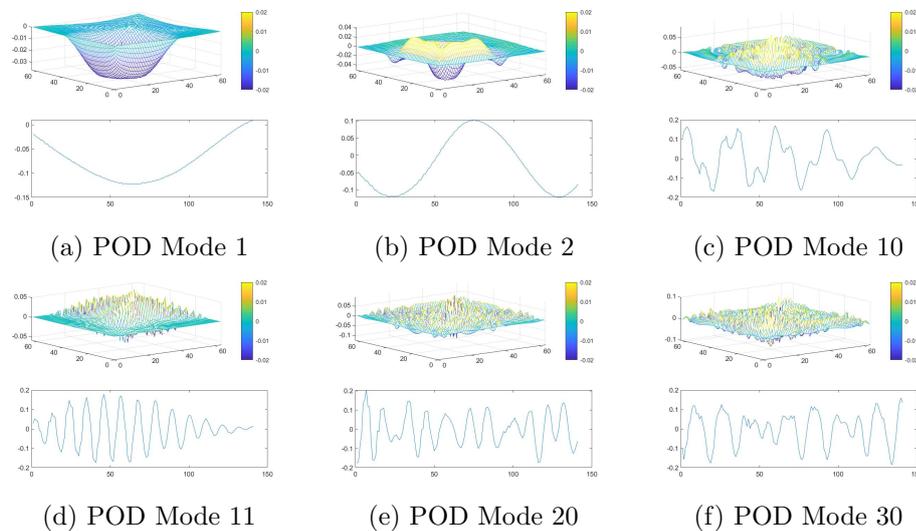


Figure 5.3.8: Presentation of POD mode and their temporal evolution, revealing that as the number of POD modes increases, the modes are revealing fast dynamics

It can be seen that the POD computes in the first 10 modes only the low frequency dynamics and only after the 11<sup>th</sup> mode, the high frequency

dynamics are presented. The POD manages to create modes that reveal low to high frequency dynamics in an ascending order. The POD manages to distinct the slow and fast dynamics directly from the first modes which makes it a powerful tool that deals successfully with models with multiscale characteristics.

#### 5.3.1.4 Wavelet Decomposition POD

The POD performed significantly good in the simulated example of the two wave model. The challenge is to evaluate if the wavelet POD can outperform the conventional POD. As proposed, in the previous section for the Wavelet-DMD, the wavelet decomposition is applied over the input data and then the POD is applied over the reconstructed images from the approximation and detail coefficients separately in the different scales.

In the following images, the first dominant modes in all selected scales are presented. The Wavelet-POD manages to separate the slow and fast moving dynamics in every single selected mode. It can be seen that even in the first mode, the reconstructed image from the approximation coefficients contains only the low frequency dynamics as presented in the single scale POD too. But the difference is that at the reconstructed images from the detail coefficients different dynamics emerge in the different scales of decomposition. The first scale image contains high frequency effects and in coarser scales, low frequency effects.

Presenting the results of the second POD mode, the same pattern is repeated. The wavelet-POD manages to separate the low and high effect in this mode among the different scales in the images reconstructed for the detail and approximation coefficients. Special interest shows the approximation representation. If it is compared to the one in the first mode, it can be seen that it starts to capture patterns of higher frequency.

Selecting the final dominant mode which for the sake of fair comparison with the previous methods, is the 12<sup>th</sup>, mode, the approximation coefficient mode representation contains high frequency dynamics too. This implies that the wavelet-POD approach can successfully reveal the system hidden characteristics using a small amount of modes, compared to the conventional single scale POD and approximates multiple state space models among the different scales that can describe the low and high frequency dynamics separately.

In a nutshell, each POD mode among the different scales, contains the high frequency dynamics in specific scales and the low frequency in finer scales separately. This introduces the big advantage of using multiple models based on the needs and purpose of the corresponding application and can give full insight of the behaviour of a given system.

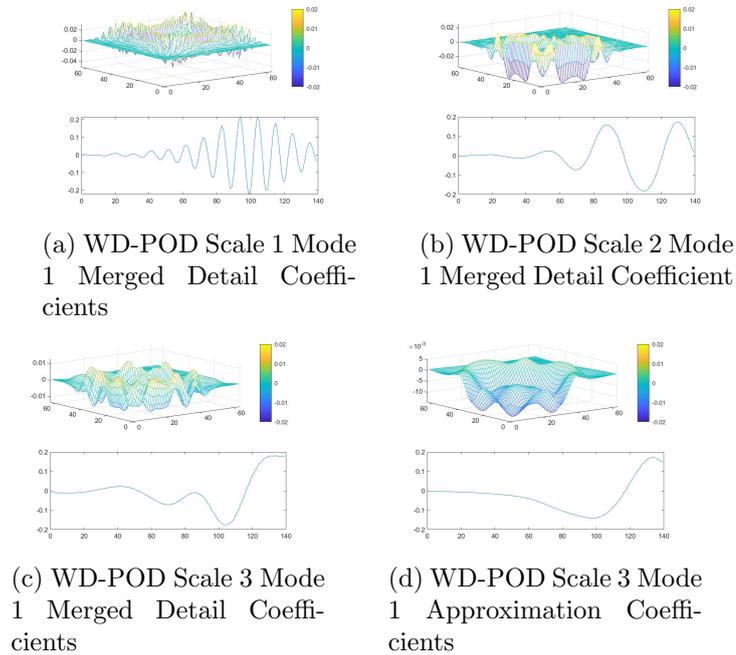


Figure 5.3.9: Presentation of the first 4 POD modes in the third scale of the detail and approximation coefficients of the wavelet decomposition using the Symlet mother wavelet, revealing that in the third scale, the slow dynamics are revealed

### 5.3.1.5 Conclusions

In conclusion, the novel DMD algorithm that is developed, gives rise to a novel class of multiscale reduced order identification methods that clearly reveals characteristics and hidden structures of a given system. Each single mode is decomposed into different scales of resolution and low and high frequency hidden dynamics are revealed separately. The second interesting point is that the hidden structures are revealed by using the minimum amount of modes and data.

These characteristics shape a powerful tool when it comes to computational efficiency and model representation of multiscale systems.

## 5.3.2 Manufacturing of 3D Printing Powder Process

The wavelet based POD and DMD showed significant improvements over the existing methods at pattern recognition and behaviour representation in the first example. A second, more complex example will be investigated that will test the consistency of the proposed method.

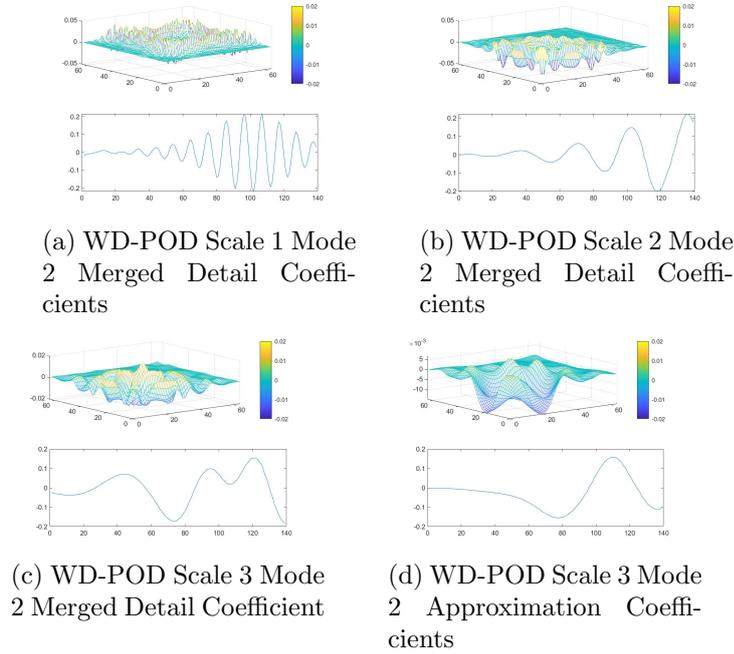


Figure 5.3.10: Presentation of the second POD mode of the detail and approximation coefficients of the wavelet decomposition using the Symlet mother wavelet, revealing that in the third scale, the slow dynamics are revealed

The new example presents a real-time manufacturing problem and the data were received from the university of Leeds and concern the manufacturing process of the 3-D printing powder that is used in the 3-D engraving printing process. During that process, one nozzle is spaying the powder and two nozzles are spraying air in order to compress the powder particles aiming at creating powder with more compact form and stronger particle connections. The problem that arises is that turbulences occur in certain areas that result in a poor quality in the final product. The aim is to reveal where the powder concentrations occur and in which areas they are most commonly presented. This will lead the user to adjust the nozzle spray speed and angle with ultimate goal the improvement of the overall process and the final product quality.

A high resolution camera is used to capture consequent image frames of the whole process in order to gather the necessary data. These frames will be used as the input dataset for the different DMD and POD algorithms. As shown in the previous example, it will be investigated if the extracted POD and DMD modes can potentially reveal coherent and dominant structures

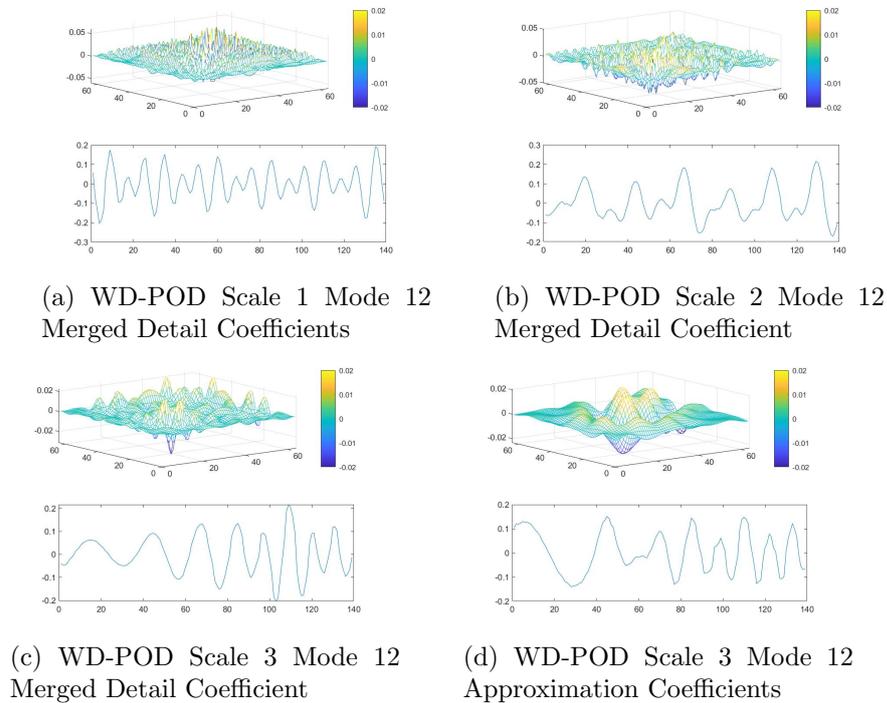


Figure 5.3.11: Presentation of the last POD mode of the detail and approximation coefficients of the wavelet decomposition using the Symlet mother wavelet, revealing that in the third scale, the slow dynamics are revealed

of the particle concentrations of these high dimensional images. The modes will target where and in which amount the turbulences occur and how much the process performance is affected.

An amount of 2100 images of size  $800 \times 312$  is given. This sequence of images will be used in the DMD algorithms and each of them will be transformed into a single column vector for the purpose of the method. Due to the high dimensionality of the images and due to computational and memory issues, the examples use a total amount of 300 images in this example.

The following figures illustrate specific snapshots presenting the manufacturing process. Each figure contains the image of the process but also its contour counterpart in order to illustrate the image variations in a more clear way.

### 5.3.2.1 Hankel-DMD

As presented in the previous example, the existing reduced order identification methods will be evaluated and compared with the new multiscale

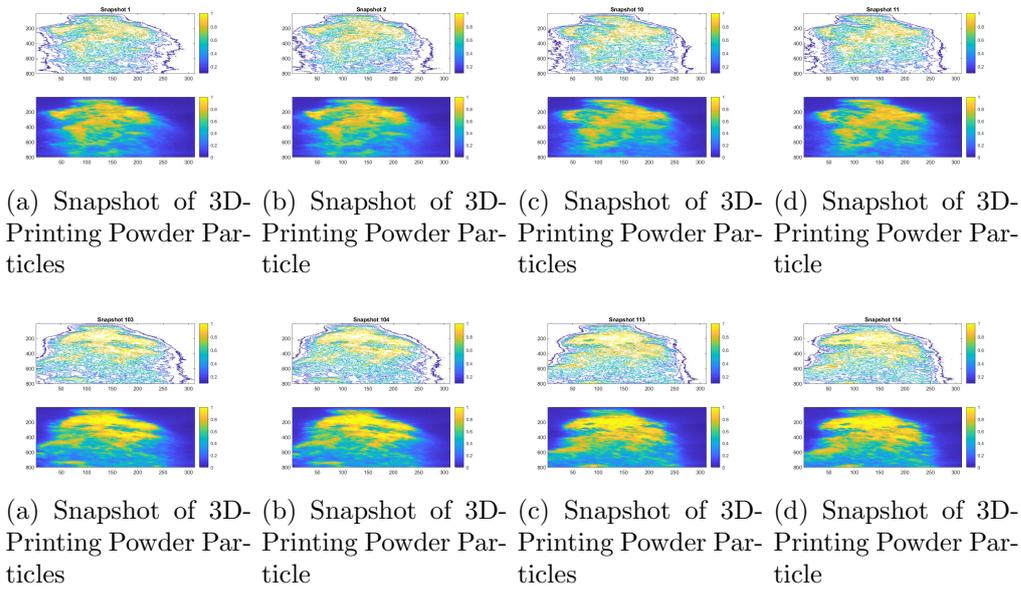


Figure 5.3.12: Simulation of the real-time process of the powder flow based on frames taken by high resolution camera

identification method. In this section, the performance and results of the Hankel-DMD will be presented.

Given 300 consequent images of the particles flow, the selection of the dominant POD modes is taken by examining the SVD plot given below. A total amount of 60 modes representing the 90% of the total system energy, is selected.

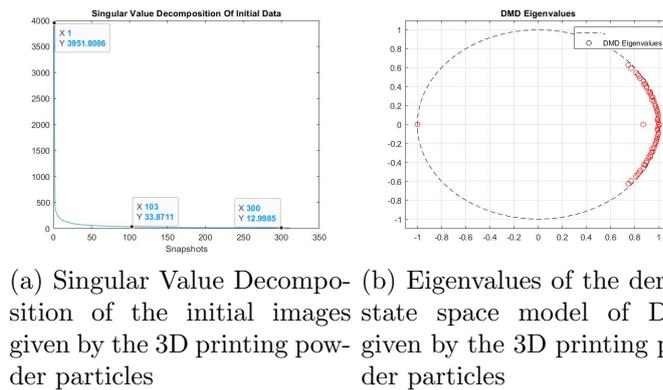


Figure 5.3.13: Analysis of the Hankel-DMD

Based on these modes, the DMD creates a state space model that best describes the given data. Examining the eigen-decomposition of the derived

model, it is shown that the eigenvalues are concentrated in the edge of the unitary circle, indicating that the new state space model is governed by high frequency dynamics.

The following images demonstrate the dominant DMD modes of the derived state space system. These modes should give insight of where the particles are concentrated, indicating the location of where the turbulences occur.

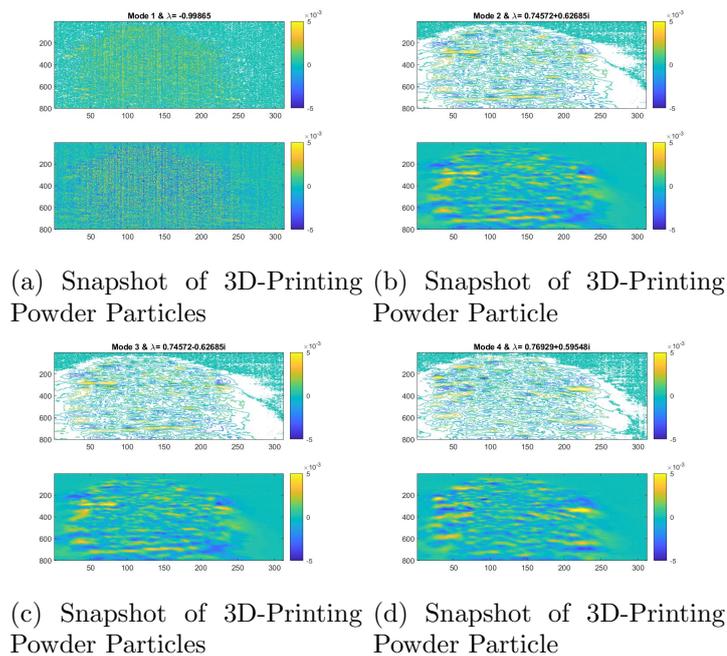


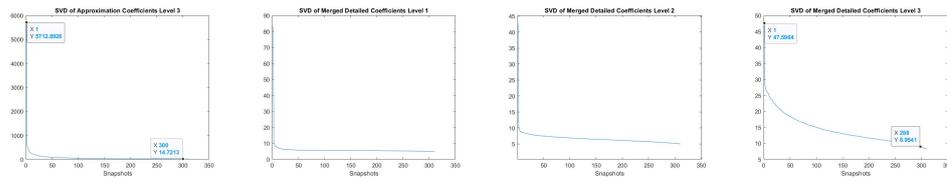
Figure 5.3.14: Simulation of the particles flow

The first impression is that all the DMD modes can only reveal high frequency dynamics and by observing the DMD modes, it is not clear how the system energy is spread and where the turbulences are occurring. Especially, by observing the first mode, it can be seen that the dominant mode is governed by noise which can justify the poor performance of the DMD algorithm at revealing the dominant structures of the system. This will be addressed by using the Wavelet Decomposition DMD.

### 5.3.2.2 Wavelet Decomposition DMD

As presented in the previous example, the initial data will be decomposed into multiple levels using the Wavelet Decomposition based on the same wavelet function, the Symlet mother wavelet. The Hankel-DMD algorithm will be

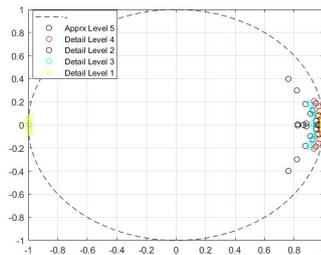
applied over the data in all scales of resolution. The amount of selected dominant modes is given by observing the SVD plot of all POD modes and the amount of them is the same for all approximation and detail level of coefficients. The following figures present the SVD decomposition step and the energy concentration of first dominant modes for the approximation and detail coefficients respectively



(a) SVD at Scale 3 - Approximation Coefficients (b) SVD at Scale 3 - Detail Coefficients (c) SVD at Scale 2 - Detail Coefficients (d) SVD at Scale 1 - Detail Coefficients

Figure 5.3.15: SVD plots of the approximation and detail coefficients for the 3 level wavelet decomposition

Following the same idea as in the previous example, the data will be decomposed into 5 levels in this case, hence a total amount of 12 dominant modes per scale is selected. Based on the SVD plots, the amount of 12 modes is totally adequate for the purpose of this project. Following the SVD step, the Hankel-DMD is applied over each set of detail and approximation coefficients in every scale, deriving one dynamic model per set of coefficients.



(a) Eigenvalues of the derived state space model of DMD given by the 3D printing powder particles

As a first comparison with the single scale Hankel-DMD where all eigenvalues are concentrated around the unity circle, in the multiscale Hankel-DMD using Wavelet Decomposition, there is presence of eigenvalues inside

the unit circle. This indicates the unveiling of slow-slower dynamics among the scales. Hence, the wavelet DMD can possibly reveal various dynamics that the conventional DMD is not capable of.

The analysis will start by the following images that present the dominant DMD modes of the reconstructed images from the coefficients in scale 5.

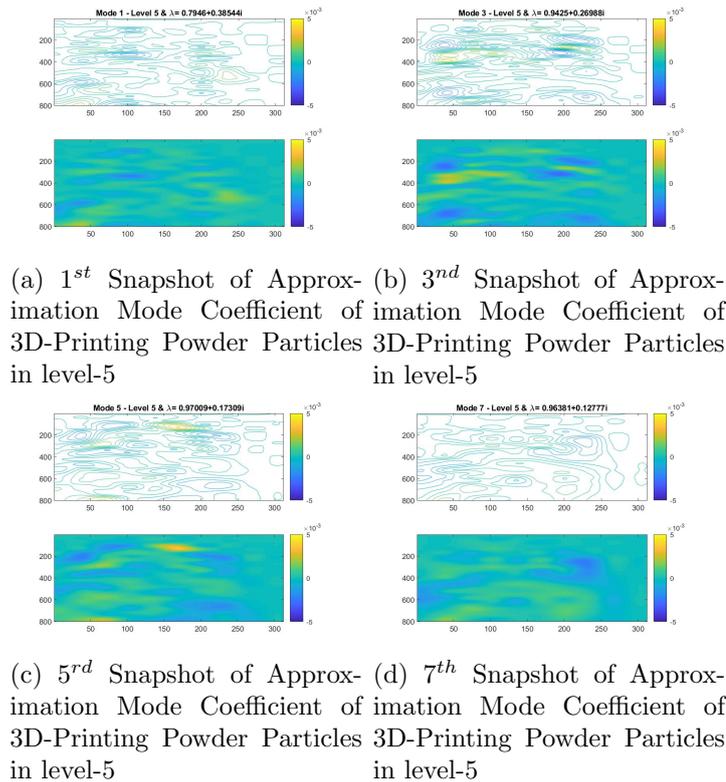
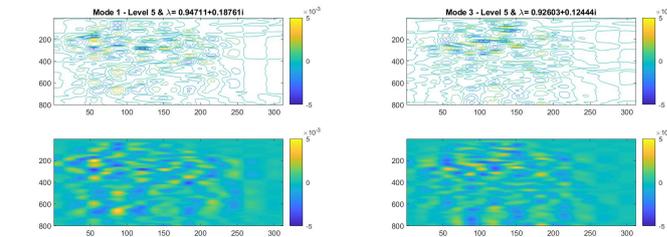


Figure 5.3.16: DMD modes of the approximation coefficients at coarser level of decomposition

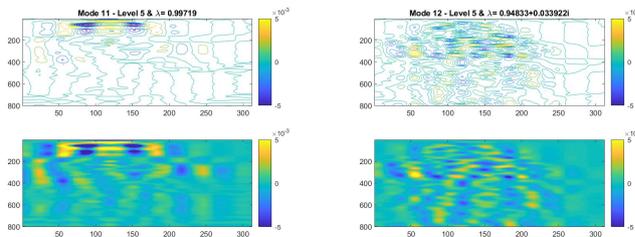
The modes derived from the reconstruction of the approximation coefficients indicate that the particles are concentrated mostly in the left hand side of the powder flow. In comparison to the modes derived in the Hankel DMD shown in Fig. 5.3.14, the particle concentration is more clear in the proposed method. Especially by observing the first mode of the Hankel DMD which is governed by noise, the wavelet DMD provides better results in order to understand the behaviour of the system.

The following images demonstrate the modes derived from the DMD when applied over reconstructed images of the wavelet detail coefficients. The results are quite interesting. The DMD modes unfold patterns that were not present in any of the DMD modes when the single scale DMD is applied

(see Fig.5.3.17). The first modes proclaim similar dynamics as shown in the single-scale DMD but in the following modes the patterns are changing.



(a) 1<sup>st</sup> Snapshot of Detail Mode Coefficient of 3D-Printing Powder Particles in level-5  
 (b) 3<sup>rd</sup> Snapshot of Detail Mode Coefficient of 3D-Printing Powder Particles in level-5



(c) 11<sup>th</sup> Snapshot of Detail Mode Coefficient of 3D-Printing Powder Particles in level-5  
 (d) 12<sup>th</sup> Snapshot of Detail Mode Coefficient of 3D-Printing Powder Particles in level-5

Figure 5.3.17: DMD modes of the detail coefficients at the coarser level 5 of decomposition

The images below present the 11<sup>th</sup> mode of the reconstructed images in scale 5. The images show an energy concentration in the whole top end area of the image which is close to the nozzles. These patterns give a more precise insight about the area where particles are concentrated and how the nozzles need to be adjusted.

The 11<sup>th</sup> mode performs a completely different pattern compared to the other modes in all DMD algorithms. Hence, the group of figures below illustrate the 11<sup>th</sup> DMD mode in every scale of decomposition. It is shown that in the first scales, the modes are governed by the horizontal detail coefficients effects and do not demonstrate any figures that can rise to multiscale characteristics or structures that cannot be seen in single scale decomposition.

In coarser scales, the DMD modes illustrate new patterns that are not shown in any single-scaled DMD algorithm. These patterns are not shown

only in the coarsest scale but also in scale No3 and No4, meaning that there are occasions where the user can see effects in finer scales without the need of decomposing in deep scales, in favour of computational cost and time.

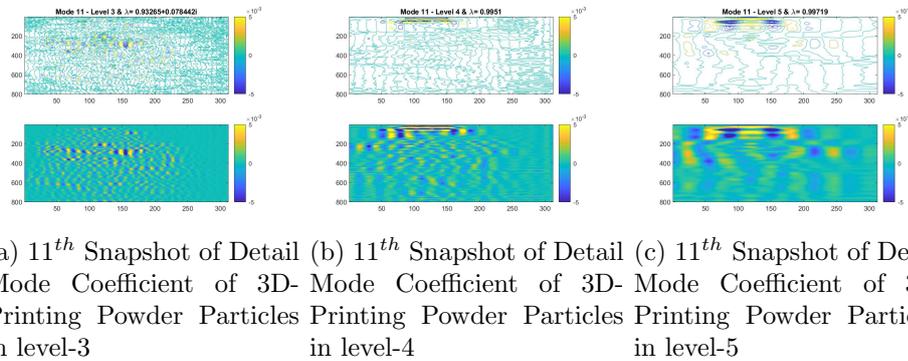


Figure 5.3.18: Example of a specific DMD mode presenting the detail coefficients at all levels of decomposition

This diversity in pattern recognition is repeated in multiple modes, giving the confidence that the wavelet decomposition DMD can help massively in understanding the behaviour of complex models with high dimensional data.

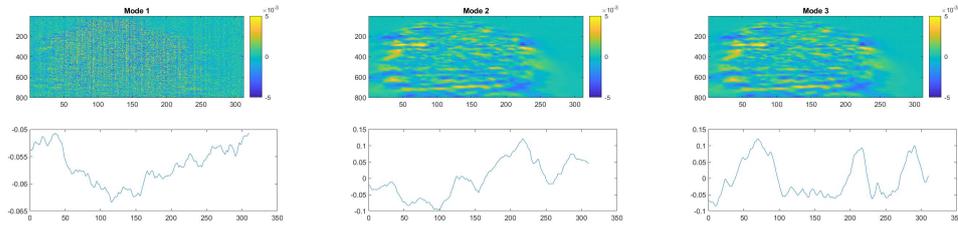
### 5.3.2.3 POD

In occasions where the derivation of a state space realisation model is not needed, the POD algorithm could be more of use for the purpose of the application. Apart from the DMD methods, a comparison between the conventional POD and the multiscale POD via wavelet decomposition will be given. Similarly to the previous algorithm, the following images demonstrate the POD modes and their temporal evolution in descending order in terms of their energy concentration.

The POD is indeed arranging the modes in descending order, in terms of the energy but it cannot reveal any hidden structure as opposed in the previous methods. The POD modes present the same pattern as the modes in the single-scale DMD where the most dominant mode is governed by noise. The modes indicate specific patterns which are similar but do not reveal any particle concentrations or structures that could guide the user over the system behaviour.

### 5.3.2.4 Wavelet Decomposition POD

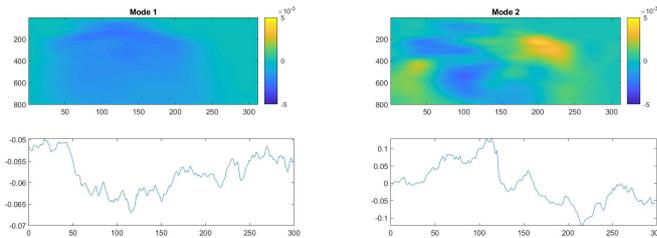
As proposed in the Wavelet decomposition DMD where hidden structures and patterns were revealed, the use of the Wavelet POD will try to reveal



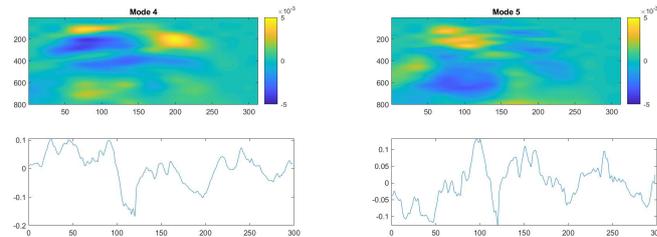
(a) Snapshot of the 1<sup>st</sup> 3D-Printing Powder Particle POD Mode  
 (b) Snapshot of the 2<sup>nd</sup> 3D-Printing Powder Particle POD Mode  
 (c) Snapshot of the 3<sup>rd</sup> 3D-Printing Powder Particle POD Mode

Figure 5.3.19: Presentation of the most dominant POD modes

similar behaviour that was not shown by using the conventional POD. The following images demonstrate the modes from the reconstructed images based on the approximation coefficients.



(a) 1<sup>st</sup> Snapshot of Approx- (b) 2<sup>nd</sup> Snapshot of Approx-  
 imation Mode Coefficient of imation Mode Coefficient of  
 3D-Printing Powder Particles 3D-Printing Powder Particles  
 in level-5 in level-5



(c) 4<sup>th</sup> Snapshot of Approx- (d) 5<sup>th</sup> Snapshot of Approx-  
 imation Mode Coefficient of imation Mode Coefficient of  
 3D-Printing Powder Particles 3D-Printing Powder Particles  
 in level-5 in level-5

Figure 5.3.20: Presentation of the dominant POD modes of the approximation coefficients in the coarsest level 5

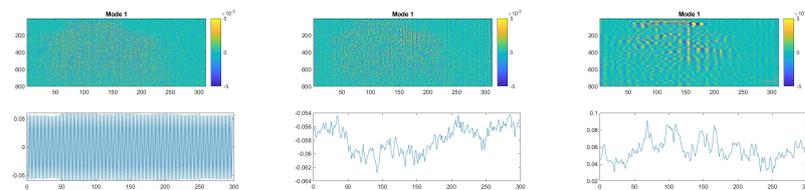
A closer observation between the first dominant modes of this method and

the one from the conventional POD, indicates the beneficial characteristics of the wavelets that filter the high frequency effects. The above POD modes show a concentration of particles in the top areas and in the top right and left corners. These patterns show a match with the wavelet DMD modes giving the user the confidence about the particles allocation.

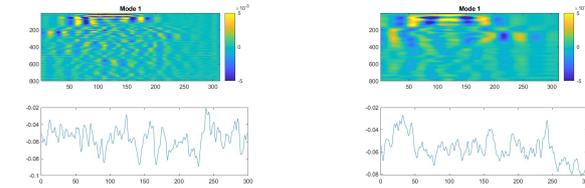
Similar results are shown if the user compares the remaining modes of the conventional POD with the ones of the approximation coefficients in the finest scale.

The POD modes at coarser scales of resolution are representing concentrated energy in locations where the conventional POD cannot reveal. In fact, they present similarities with the wavelet DMD modes, enhancing the assumptions that the wavelet based reduced order algorithms, reveal hidden structures and behaviours that cannot be seen with singled scaled methods.

It is of a great interest to demonstrate the variation of a single mode across the different scales. The following images present the most dominant mode of the wavelet DMD across the 5 scales of decomposition. It is shown that there is a big variation in the patterns that are captured among the scales for the modes derived from the reconstructed images of the detail coefficients. The mode in the first scale is corrupted with noise while the rest of the modes reveal patterns of the actual system behaviour. As modes of coarser scales are presented, different patterns are unfolded, presenting the concentration of the particles. As shown in the application of the wavelet DMD, the POD modes show high particle concentration in the top areas of the images.



(a) 1<sup>st</sup> Snapshot of De- (b) 1<sup>st</sup> Mode of De- (c) 1<sup>st</sup> Mode of De-  
 tail Mode Coefficient tail Coefficient of 3D- tail Coefficient of 3D-  
 of 3D-Printing Powder Printing Powder Parti- Printing Powder Parti-  
 cles in level-1 cles in level-2 cles in level-3



(a) 1<sup>st</sup> Mode of Detail Coefficient of 3D-Printing Powder Particles in level-4

(b) 1<sup>st</sup> Mode of Detail Coefficient of 3D-Printing Powder Particles in level-5

## 5.4 Summary

In this chapter, a novel algorithm based on the DMD is created for models with multiscale characteristics. The novel multiscale algorithm combines the properties of the wavelet decomposition and the DMD in order to create a framework that can deal with complex multiscale models. The use of the wavelets allows a pre-stage the filtering of the input data and secondly the wavelet properties have the ability to separate the system mixed dynamics among the different scales of decomposition. Moreover, due to the nature of the wavelets, they are able to capture abrupt changes and behaviours in the system that conventional methods cannot reveal.

The wavelet DMD was applied over two multiscale examples, one that was designed and simulated for the purpose of the thesis and one for a real time application for dataset with unknown characteristics. A straight comparison with the existing DMD and POD methods was applied showing the fact that the wavelet DMD outperforms the other methods. The wavelet DMD manages to reveal structures inside the data that cannot be shown with other methods. Mixed frequency patterns are separated and the system behaviour is clear to the user.

Hence, the wavelet DMD comprises a new very useful tool in data-driven system identification techniques with broad application in data of any nature; linear, non-linear, time-invariant or time-varying.

In the next chapter, the two presented examples will be demonstrated in detail in order to indicate the consistency of the proposed Wavelet-DMD and POD approaches.



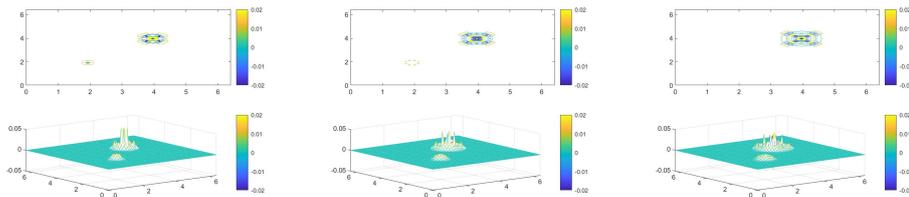
# Chapter 6

## Results & Discussions

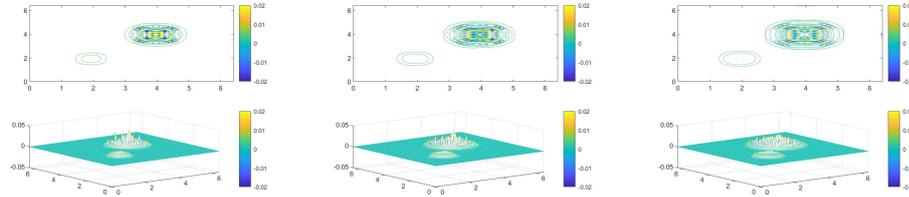
This chapter will present extensively the simulations of the two examples described above. In each of the two examples, the details simulations of the initial models, the derived DMD modes for each case will be demonstrated, providing an image to image comparison of the proposed method to the existing algorithms. In this chapter, the ascendancy of the Wavelet-DMD will be lucid over the existing single-scale and multi-scale DMD methods.

### 6.1 Wave Diffusion Simulation

As exhibited in the previous chapter, the first example that will be analysed is the simulation of a two wave system that oscillate in different frequencies and amplitudes, generating a two scale dynamic system. The following images present the initial model behaviour over the first 85 out of 150 time steps.



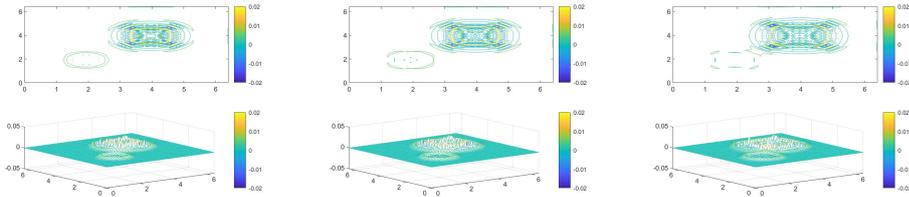
(a) Wave Diffusion Simulation of oil and water drop in step  $k=1$  (b) Wave Diffusion Simulation of oil and water drop in step  $k=5$  (c) Wave Diffusion Simulation of oil and water drop in step  $k=10$



(a) Wave Diffusion Simulation of oil and water drop in step  $k=15$

(b) Wave Diffusion Simulation of oil and water drop in step  $k=20$

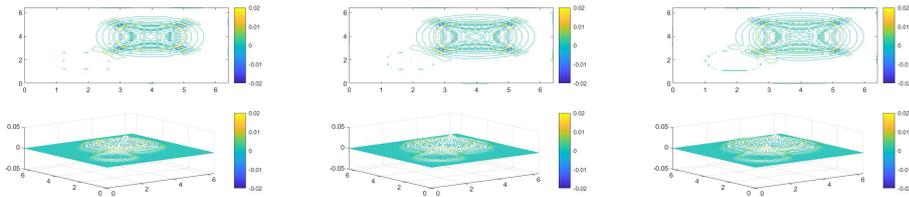
(c) Wave Diffusion Simulation of oil and water drop in step  $k=25$



(d) Wave Diffusion Simulation of oil and water drop in step  $k=30$

(e) Wave Diffusion Simulation of oil and water drop in step  $k=35$

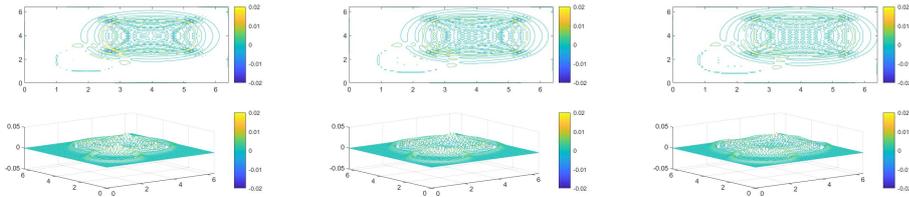
(f) Wave Diffusion Simulation of oil and water drop in step  $k=40$



(g) Wave Diffusion Simulation of oil and water drop in step  $k=45$

(h) Wave Diffusion Simulation of oil and water drop in step  $k=50$

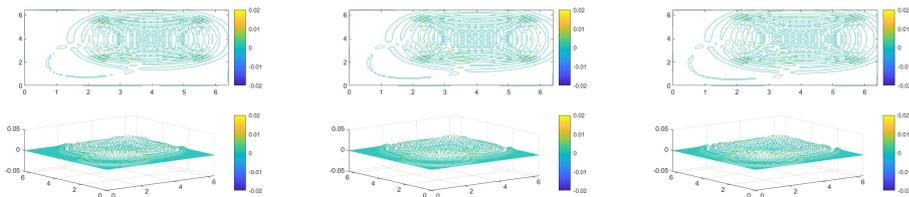
(i) Wave Diffusion Simulation of oil and water drop in step  $k=55$



(j) Wave Diffusion Simulation of oil and water drop in step  $k=60$

(k) Wave Diffusion Simulation of oil and water drop in step  $k=65$

(l) Wave Diffusion Simulation of oil and water drop in step  $k=70$

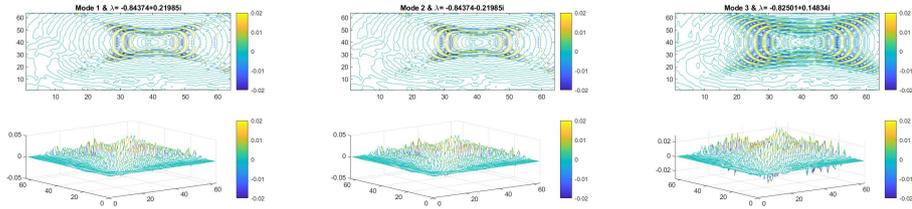
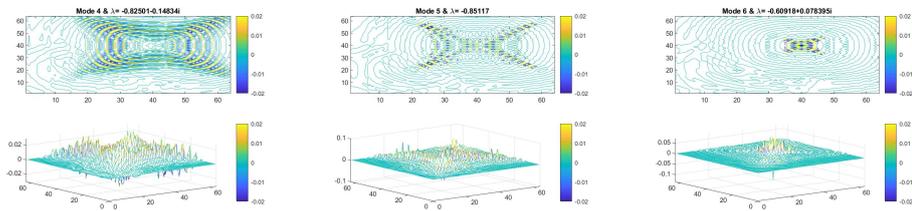
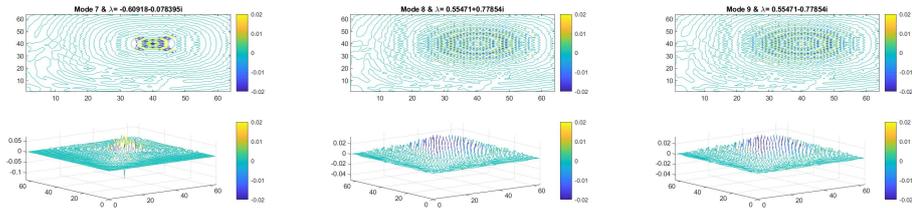
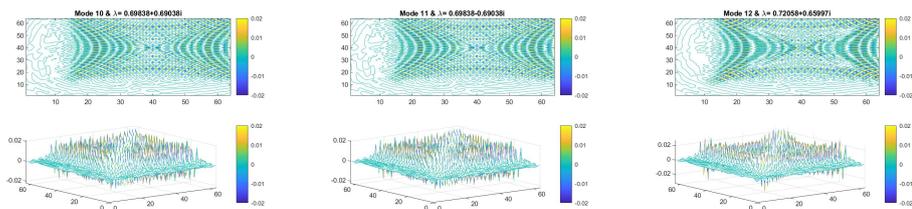


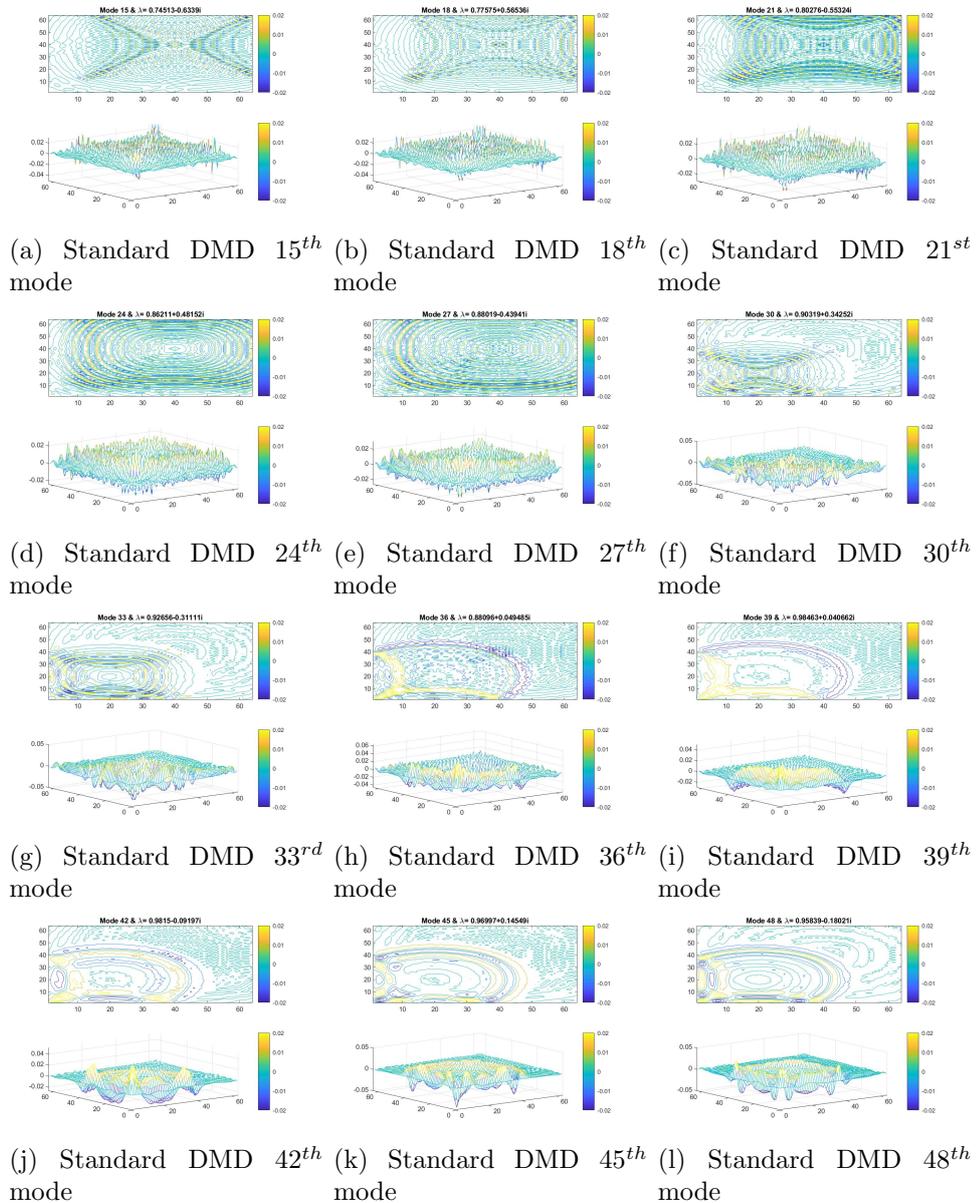
(m) Wave Diffusion Simulation of oil and water drop in step  $k=75$

(n) Wave Diffusion Simulation of oil and water drop in step  $k=80$

(o) Wave Diffusion Simulation of oil and water drop in step  $k=85$

The following images demonstrate the modes derived after applying Hankel-2 DMD algorithm.

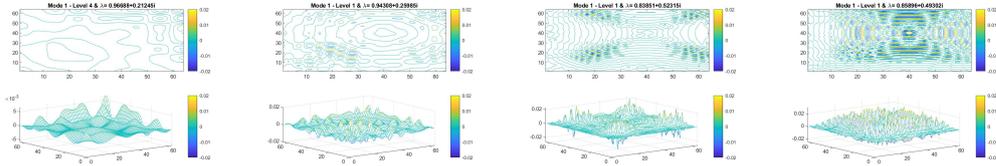
(a) Standard DMD 1<sup>st</sup> mode(b) Standard DMD 2<sup>nd</sup> mode(c) Standard DMD 3<sup>rd</sup> mode(d) Standard DMD 4<sup>th</sup> mode(e) Standard DMD 5<sup>th</sup> mode(f) Standard DMD 6<sup>th</sup> mode(g) Standard DMD 7<sup>th</sup> mode(h) Standard DMD 8<sup>th</sup> mode(i) Standard DMD 9<sup>th</sup> mode(j) Standard DMD 10<sup>th</sup> mode(k) Standard DMD 11<sup>th</sup> mode(l) Standard DMD 12<sup>th</sup> mode



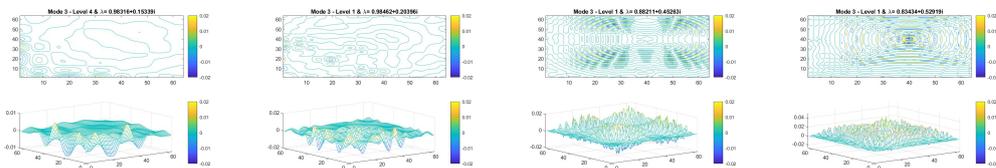
As presented in the previous chapter, the Hankel-2 DMD manages to separate the different frequency dynamics after using a minimum of 30 modes. This indicates that 20% of the amount of given images are used for determining the different dynamics. This poses a problem in cases where a smaller amount of modes is selected.

The wavelet-DMD will be presented below, addressing and resolving this problem. As shown in the previous chapter, when the wavelet-DMD is applied, every mode reveals the dynamic effects in different scales. Therefore,

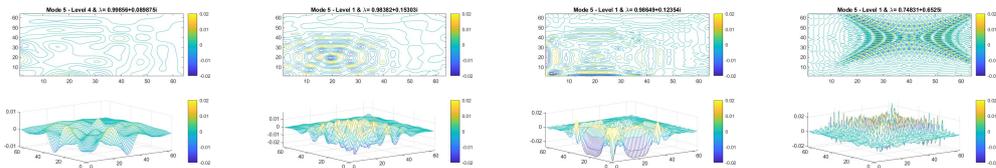
this approach can provide full insight on the system behaviour.



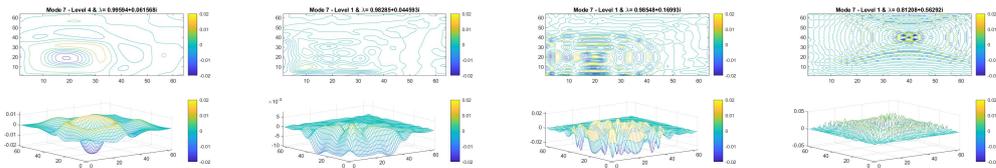
(a) Wavelet-DMD 1<sup>st</sup> Mode at Scale 4 (b) Wavelet-DMD 1<sup>st</sup> Mode at Scale 3 (c) Wavelet-DMD 1<sup>st</sup> Mode at Scale 2 (d) Wavelet-DMD 1<sup>st</sup> Mode at Scale 1



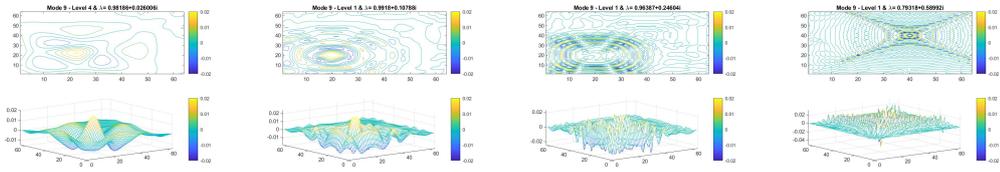
(a) Wavelet-DMD 3<sup>rd</sup> Mode at Scale 4 (b) Wavelet-DMD 3<sup>rd</sup> Mode at Scale 3 (c) Wavelet-DMD 3<sup>rd</sup> Mode at Scale 2 (d) Wavelet-DMD 3<sup>rd</sup> Mode at Scale 1



(a) Wavelet-DMD 5<sup>th</sup> Mode at Scale 4 (b) Wavelet-DMD 5<sup>th</sup> Mode at Scale 3 (c) Wavelet-DMD 5<sup>th</sup> Mode at Scale 2 (d) Wavelet-DMD 5<sup>th</sup> Mode at Scale 1



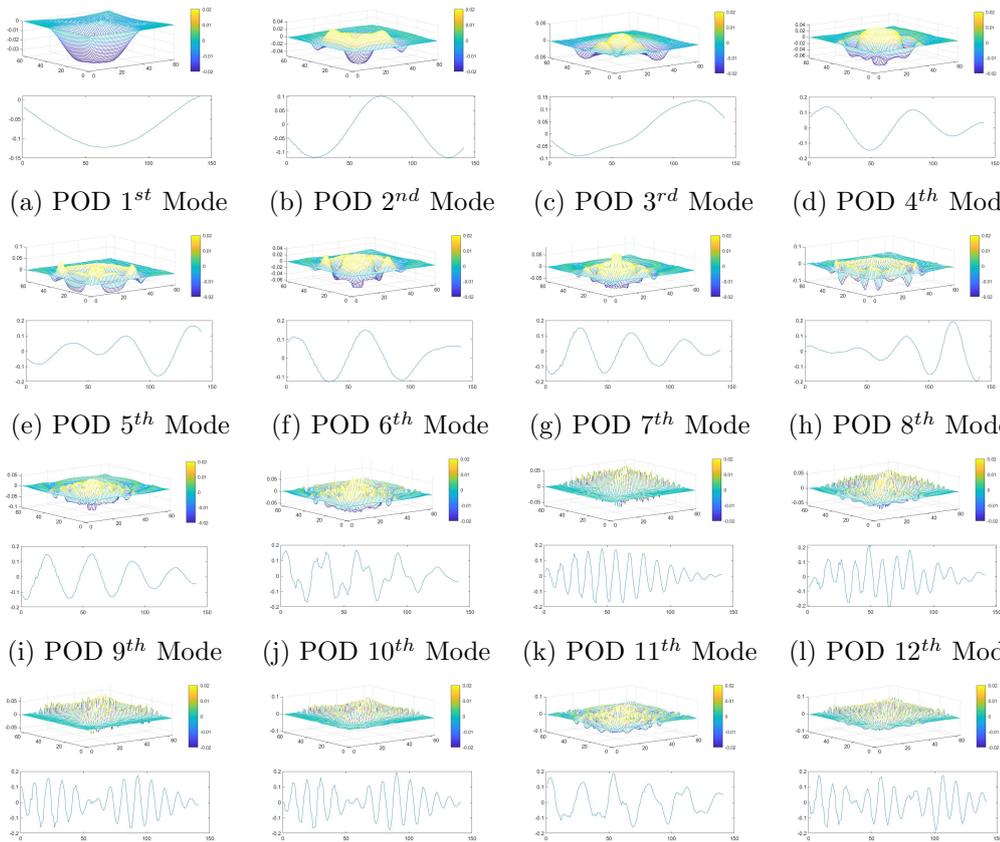
(a) Wavelet-DMD 7<sup>th</sup> Mode at Scale 4 (b) Wavelet-DMD 7<sup>th</sup> Mode at Scale 3 (c) Wavelet-DMD 7<sup>th</sup> Mode at Scale 2 (d) Wavelet-DMD 7<sup>th</sup> Mode at Scale 1



(a) Wavelet-DMD 9<sup>th</sup> Mode at Scale 4    (b) Wavelet-DMD 9<sup>th</sup> Mode at Scale 3    (c) Wavelet-DMD 9<sup>th</sup> Mode at Scale 2    (d) Wavelet-DMD 9<sup>th</sup> Mode at Scale 1

In occasions where the user does not require to extract a new dynamic model, the POD algorithms can be used in order to separate the given model into its dominant structures. Therefore, a comparison between the standard POD and Wavelet-POD was given.

The following images, demonstrate the POD modes derived by using the standard POD.

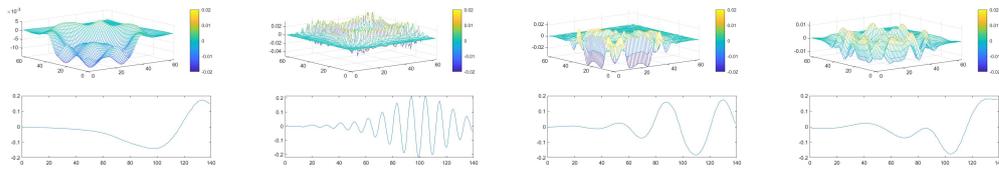


(a) POD 1<sup>st</sup> Mode    (b) POD 2<sup>nd</sup> Mode    (c) POD 3<sup>rd</sup> Mode    (d) POD 4<sup>th</sup> Mode  
 (e) POD 5<sup>th</sup> Mode    (f) POD 6<sup>th</sup> Mode    (g) POD 7<sup>th</sup> Mode    (h) POD 8<sup>th</sup> Mode  
 (i) POD 9<sup>th</sup> Mode    (j) POD 10<sup>th</sup> Mode    (k) POD 11<sup>th</sup> Mode    (l) POD 12<sup>th</sup> Mode  
 (m) POD 13<sup>th</sup> Mode    (n) POD 14<sup>th</sup> Mode    (o) POD 15<sup>th</sup> Mode    (p) POD 16<sup>th</sup> Mode

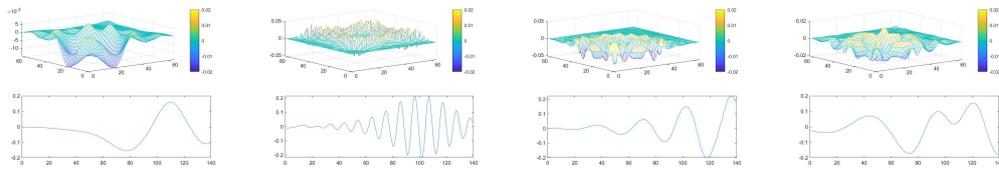
As discussed in the previous chapter, the standard POD has shown good performance at separating the different system effects. The first 9 modes

contain the low frequency spatial dynamics and the latter modes comprise the high frequency dynamics.

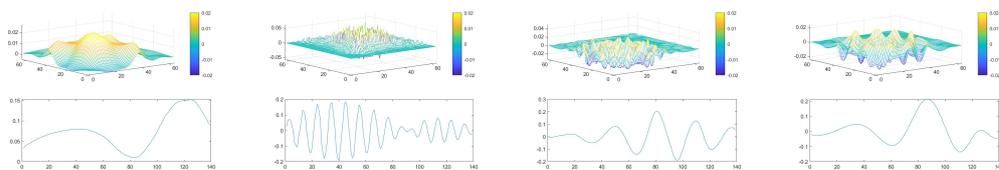
The following images include the first ten dominant modes of the Wavelet-POD. In each row, the modes derived from the reconstructed approximation and detail coefficients are presented. It is clearly shown, that in each scale, the modes derived for the reconstructed images from the detail and approximation coefficients, include different dynamics and the mixed effects are separated.



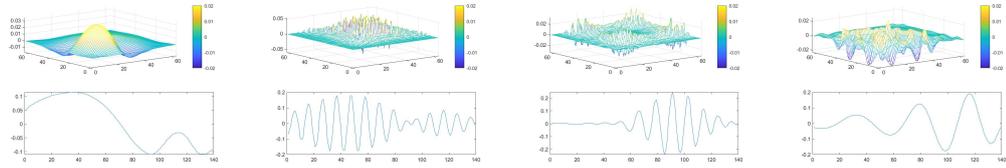
(a) Wavelet-POD 1<sup>th</sup> Mode Scale-3 (Ap- prx.) (b) Wavelet-POD 1<sup>th</sup> Mode Scale-1 (c) Wavelet-POD 1<sup>th</sup> Mode Scale-2 (d) Wavelet-POD 1<sup>th</sup> Mode Scale-3



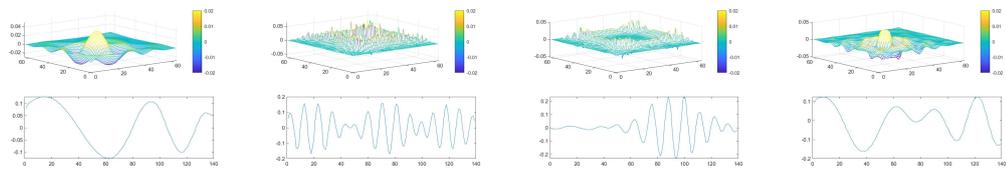
(a) Wavelet-POD 2<sup>nd</sup> Mode Scale-3 (Ap- prx.) (b) Wavelet-POD 2<sup>nd</sup> Mode Scale-1 (c) Wavelet-POD 2<sup>nd</sup> Mode Scale-2 (d) Wavelet-POD 2<sup>nd</sup> Mode Scale-3



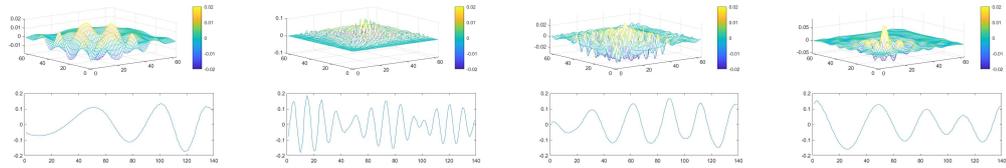
(a) Wavelet-POD 3<sup>rd</sup> Mode Scale-3 (Ap- prx.) (b) Wavelet-POD 3<sup>rd</sup> Mode Scale-3 (c) Wavelet-POD 3<sup>rd</sup> Mode Scale-1 (d) Wavelet-POD 3<sup>rd</sup> Mode Scale-2



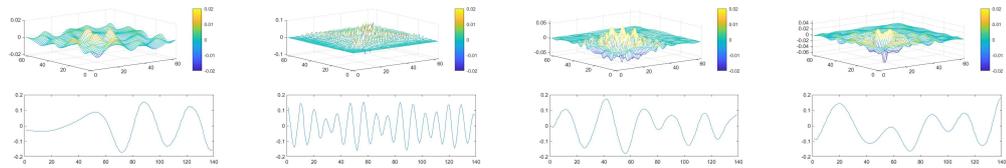
(a) Wavelet-POD 4<sup>th</sup> Mode Scale-3 (Ap-prx.) (b) Wavelet-POD 4<sup>th</sup> Mode Scale-3 (c) Wavelet-POD 4<sup>th</sup> Mode Scale-1 (d) Wavelet-POD 4<sup>th</sup> Mode Scale-2



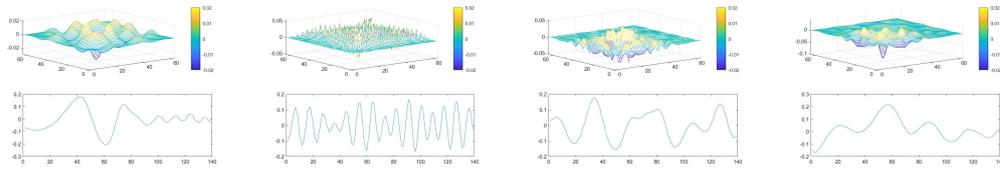
(a) Wavelet-POD 5<sup>th</sup> Mode Scale-3 (Ap-prx.) (b) Wavelet-POD 5<sup>th</sup> Mode Scale-3 (c) Wavelet-POD 5<sup>th</sup> Mode Scale-1 (d) Wavelet-POD 5<sup>th</sup> Mode Scale-2



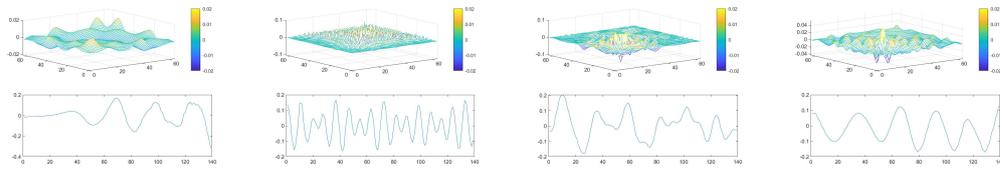
(a) Wavelet-POD 6<sup>th</sup> Mode Scale-3 (Ap-prx.) (b) Wavelet-POD 6<sup>th</sup> Mode Scale-3 (c) Wavelet-POD 6<sup>th</sup> Mode Scale-1 (d) Wavelet-POD 6<sup>th</sup> Mode Scale-2



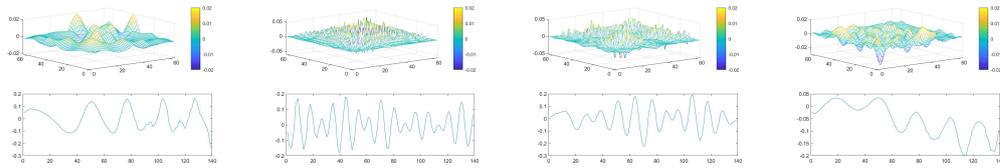
(a) Wavelet-POD 7<sup>th</sup> Mode Scale-3 (Ap-prx.) (b) Wavelet-POD 7<sup>th</sup> Mode Scale-3 (c) Wavelet-POD 7<sup>th</sup> Mode Scale-1 (d) Wavelet-POD 7<sup>th</sup> Mode Scale-2



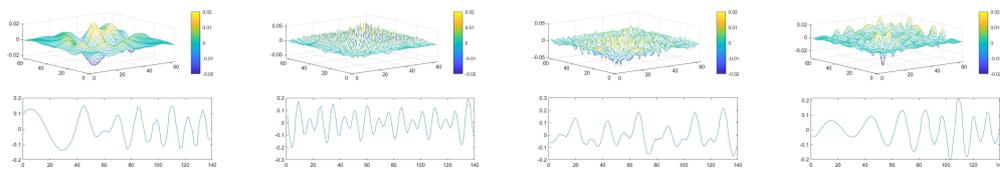
(a) Wavelet-POD 8<sup>th</sup> Mode Scale-3 (Appr.) (b) Wavelet-POD 8<sup>th</sup> Mode Scale-3 (c) Wavelet-POD 8<sup>th</sup> Mode Scale-1 (d) Wavelet-POD 8<sup>th</sup> Mode Scale-2



(a) Wavelet-POD 9<sup>th</sup> Mode Scale-3 (Appr.) (b) Wavelet-POD 9<sup>th</sup> Mode Scale-3 (c) Wavelet-POD 9<sup>th</sup> Mode Scale-1 (d) Wavelet-POD 9<sup>th</sup> Mode Scale-2



(a) Wavelet-POD 10<sup>th</sup> Mode Scale-3 (Appr.) (b) Wavelet-POD 10<sup>th</sup> Mode Scale-3 (c) Wavelet-POD 10<sup>th</sup> Mode Scale-1 (d) Wavelet-POD 10<sup>th</sup> Mode Scale-2



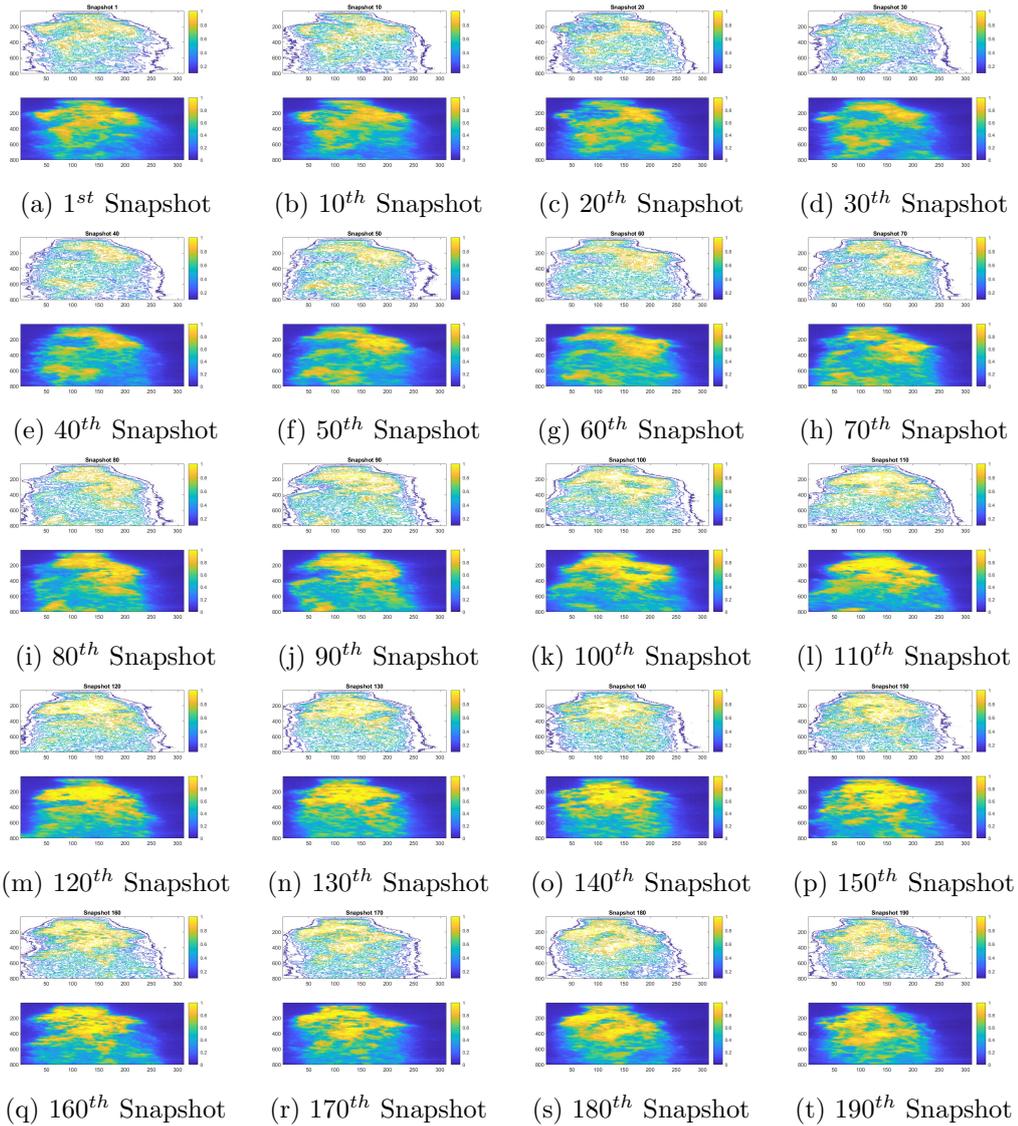
(a) Wavelet-POD 12<sup>th</sup> Mode Scale-3 (Appr.) (b) Wavelet-POD 12<sup>th</sup> Mode Scale-1 (c) Wavelet-POD 12<sup>th</sup> Mode Scale-2 (d) Wavelet-POD 12<sup>th</sup> Mode Scale-3

## 6.2 Flow of molten 3-D printing powder

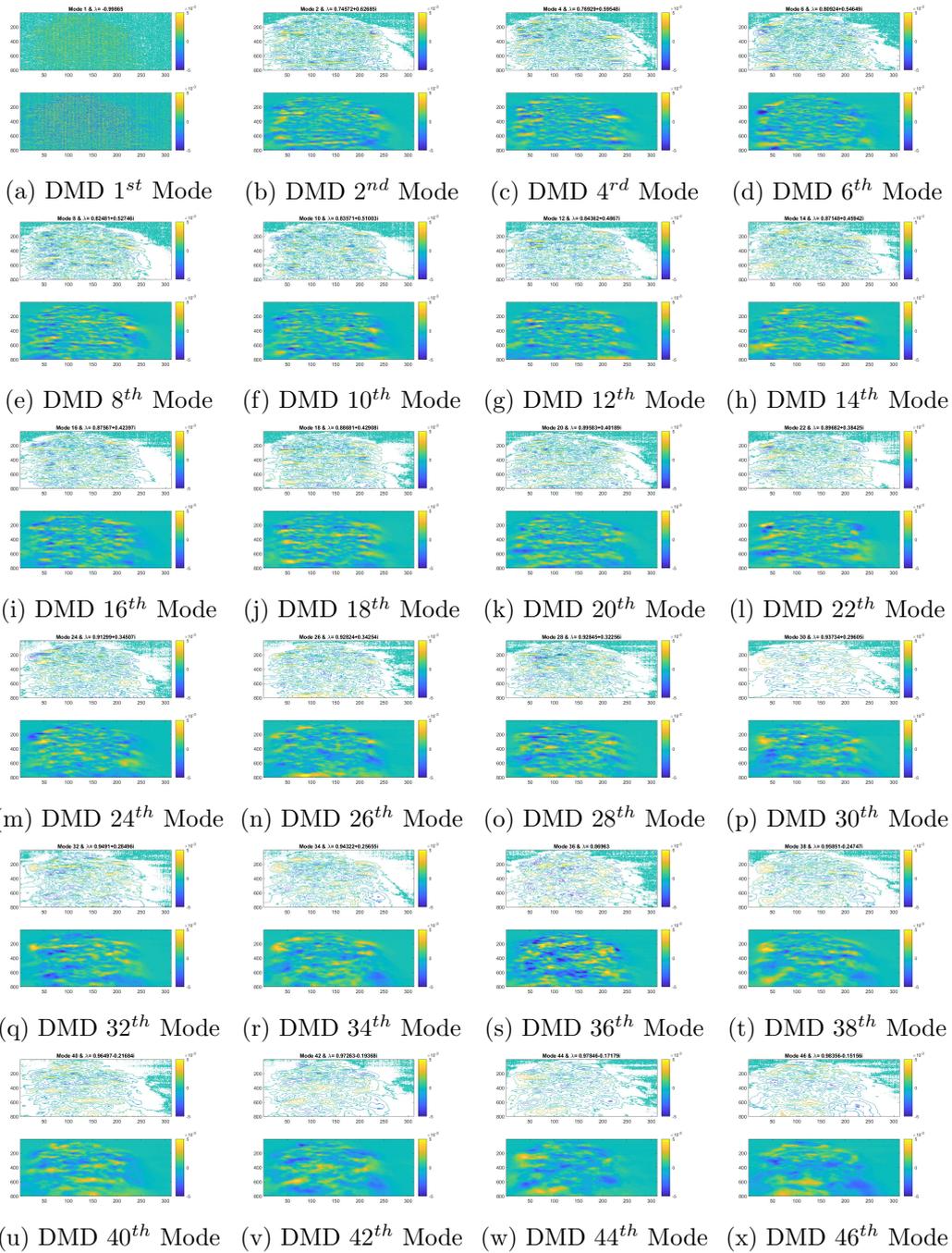
The second example that was evaluated concerned the process during manufacturing the powder of 3D printers. In this process, the flow of molten

powder particles is disturbed by turbulences occurring in certain areas. Key target is the identification of these areas and the configuration of the 3 nozzle-system in order to avoid that phenomenon and optimize the product quality.

The following images demonstrate the particles flow analytically, providing an understanding of the model behaviour.

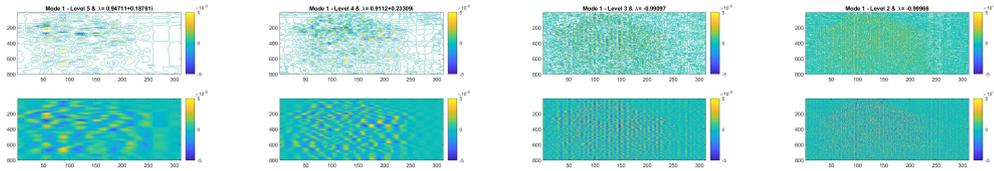


The following images present the DMD modes as derived by implementing the Hankel-2 DMD algorithm. It is interesting to mention that the dominant mode is governed by noise. Moreover, by observing the different modes, it is shown that this approach cannot provide specific patterns that could reveal where the particles are concentrated.

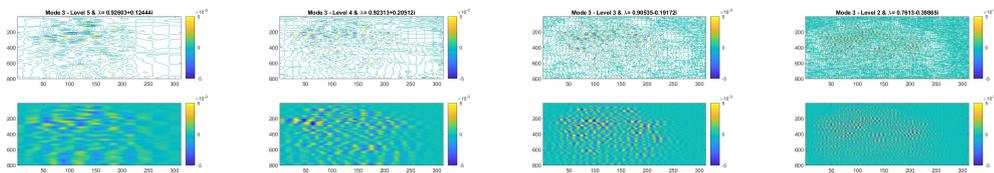


The following images present the performance of the Wavelet-DMD algorithm over the same example for 5 scales of decomposition. Therefore 12 modes were selected for the Wavelet-DMD approach. As discussed in Chapter 5, the wavelet-DMD manages to reveal structures where the alternative

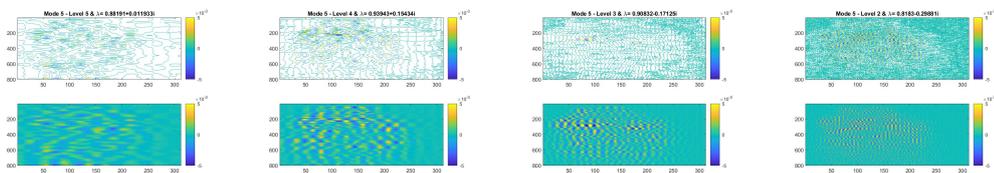
approaches cannot identify even by using a substantially smaller amount of modes.



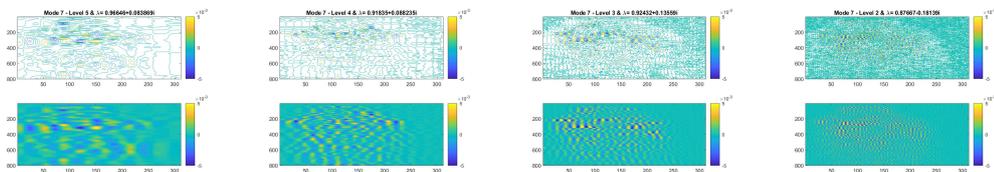
(a) Wavelet-DMD 1<sup>st</sup> Mode Scale-5 (b) Wavelet-DMD 1<sup>st</sup> Mode Scale-4 (c) Wavelet-DMD 1<sup>st</sup> Mode Scale-3 (d) Wavelet-DMD 1<sup>st</sup> Mode Scale-2



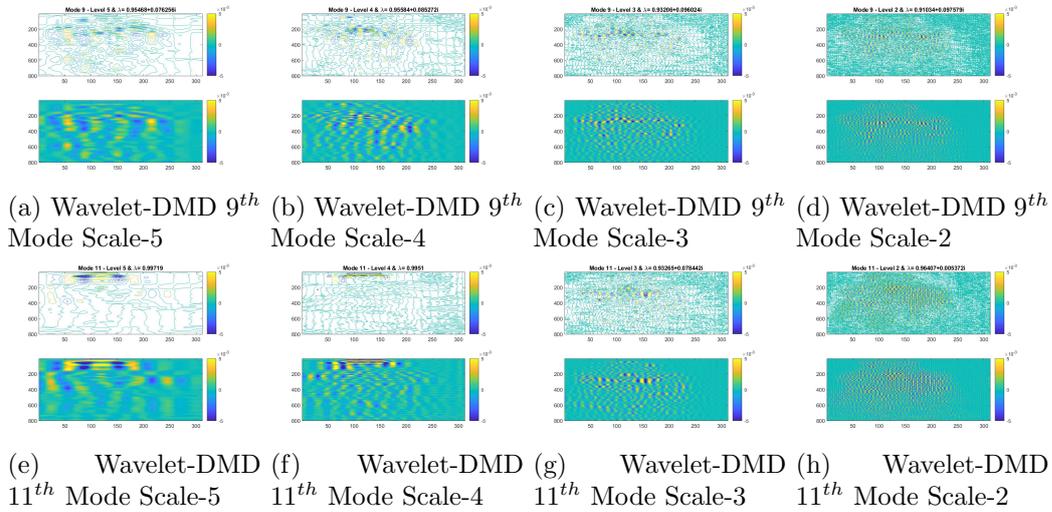
(a) Wavelet-DMD 3<sup>rd</sup> Mode Scale-5 (b) Wavelet-DMD 3<sup>rd</sup> Mode Scale-4 (c) Wavelet-DMD 3<sup>rd</sup> Mode Scale-3 (d) Wavelet-DMD 3<sup>rd</sup> Mode Scale-2



(a) Wavelet-DMD 5<sup>th</sup> Mode Scale-5 (b) Wavelet-DMD 5<sup>th</sup> Mode Scale-4 (c) Wavelet-DMD 5<sup>th</sup> Mode Scale-3 (d) Wavelet-DMD 5<sup>th</sup> Mode Scale-2

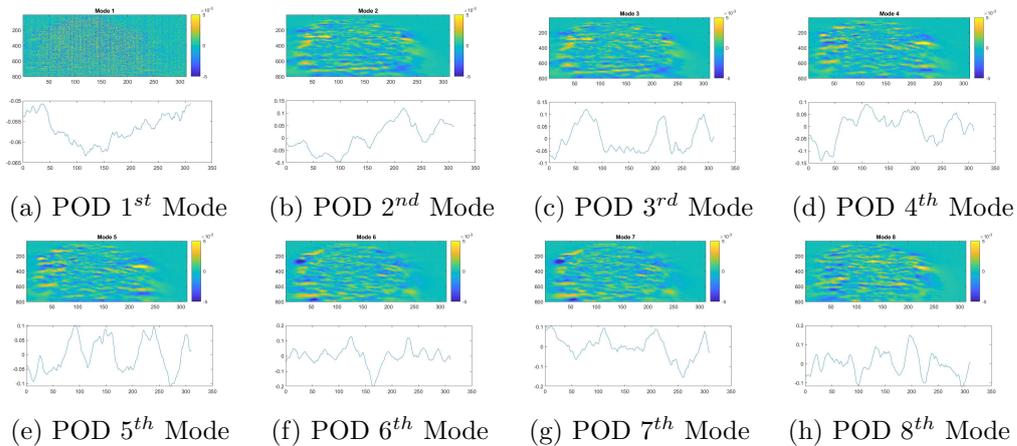


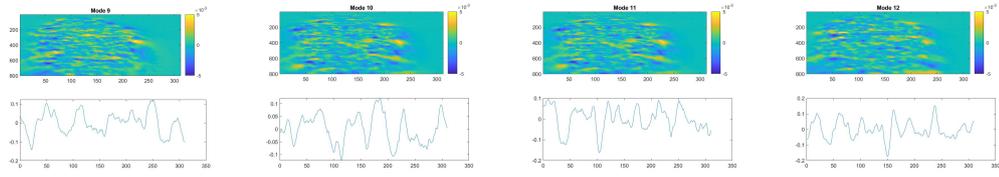
(a) Wavelet-DMD 7<sup>th</sup> Mode Scale-5 (b) Wavelet-DMD 7<sup>th</sup> Mode Scale-4 (c) Wavelet-DMD 7<sup>th</sup> Mode Scale-3 (d) Wavelet-DMD 7<sup>th</sup> Mode Scale-2



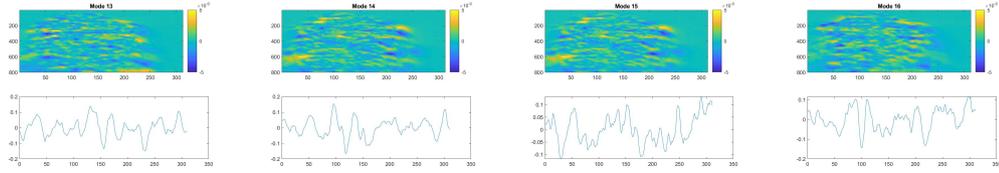
By observing these modes, it can be seen that the first DMD modes in the first scales separating the noise from the actual dynamic effects and as the scales are increased, different patterns are revealed. Special attention should be given in the last modes, where patterns that were not revealed in any DMD or POD approach are reveal. These patterns are indicating where the particle are concentrated that could possibly lead into turbulences to the manufacturing process.

As presented in the previous example, the standard Hankel POD and the Wavelet-POD are compared in this occasion too. Firstly, the POD modes are demonstrated. It can be observed that the POD is unable to reveal the possible areas where there is particles concentration in contrast to the previous example.





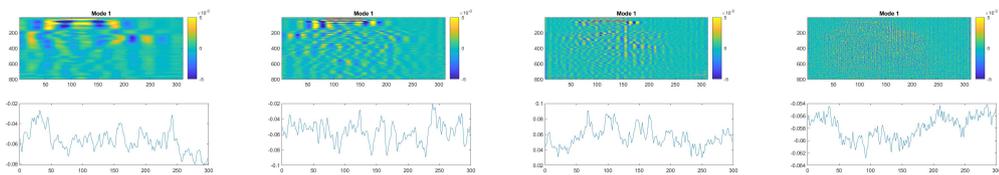
(a) POD 9<sup>th</sup> Mode (b) POD 10<sup>th</sup> Mode (c) POD 11<sup>th</sup> Mode (d) POD 12<sup>th</sup> Mode



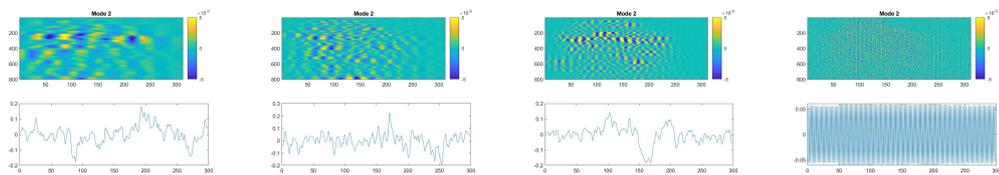
(e) POD 13<sup>th</sup> Mode (f) POD 14<sup>th</sup> Mode (g) POD 15<sup>th</sup> Mode (h) POD 16<sup>th</sup> Mode

The standard Hankel-POD is unable to reveal in this complex real time example the areas of particles concentration, as followed by the standard Hankel-DMD. It is interesting to exhibit the consistency of the wavelet based approaches The wavelet-POD is presented below.

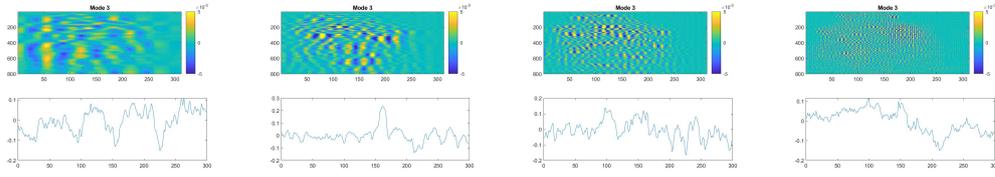
The modes from the reconstructed images from the detail coefficients in each scale are presented. In this example, the level of depth of the wavelet decomposition was selected to be 5 in order to reveal as many details as possible.



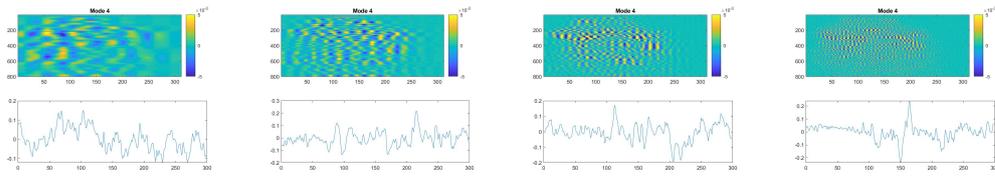
(a) Wavelet-POD 1<sup>st</sup> Mode Scale-5 (b) Wavelet-POD 1<sup>st</sup> Mode Scale-4 (c) Wavelet-POD 1<sup>st</sup> Mode Scale-3 (d) Wavelet-POD 1<sup>st</sup> Mode Scale-2



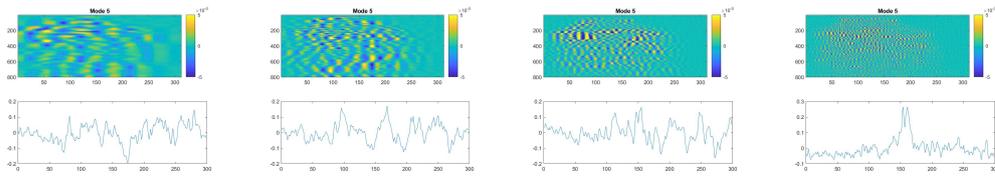
(a) Wavelet-POD 2<sup>nd</sup> Mode Scale-5 (b) Wavelet-POD 2<sup>nd</sup> Mode Scale-4 (c) Wavelet-POD 2<sup>nd</sup> Mode Scale-3 (d) Wavelet-POD 2<sup>nd</sup> Mode Scale-2



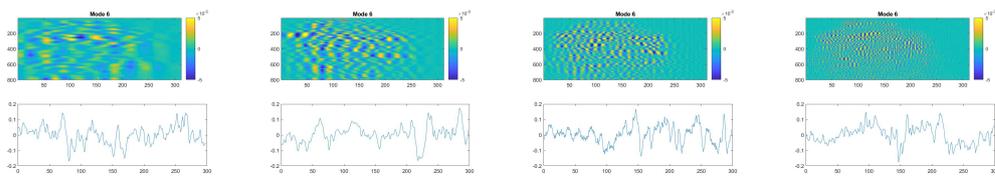
(a) Wavelet-POD 3<sup>rd</sup> Mode Scale-5 (b) Wavelet-POD 3<sup>rd</sup> Mode Scale-4 (c) Wavelet-POD 3<sup>rd</sup> Mode Scale-3 (d) Wavelet-POD 3<sup>rd</sup> Mode Scale-2



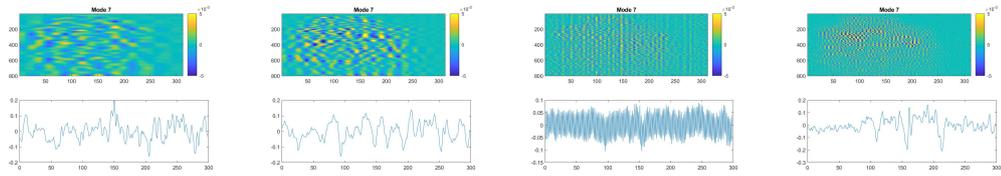
(a) Wavelet-POD 4<sup>th</sup> Mode Scale-5 (b) Wavelet-POD 4<sup>th</sup> Mode Scale-4 (c) Wavelet-POD 4<sup>th</sup> Mode Scale-3 (d) Wavelet-POD 4<sup>th</sup> Mode Scale-2



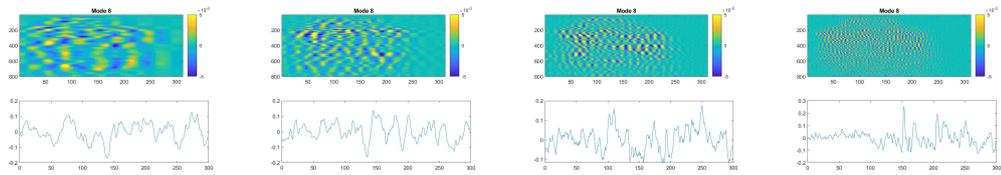
(a) Wavelet-POD 5<sup>th</sup> Mode Scale-5 (b) Wavelet-POD 5<sup>th</sup> Mode Scale-4 (c) Wavelet-POD 5<sup>th</sup> Mode Scale-3 (d) Wavelet-POD 5<sup>th</sup> Mode Scale-2



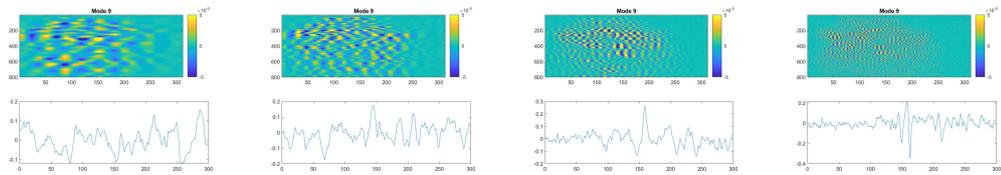
(a) Wavelet-POD 6<sup>th</sup> Mode Scale-5 (b) Wavelet-POD 6<sup>th</sup> Mode Scale-4 (c) Wavelet-POD 6<sup>th</sup> Mode Scale-3 (d) Wavelet-POD 6<sup>th</sup> Mode Scale-2



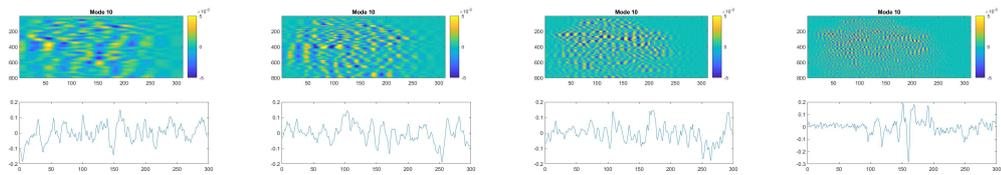
(a) Wavelet-POD 7<sup>th</sup> Mode Scale-5 (b) Wavelet-POD 7<sup>th</sup> Mode Scale-4 (c) Wavelet-POD 7<sup>th</sup> Mode Scale-3 (d) Wavelet-POD 7<sup>th</sup> Mode Scale-2



(a) Wavelet-POD 8<sup>th</sup> Mode Scale-5 (b) Wavelet-POD 8<sup>th</sup> Mode Scale-4 (c) Wavelet-POD 8<sup>th</sup> Mode Scale-3 (d) Wavelet-POD 8<sup>th</sup> Mode Scale-2

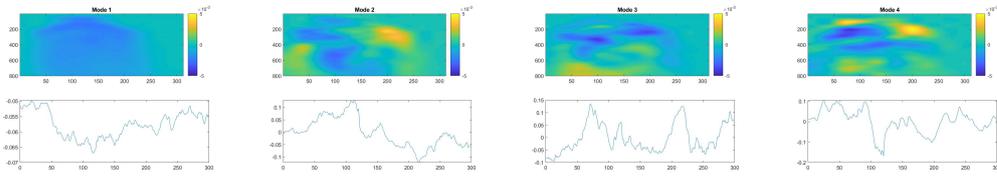


(a) Wavelet-POD 9<sup>th</sup> Mode Scale-5 (b) Wavelet-POD 9<sup>th</sup> Mode Scale-4 (c) Wavelet-POD 9<sup>th</sup> Mode Scale-3 (d) Wavelet-POD 9<sup>th</sup> Mode Scale-2

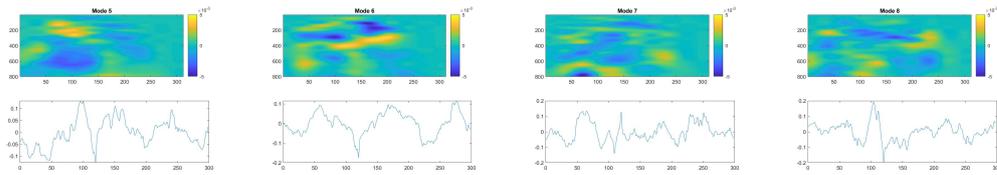


(a) Wavelet-POD 10<sup>th</sup> Mode Scale-5 (b) Wavelet-POD 10<sup>th</sup> Mode Scale-4 (c) Wavelet-POD 10<sup>th</sup> Mode Scale-3 (d) Wavelet-POD 10<sup>th</sup> Mode Scale-2

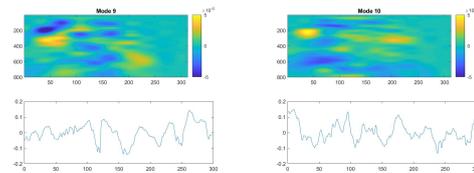
Additionally, the modes from the reconstructed images from the approximation coefficients in the coarsest scale are presented.



(a) Wavelet-POD 1<sup>st</sup> Mode (b) Wavelet-POD 2<sup>nd</sup> Mode (c) Wavelet-POD 3<sup>rd</sup> Mode (d) Wavelet-POD 4<sup>th</sup> Mode



(a) Wavelet-POD 5<sup>th</sup> Mode (b) Wavelet-POD 6<sup>th</sup> Mode (c) Wavelet-POD 7<sup>th</sup> Mode (d) Wavelet-POD 8<sup>th</sup> Mode



(a) Wavelet-POD 9<sup>th</sup> Mode (b) Wavelet-POD 10<sup>th</sup> Mode

In conclusion, the proposed wavelet-based reduced order methods are capable of revealing hidden structures and system characteristics that cannot be revealed with the standard decomposition methods. All these approaches were tested on models with multiscale characteristics, abrupt changes and linear dependencies and managed to exhibit the system dynamics successfully.



# Chapter 7

## Conclusions & Future Work

### 7.1 Summary of Contributions

The thesis investigated temporal and spatial multiscale models where the catholic understanding of the underlying dynamics of these complex systems is of main importance in many fields. Data-driven models offer flexibility in the field of system identification due to the fact that they make use of raw dataset and do not require any, in advance, speculation about the system dynamics. The ultimate goal of this thesis was to address the gaps of the data-driven identification methods in the multiscale world and propose alternative solutions for improving precision and accuracy.

In the field of temporal systems, the performance of the  $\delta$ -operator was introduced in the SID algorithms. The aim was to improve the performance of the SID methods over fast-sampled single-scale and multiscale models due to singularities that can emerge. Moreover, the use of the  $\delta$ -operator can link the discrete SID method with its continuous counterpart for better interpretability.

In the field of spatio-temporal systems, a comprehensive analysis and investigation of the performance boundaries for the existing reduced order identification algorithms was addressed. Up to date, there was no detailed analysis of the various POD and DMD algorithms over signals with abrupt changes and linear dependencies. Moreover, the performance of these approaches over systems with multiscale and mixed frequency characteristics is considered. Their limitations are revealed, unfolding the necessity of new multiscale approaches that could overcome the boundaries discussed.

All previous led to the development of a novel spatial multiscale DMD algorithm. The novel multiscale POD and DMD approaches were tailored to resolve the problems of the existing ROM methods. The novel algorithm

combines the properties of the wavelet decomposition and the higher order DMD algorithms. The wavelet-DMD algorithm is proven to preserve the eigenvalues of any conventional DMD algorithm, hence it can substitute them without loss of information. The performance of the proposed algorithm was evaluated for complex multiscale models with mixed frequency and multiscale dynamics. The results showed that the wavelet DMD is outperforming all existing identification methods in terms of separating and revealing structures that give insight into the system behaviour. At the same time, the wavelet-DMD introduces a multiscale model that can accentuate different dynamic effects simultaneously.

In chapter 3, in the field of temporal series of data, a detailed analysis of the SID methods was addressed where a new SID algorithm was proposed for high sampled data. The SID methods saw excellent performance at identifying accurately a system when it comes to noise-free measurements. The new algorithm based on the  $\delta$ -operator was introduced for high sampled datasets. Moreover, it was introduced due to the fact that till now, all SID methods are based on discrete model datasets. Hence, a model that could link and interpret the discrete models to their continuous counterpart had to be addressed. The results and comparisons, though, to the existing methods did not show an improvement in the SID algorithm accuracy and efficiency. On the other hand, all methods have shown weaknesses when it comes to noisy measurements, especially for occasions where the SNR was very low.

## 7.2 Future Work

The PhD investigated identification of systems that showed both temporal and spatial characteristics. The thesis made some contributions towards advancing the analysis in this area. However, this work can be extended further in a number of different ways.

In the case of the SID methods, it was shown that no SID method manages to deal with the noisy measurements when introduced to the system. Following this, a preprocessing of the input data using Kalman filters could be introduced and analysed aiming at more accurate representations when it comes to noisy datasets. A recent study based on the wavelet decomposition as a preprocessing step showed promise. Hence, it would be of great interest the use of other filtering methods, such as Kalman filters or Multi-scale Kalman filters combined with the SID and investigate whether it offers performance gains.

In the case of the wavelet DMD, new criteria for the number of the selected modes can be exploited. The DMD performance is indissolubly linked to the

---

number of selected modes in the SVD step. Up to now, the amount of modes is selected empirically only by observing the singular values plot of the SVD or MSE criteria. System identification literature has developed model selection methods that trade-off reconstruction error against complexity. Such an approach can be adapted to the selection of the number of DMD modes.



# Appendices



# Appendix A

## Orthogonality Properties

**Proof No 1:** Based on the multiresolution properties,, it will be proven that the set  $\{\psi_{0k} = \psi(x - k), k \in \mathbf{Z}\}$  is orthonormal. For any  $k, l \in \mathbf{Z}$ , the following inner product is computed:

$$\langle \psi_k, \psi_l \rangle = \langle \psi(x - k), \psi(x - l) \rangle \quad (\text{A.1})$$

Combining equation G.8 and using G.9:

$$\psi(x - k) = \sum_{m \in \mathbf{Z}} g_m \sqrt{2} \phi(2x - 2k - m). \quad (\text{A.2})$$

Substitution into A.1 yields

$$\begin{aligned} \langle \psi(x - k), \psi(x - l) \rangle &= \left\langle \sum_{m \in \mathbf{Z}} g_m \sqrt{2} \phi(2x - 2k - m), \sum_{n \in \mathbf{Z}} g_n \sqrt{2} \phi(2x - 2l - n) \right\rangle \\ &= \sum_{m \in \mathbf{Z}} g_m g_{m+2k-2l} \quad (2k + m = 2l + n \Rightarrow n = m + 2k - 2l) \\ &= \sum_{m \in \mathbf{Z}} (-1)^m h_{1-m} (-1)^m h_{1-m-2k+2l} \\ &= \sum_{m \in \mathbf{Z}} h_{1-m} h_{1-m-2k+2l} \end{aligned} \quad (\text{A.3})$$

The above expression is zero for  $k \neq l$  and 1 for  $k = l$  due to the corresponding orthogonality relations for the translated scaling functions.

$$\begin{aligned}
\langle \phi(x-k), \phi(x-l) \rangle &= \left\langle \sum_{m \in \mathbf{Z}} h_m \sqrt{2} \phi(2x-2k-m), \sum_{n \in \mathbf{Z}} h_n \sqrt{2} \phi(2x-2l-n) \right\rangle \\
&= \sum_{m \in \mathbf{Z}} h_m h_{m+2k-2l} \quad (2k+m=2l+n \Rightarrow n=m+2k-2l) \\
&= \delta_{kl}
\end{aligned} \tag{A.4}$$

Replacement of  $m$  with  $1-m$  and substitution into A.3 yields the desired result

$$\langle \psi(x-k), \psi(x-l) \rangle = \delta_{kl} .$$

**Proof No 2:** It will be proven that the set  $\{\psi_{0k}(x) = \psi(x-k), k \in \mathbf{Z}\}$  is orthogonal to  $V_0$ . It is sufficient to show that

$$\langle \phi(x-k), \psi(x-l) \rangle = 0.$$

From the scaling results,

$$\begin{aligned}
\langle \phi(x-k), \psi(x-l) \rangle &= \left\langle \sum_{m \in \mathbf{Z}} h_m \sqrt{2} \phi(2t-2k-m), \sum_{n \in \mathbf{Z}} g_n \sqrt{2} \phi(2t-2l-n) \right\rangle \\
&= \sum_{m \in \mathbf{Z}} h_m g_{m+2k-2l} \quad (2k+m=2l+n \Rightarrow n=m+2k-2l) \\
&= \sum_{m \in \mathbf{Z}} (-1)^{m+2k-2l} h_m h_{1-m-2k+2l} \\
&= \sum_{m \in \mathbf{Z}} (-1)^m h_m h_{1-m-2k+2l}.
\end{aligned} \tag{A.5}$$

It has to be shown that:

$$\sum_{m \in \mathbf{Z}} (-1)^m h_m h_{1-m+2p} = 0, \quad \text{for any } p \in \mathbf{Z}$$

It is shown from equation A.4 that is true for value of  $p$  is equal to  $p = 0$  : The series becomes a "cancelling sum" about the terms  $h_0 h_1$  and  $h_1 h_0$ . Recall that this was the basis of the definition of the  $g_k$  in terms of the  $h_k$ . But for a shift of  $2p$  in one of the arguments, it is also a "cancelling sum" about  $m = p$  and  $m = p + 1$  :

$$\begin{aligned} \sum_{m \in \mathbf{Z}} (-1)^m h_m h_{1-m+2p} &= \cdots + (-1)^p h_p h_{1+p} + (-1)^{p+1} h_{p+1} h_p + \cdots \\ &= 0. \end{aligned} \quad (\text{A.6})$$

Since  $-2k + 2l$  is an even number, it follows that

$$\langle \phi(x - k), \psi(x - l) \rangle = 0.$$

**Proof No 3:** Any element  $y \in W_0$  admits an expansion in the functions  $\psi(x - k)$ . It will be shown that the space  $V_1$  is spanned by integer translates of  $\phi$  and corresponding translates of  $\psi$ .

Recall the fact that the functions  $\phi_{1k} = 2^{1/2}\phi(2x - k)$  span  $V_1$ . It ll be shown that for each  $j$ ,

$$\phi(2x - j) = \sum_k a_k \phi(x - k) + b_k \psi(x - k) \quad (\text{A.7})$$

for an appropriate set of constants  $a_k$  and  $b_k$ . From the orthogonality of the  $\phi(x - k)$ , it follows that:

$$\begin{aligned} a_k &= \langle \phi(2x - j), \phi(x - k) \rangle \\ &= \left\langle \phi(2x - j), \sum_l h_l \sqrt{2} \phi(2x - 2k - l) \right\rangle \\ &= \frac{1}{\sqrt{2}} h_{j-2k} \quad (j = 2k + l \Rightarrow l = j - 2k) \end{aligned} \quad (\text{A.8})$$

Likewise, we find that

$$\begin{aligned} b_k &= \langle \phi(2x - j), \psi(x - k) \rangle \\ &= \left\langle \phi(2x - j), \sum_l g_l \sqrt{2} \phi(2x - 2k - l) \right\rangle \\ &= \frac{1}{\sqrt{2}} g_{j-2k} \\ &= \frac{1}{\sqrt{2}} (-1)^j h_{1-j+2k} \end{aligned} \quad (\text{A.9})$$

---

If the equality in A.7 holds, in the  $L^2$ -sense, then the following result must hold:

$$\langle \phi(2x - j), \phi(2x - j) \rangle = \frac{1}{2} = \sum_k [|a_k|^2 + |b_k|^2].$$

But from the scaling equation G.5:

$$\langle \phi(x), \phi(x) \rangle = \sum_k |h_k|^2 = 1 \quad (\text{A.10})$$

In summary, any element  $u \in V_1$  admits a unique expansion in terms of the functions  $\phi(2x - j)$  which, in turn, admit unique expansions in terms of the  $\phi(x - k)$  and  $\psi(x - k)$  functions. Since the  $\phi(x - k)$  span  $V_0$ , it follows that the  $\psi(x - k)$  span  $W_0$ .

# Appendix B

## Kalman Filtering & Kalman Smoothing

The Kalman filter is an optimal estimation algorithm for stochastic state space estimation from noisy sensor measurements with broad applications in the field of signal processing and control. Kalman filter is predictor-corrector estimation algorithm that creates a cost-loss function that minimizes the estimated error covariance in an optimal way. Objective of the Kalman filter algorithm is to minimise the mean squared error between the actual and estimated data. The term optimal means that all noise is Gaussian, thus the Kalman filter minimises the mean square error of the estimated parameters. In occasions where the noise does not present these characteristics, other non-linear estimators may be a better approach, such as the Extended Kalman Filter (EKF) or the Unscented Kalman Filter (UKF) which will not be analysed and used for the purposes of this thesis. [21, 22, 68]

The Discrete Kalman filter (DKF) will be demonstrated due to its use in the next chapter for the system identification purposes. The DKF estimates the state  $x \in \mathfrak{R}^n$  of a discrete-time controlled process that is governed by the linear stochastic difference equation:

$$x_k = Ax_{k-1} + Bu_k + w_{k-1} \quad (\text{B.1})$$

with a measurement  $y \in \mathfrak{R}^m$  that is

$$y_k = Cx_k + v_k \quad (\text{B.2})$$

where  $w_k$  and  $v_k$  represent the input and measurement noise respectively. They are independent from each other and present normal probability distributions.

$$\begin{aligned} p(w_k) &\sim N(0, Q), \\ p(v_k) &\sim N(0, R) \end{aligned} \quad (\text{B.3})$$

where  $Q$  is the process noise covariance and  $R$  is the measurement noise covariance.

Kalman filtering requires a derived state space model based on the a priori state  $\hat{x}_k^- \in \mathfrak{R}^n$  knowledge and obtains the optimal state by minimizing the error:

$$\begin{aligned} e_k^- &\equiv x_k - \hat{x}_k^-, \text{ and} \\ e_k &\equiv x_k - \hat{x}_k \end{aligned} \quad (\text{B.4})$$

where  $\hat{x}_k \in \mathfrak{R}^n$  is the posteriori state.

The a priori estimate error covariance is then

$$P_k^- = E [e_k^- e_k^{-T}] \quad (\text{B.5})$$

and the a posteriori estimate error covariance is

$$P_k = E [e_k e_k^T] = E [(x_k - \hat{x}_k) (x_k - \hat{x}_k)^T] \quad (\text{B.6})$$

An update equation for the new estimate, combining the old estimate with measurement data is given by:

$$\hat{x}_k = \hat{x}_k^- + K_k (y_k - C\hat{x}_k^-) \quad (\text{B.7})$$

where  $K_k$  is the Kalman gain, which will be derived in the next step. The term  $y_k - C\hat{x}_k^-$  is known as the innovation or measurement residual:

$$i_k = y_k - C\hat{x}_k \quad (\text{B.8})$$

Substituting equation B.2 into equation B.7 gives:

$$\hat{x}_k = \hat{x}_k^- + K_k (Cx_k + v_k - C\hat{x}_k^-) \quad (\text{B.9})$$

Substituting equation B.9 into equation B.6, the expectation equation is given by:

$$P_k = E \left[ [(I - K_k C) (x_k - \hat{x}'_k) - K_k v_k] \right. \\ \left. [(I - K_k C) (x_k - \hat{x}'_k) - K_k v_k]^T \right] \quad (\text{B.10})$$

The term  $(x_k - \hat{x}'_k)$  indicates the error of the prior estimate. There is no correlation of the error with the measurement noise, therefore the expectation is expressed as follows:

$$P_k = (I - K_k C) E \left[ (x_k - \hat{x}'_k) (x_k - \hat{x}'_k)^T \right] (I - K_k C) \\ + K_k E [v_k v_k^T] K_k^T \quad (\text{B.11})$$

The expectation  $P'_k$  can be expressed as  $P'_k = E \left[ (x_k - \hat{x}'_k) (x_k - \hat{x}'_k)^T \right]$  and the  $E [v_k v_k^T]$  is the measurement noise covariance  $R$ . Therefore:

$$P_k = (I - K_k C) P'_k (I - K_k C)^T + K_k R K_k^T \\ P_k = P_k^- - K_k C P_k^- - P_k^- C^T K_k^T + K_k (C P_k^- C^T + R) K_k^T \quad (\text{B.12})$$

The trace of a matrix is equal to the trace of its transpose, therefore it may written as:

$$T [P_k] = T [P_k^-] - 2T [K_k C P_k^-] + T [K_k (C P_k^- C^T + R) K_k^T] \quad (\text{B.13})$$

where  $T [P_k]$  is the trace of the matrix  $P_k$ .  
Differentiating with respect to  $K_k$  gives:

$$\frac{dT [P_k]}{dK_k} = -2 (C P_k^-)^T + 2K_k (C P_k^- C^T + R) \quad (\text{B.14})$$

The optimal solution is given by setting  $\frac{dT [P_k]}{dK_k} = 0$ . Therefore:

$$(C P_k^-)^T = K_k (C P_k^- C^T + R) \quad (\text{B.15})$$

Solving this equation for  $K_k$ , it gives:

$$K_k = P_k^- C^T (C P_k^- C^T + R)^{-1} \quad (\text{B.16})$$

This last equation represents the Kalman gain equation. The innovation,  $i_k$  has an associated measurement prediction covariance. This is given by:

$$S_k = CP_k^- C^T + R \quad (\text{B.17})$$

Finally, substituting the last equation into equation B.12 gives:

$$\begin{aligned} P_k &= P_k^- - P_k^- C^T (CP_k^- C^T + R)^{-1} CP_k^- \\ &= P_k^- - K_k CP_k^- \\ &= (I - K_k C) P_k^- \end{aligned} \quad (\text{B.18})$$

The equation B.18 represents the update equation for the error covariance matrix with optimal gain. The three equations B.4, B.7 and B.9 develop an estimate of the variable  $x_k$ . The state projection is achieved by:

$$\hat{x}_{k+1}^- = A' \hat{x}_k \quad (\text{B.19})$$

In order for the recursion step to be completed, an equation that projects the error covariance matrix into the next time interval,  $k + 1$  needs to be obtained. This is achieved by first expressing an equation for the prior error:

$$e_{k+1}^- = x_{k+1} - \hat{x}_{k+1}^- \quad (\text{B.20})$$

$$= (A' x_k + w_k) - A' \hat{x}_k \quad (\text{B.21})$$

$$= A' e_k + w_k \quad (\text{B.22})$$

Extending equation B.6 to time-step  $k + 1$ , gives:

$$P_{k+1}^- = E [e_{k+1}^- e_{k+1}^{T-}] = E [(A' e_k + w_k) (A' e_k + w_k)^T] \quad (\text{B.23})$$

The error  $e_k$  and input noise  $w_k$  have zero cross-correlation because the noise  $w_k$  actually accumulates between  $k$  and  $k + 1$  whereas the error  $e_k$  is the error up until time  $k$ . Therefore:

$$\begin{aligned} P_{k+1}^- &= E [e_{k+1}^- e_{k+1}^{T-}] \\ &= E [A' e_k (A' e_k)^T] + E [w_k w_k^T] \\ &= A' P_k A'^T + Q \end{aligned} \quad (\text{B.24})$$

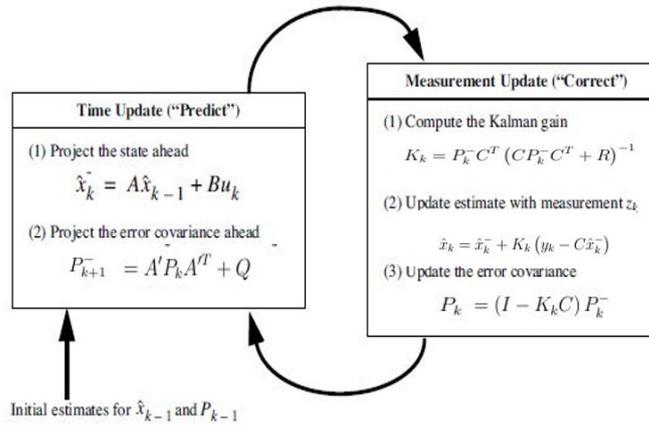


Figure B.1: A schematic presenting of the Kalman filter steps

This is the recursive Kalman filter. The algorithm is repeated till the error tends to reach very small values. The following diagram summarizes the Kalman filter algorithm steps.

An extension of the traditional Kalman filtering can be given by the Kalman smoother. The smoothed version improves the standard one, by adding future measurements instead of only the a priori data. Kalman smoother is an offline estimation algorithm, that improves the Kalman filter performance when it comes to estimate the sequence of states and is based on backwards recursion.

There are three types of Kalman smoothing algorithms :

- Fixed-lag smoothing: Estimate  $\hat{x}_{j|j+\ell_0}$ , for  $j = 0, 1, \dots$
- Fixed-point smoothing: Estimate  $\hat{x}_{j_0|k}$ , for  $k = j_0, j_0 + 1, \dots$
- Fixed-interval smoothing: Estimate  $\hat{x}_{j|k_0}$ , for  $j = 0, \dots, k_0$

The most common type is the fixed-interval smoothing. There are two types of smoothing algorithms in this category known as:

- Two-pass smoother or RTS (Rauch-Tung-Striebel) smoother where the standard Kalman filter is computed in a forward pass, and the smoothed values are computed in a backward pass.

- Forward-backward smoother where the smoother combines two estimates of the state, one given the past and the other given the future

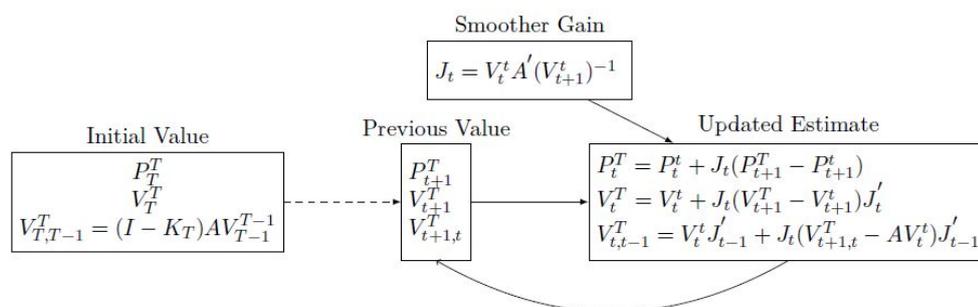


Figure B.2: A schematic presenting the Kalman smoothing steps

The Kalman filter is used in system identification methods such as the Subspace identification method in order to derive the uncertainties when they are present. On the other hand, all data-driven models do not require making any priori assumptions about the model, for deterministic and stochastic systems. Hence, non of these iterative methods are applied on them. The aim is to overcome the difficulties and ill-conditioned situations of the above iterative methods by introducing computationally efficient and non-iterative algorithms. Extensions of these approaches will be applied for data with multiscale characteristics. Hence, non-iterative reduced order models will be introduced in order to overcome these obstacles.

# Appendix C

## Parameter Estimation

### C.1 Autoregressive Model with exogenous input (ARX)

The commonly used model structure of a linear time-invariant deterministic or stochastic system is the simple linear input-output difference equation called Autoregressive Model with exogenous input (ARX). ARX structure is simple and can capture the linear relationship in different real world problems but implementing simple L2-norm related solutions [39,63]. The ARX model is a linear difference equation formed as follows:

$$y(t) + a_1y(t-1) + \dots + a_{n_a}y(t-n_a) = b_0u(t-d) + \dots + b_{n_b}u(t-d-n_b) + e(t) \quad (\text{C.1})$$

which relates the current output  $y(t)$  to a finite number of past outputs  $y(t-d)$  and inputs  $u(t-d)$ .  $e(t)$  is a random white-noise which indicates the error in the difference equation.

The aim is to estimate the adjustable model parameters  $a_i$  and  $b_i$  which can be summarized as:

$$\theta = [ a_1 \ a_2 \ \dots \ a_{n_a} \ b_0 \ b_2 \ \dots \ b_{n_b} ] \quad (\text{C.2})$$

Similarly, the input-output data can be written as:

$$\varphi_t^T = (-y_{t-1}, \dots, -y_{t-n_a}, u_{t-1}, \dots, u_{t-n_b})^T \quad (\text{C.3})$$

In the special case where the  $n_a = 0$ , the model is no longer called ARX but finite impulse response (FIR) model. Then, equation C.1 can be written as:

$$y(t) = \varphi_t^T(t)\theta + e(t) \tag{C.4}$$

As a summary, the input-output structure of the ARX model can be given in a transfer function form as:

$$y(t) = \frac{B(q^{-1})}{A(q^{-1})}u(t) + \frac{1}{A(q^{-1})}e(t) \tag{C.5}$$

where

$$A(q^{-1}) = [ 1 \quad a_1 \quad a_2 \quad a_3 \quad \dots \quad a_n ]$$

$$B(q^{-1}) = [ 1 \quad b_1 \quad b_2 \quad b_3 \quad \dots \quad b_n ]$$

The ARX models can be estimated using prediction error methods by solving a linear regression problem. The disadvantage of the ARX model is that disturbances are part of the system dynamics. The transfer function of the input part ( deterministic ) and the transfer function of the disturbance share the same poles which is unrealistic [39, 63].

## C.2 Autoregressive moving-average with exogenous input (ARMAX)

A more flexible parameter estimation model is the autoregressive moving-average with exogenous input (ARMAX) models. Unlike the ARX model, the ARMAX model structure introduces an extra term called disturbance dynamics. The advantage of the ARMAX over the ARX is the flexibility of handling the disturbances, when applied on a system [39, 63].

The ARMAX model is described by the following relationship between the input-output data:

$$\begin{aligned} & y(t) + a_1y(t-1) + \dots + a_{n_a}y(t-n_a) \\ & = b_0u(t-d) + \dots + b_{n_b}u(t-d-n_b) \\ & + e(t) + c_1e(t-1) + \dots + c_{n_e}e(t-n_e) \end{aligned} \tag{C.1}$$

where the unknown adjustable variables are expressed as:

$$\theta = [ a_1 \ a_2 \ \dots \ a_{n_a} \ b_0 \ b_1 \ \dots \ b_{n_b} \ c_1 \ c_2 \ \dots \ c_{n_c} ]^T \quad (\text{C.2})$$

and the input-output data as:

$$\varphi_t^T = (-y_{t-1}, \dots, -y_{t-n_a}, u_{t-1}, \dots, u_{t-n_b})^T \quad (\text{C.3})$$

Hence the system in either the ARX model or the ARMAX model can be expressed in the same way but the ARMAX model contains the disturbance parameters:

$$y_t = \varphi_t^T \theta + e_t, \forall t = \dots, 0, 1, 2, \dots \quad (\text{C.4})$$

The general ARMAX model is given by the transfer function equation :

$$y(t) = \frac{B(q^{-1})}{A(q^{-1})}u(t) + \frac{C(q^{-1})}{A(q^{-1})}e(t) \quad (\text{C.5})$$

where

$$A(q^{-1}) = [ 1 \ a_1 \ a_2 \ a_3 \ \dots \ a_n ]$$

$$B(q^{-1}) = [ 1 \ b_1 \ b_2 \ b_3 \ \dots \ b_n ]$$

$$C(q^{-1}) = [ 1 \ c_1 \ c_2 \ c_3 \ \dots \ c_n ]$$

The ARMAX model is suitable in areas of control, processes and econometrics for both system modelling and control scheme design. The input-output equation model structure can equivalently be given by:

The existence of the numerator  $C(q, \theta)$  in the ARMAX model introduces some independence in the parametrization, but similarly to the ARX model, the noise sequence  $\{e_t\}$  is still filtered by the same denominator dynamics as the input, having the same poles.

## C.3 Box-Jenkins (BJ)

An even more flexible model description compared to the ARX and ARMAX models is given by Box-Jenkins structure. BJ models introduce models where the disturbance properties are modelled separately from system dynamics, as opposed in ARMAX models [39, 63].

The general BJ model is given by the transfer function equation:

$$G(q, \theta) = \frac{B(q, \theta)}{A(q, \theta)}, \quad H(q, \theta) = \frac{C(q, \theta)}{D(q, \theta)},$$

where

$$\begin{aligned} A(q, \theta) &= 1 + a_1 q^{-1} + \dots + a_{n_a} q^{-n_a}, \\ B(q, \theta) &= b_1 q^{-1} + \dots + b_{n_b} q^{-n_b}, \\ C(q, \theta) &= 1 + c_1 q^{-1} + \dots + c_{n_c} q^{-n_c}, \\ D(q, \theta) &= 1 + d_1 q^{-1} + \dots + d_{n_d} q^{-n_d} \end{aligned}$$

Here, the parameter vector to be estimated is written as:

$$\theta = \begin{bmatrix} a_1 & \dots & a_{n_a} & b_1 & \dots & b_{n_b} \\ c_1 & \dots & c_{n_c} & d_1 & \dots & d_{n_d} \end{bmatrix}^T \in \mathbb{R}^{n_a+n_b+n_c+n_d}.$$

This model is called Box-Jenkins (BJ), as it was proposed by Box and Jenkins in [26]. Its main advantage is the completely independent parametrization of  $G(q, \theta)$  and  $H(q, \theta)$ , both parametrized with numerator and denominator polynomials. It is, thus, a quite general and flexible parametrization.

In the case where the additive output disturbance is white noise, then the  $n_d = 0 = n_c$ . In terms of  $G(q, \theta)$  and  $H(q, \theta)$ , it gives that:

$$G(q, \theta) = \frac{B(q, \theta)}{A(q, \theta)}, \quad H(q, \theta) = 1.$$

This is called an output-error (OE) model.

All parametric models can be described either in continuous or discrete time. The equivalent discrete time model of the ARX and ARMAX model equation C.4 can be described as follows:

$$y[k] = \varphi^T[k] \theta + e[k] \tag{C.1}$$

where in discrete time, the input-output data are written as

$$\varphi[k] = \begin{bmatrix} y[k-1] & \dots & y[k-n] & u[k] & u[k-1] & \dots & u[k-m] \end{bmatrix}^T \in \mathbb{R}^{n+m \times 1} \tag{C.2}$$

and the unknown adjustable variables for the ARX model as:

$$\theta = [ a_1 \ \cdots \ a_n \ b_0 \ b_1 \ \cdots \ b_m ]^T \in \mathbb{R}^{n+m+1} \quad (\text{C.3})$$

or

$$\theta = [ a_1 \ \cdots \ a_n \ b_0 \ b_1 \ \cdots \ b_m \ c_0 \ c_1 \ \cdots \ c_m ]^T \in \mathbb{R}^{n+m+1} \quad (\text{C.4})$$

for the ARMAX model.

The parameter  $e[k]$  indicates the usual zero-mean Gaussian white noise process.

$$\mathbf{e} = [e[k_0] \ e[k_0 + 1] \ \cdots \ e[k_0 + M - 1]]^T \quad (\text{C.5})$$



# Appendix D

## SVD Geometrical Derivation

Geometrically, SVD represents the decomposition of a given vector onto two orthogonal axis. For instance, the unit vectors ( $u_1$  and  $u_2$ ) represent the directions of the vectors. The lengths of projection ( the line segments  $s_{a_1}$  and  $s_{a_2}$  ) indicate the amount of the vector contained in each direction of projection. The vectors of projection ( $p_{a_1}$  and  $p_{a_2}$ ) are used to reconstruct the original vector  $a$  by adding them together ( as a vector sum ). It is obvious to verify that  $p_{a_1} = s_{a_1} u_1$  and  $p_{a_2} = s_{a_2} u_2$ .

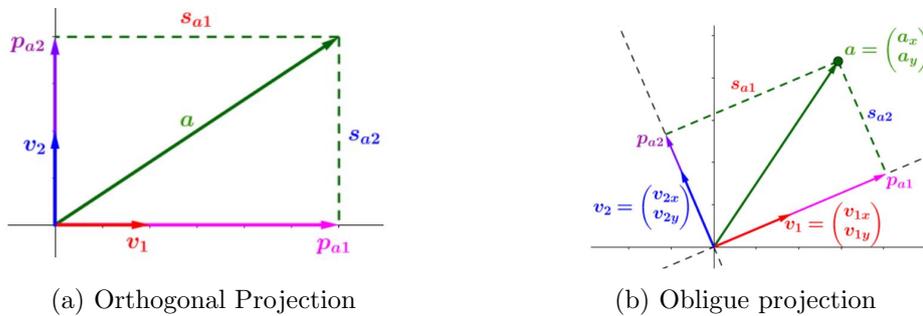


Figure D.1: Projection of vector  $a_i$

Based on the image above, every vector  $s_{a_i}$  is equal to:

$$a^T \cdot v_1 = \begin{pmatrix} a_x & a_y \end{pmatrix} \cdot \begin{pmatrix} v_{1x} \\ v_{1y} \end{pmatrix} = s_{a1}$$

$$a^T \cdot v_2 = \begin{pmatrix} a_x & a_y \end{pmatrix} \cdot \begin{pmatrix} v_{2x} \\ v_{2y} \end{pmatrix} = s_{a2}$$

The two different vectors can be merged into one matrix and can be expressed as:

$$a^T \cdot V = (a_x \ a_y) \cdot \begin{pmatrix} v_{1x} & v_{2x} \\ v_{1y} & v_{2y} \end{pmatrix} = (s_{a1} \ s_{a2})$$

This process can be expanded for more points, which may illustrate the pixel points in images. Therefore, the previous equation can be rewritten as:

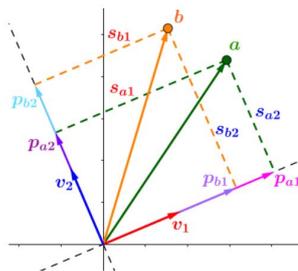


Figure D.2: Projection

$$A \cdot V = \begin{pmatrix} a_x & a_y & \dots \\ b_x & b_y & \dots \\ \vdots & \vdots & \ddots \end{pmatrix} \cdot \begin{pmatrix} v_{1x} & v_{2x} & \dots \\ v_{1y} & v_{2y} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix} = \begin{pmatrix} s_{a1} & s_{a2} & \dots \\ s_{b1} & s_{b2} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix} = S \quad (\text{D.1})$$

$$S = \begin{pmatrix} s_{a1} & s_{a2} & \dots \\ s_{b1} & s_{b2} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix} \quad (\text{D.2})$$

where the square root of the sum of the square in every column of the matrix  $S$ , is referred as the magnitude of the column vector and is expressed as follows:

$$\text{Magnitude of } 1^{st} \text{ column} = \sigma_1 = \sqrt{(s_{a1})^2 + (s_{b1})^2 + \dots + (s_{i1})^2} \quad (\text{D.3})$$

$$\text{Magnitude of } 2^{nd} \text{ column} = \sigma_2 = \sqrt{(s_{a2})^2 + (s_{b2})^2 + \dots + (s_{i2})^2} \quad (\text{D.4})$$

Therefore, for a set of  $n$  different vectors, the matrix  $S$  can be normalized as:

$$S = \begin{pmatrix} \frac{s_{a1}}{\sigma_1} & \frac{s_{a2}}{\sigma_2} & \dots \\ \frac{s_{b1}}{\sigma_1} & \frac{s_{b2}}{\sigma_2} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix} \begin{pmatrix} \sigma_1 & 0 & \dots \\ 0 & \sigma_2 & \dots \\ \vdots & \vdots & \ddots \end{pmatrix} = \begin{pmatrix} u_{a1} & u_{a2} & \dots \\ u_{b1} & u_{b2} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix} \begin{pmatrix} \sigma_1 & 0 & \dots \\ 0 & \sigma_2 & \dots \\ \vdots & \vdots & \ddots \end{pmatrix} \quad (\text{D.5})$$

Hence  $\Sigma$  is the normalized valued matrix :

$$\Sigma = \begin{pmatrix} \sigma_1 & 0 & \dots \\ 0 & \sigma_2 & \dots \\ \vdots & \vdots & \ddots \end{pmatrix} \quad (\text{D.6})$$

The SVD of a matrix  $A$  of size  $m \times n$  is summarized as:

$$A = U\Sigma V^T \quad (\text{D.7})$$

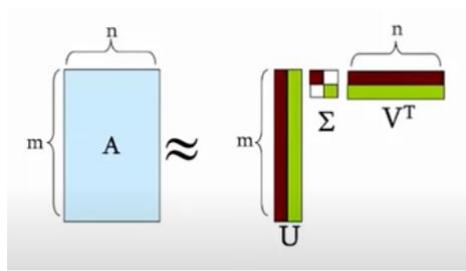
In a nutshell:

- $U$  is called the left singular vectors and is orthogonal of size  $m \times m$ .
- $\Sigma$  is called the singular values and is diagonal of size  $m \times n$ .
- $V$  is called the right singular vectors and is orthogonal of size  $n \times n$ .

Mathematically, the SVD is derived as follows. The singular values  $\sigma_i$  are defined by finding the eigenvalues of  $A^T A$ . The right singular vectors, meaning the columns of the matrix  $V$ , are computed by finding an orthonormal set of eigenvectors of  $A^T A$ . The eigendecomposition for the symmetric positive (semi) definite matrix  $A^T A$  gives rise to the  $\Sigma$  and  $V$  matrices. Finally, the  $U$  is defined by using the equation  $AV = U\Sigma$ .

The beneficial property of the SVD is that it can be applied over any matrix, not just diagonalizable ones and additionally, it uses orthonormal basis (unitary matrices), not just unitarily diagonalizable matrices.

The values of the diagonal matrix  $\Sigma$  are organized in descending order from the most to the least important value. Hence, this matrix indicates the energy concentration of the input-output matrix in descending mode and therefore the rank of the system can be determined based on the dominant patterns of this matrix, called modes. There is no clear selection criteria for the optimum amount of modes. The selection of the dominant modes, hence

Figure D.3: The SVD of a  $m \times n$  matrix

the system rank, varies in terms of the accuracy needed for the estimated model.

SVD is the fundamental tool for all reduction order methods mentioned above and is the key feature that indicates the rank of the reduced order model according to the dominant values of the  $\Sigma$  matrix and extracts dominant features of a high dimensional dynamical systems, used for analysing, understanding and describing the data.

# Appendix E

## Combined (Stochastic - Deterministic) SID

The subspace identification methods are developed to handle successfully multiple-input and multiple-output (MIMO) linear systems. The objective of these algorithms is to estimate linear and time invariant state space model directly from input- outputs without any prior information about the system.

Any state space system with given output measurements  $y_k$  and inputs  $u_k$  is expressed in the following form:

$$x(k+1) = Ax(k) + Bu(k) + Gw(t), \quad (\text{E.1})$$

$$y(k) = Cx(k) + Du(k) + v(t), \quad (\text{E.2})$$

The subspace identification algorithm estimates a state space model where the order  $n$  and  $A, B, C, D, K$  matrices are determined through the projection of input and output data. These algorithms are convergent (non-iterative) and numerically stable since they are based on the QR Decomposition and Singular Value Decomposition.

The subspace identification algorithm is initiated by forming the Hankel matrices of the input- output data. Hankel matrices are found to be useful for decomposition of non-stationary signals and time-frequency representation and definition of a state space realization.

$$U_{0|2i-1} = \begin{bmatrix} u_0 & u_1 & \dots & u_{j-1} \\ u_1 & u_2 & \dots & u_j \\ \vdots & \vdots & \vdots & \vdots \\ u_{i-1} & u_i & \dots & u_{i+j-2} \\ u_i & u_{i+1} & \dots & u_{i+j-1} \\ \vdots & \vdots & \vdots & \vdots \\ u_{2i-1} & u_{2i+1} & \dots & u_{2i+j-2} \end{bmatrix} \quad (\text{E.3})$$

which can be rephrased to:

$$U_{0|2i-1} = \frac{U_p}{U_f} \quad (\text{E.4})$$

where  $i$  is the number of block rows and  $j$  is the number of sampled data. The number of block rows  $i$  is user defined and has to be at least greater than the order of the system that will be defined [62]. Moreover, the number of columns is equal to  $s = 2i + j - 2$ , assuming that all given data are used in the identification process.

Analysing equation E.3, the past and future inputs are defined as:

$$U_p = \begin{bmatrix} u_0 & u_1 & \dots & u_{j-1} \\ u_1 & u_2 & \dots & u_j \\ \vdots & \vdots & \vdots & \vdots \\ u_{i-1} & u_i & \dots & u_{i+j-2} \end{bmatrix} \quad (\text{E.5})$$

$$U_f = \begin{bmatrix} u_i & u_{i+1} & \dots & u_{i+j-1} \\ u_{i+1} & u_{i+2} & \dots & u_{i+j} \\ \vdots & \vdots & \vdots & \vdots \\ u_{2i-1} & u_{2i+1} & \dots & u_{2i+j-2} \end{bmatrix} \quad (\text{E.6})$$

Accordingly, the Hankel matrices of past and future outputs as  $Y_p$  and  $Y_f$  can be defined. Moreover, the matrix  $W_p$  presents the past inputs and outputs is shown below:

$$W_p = \frac{U_p}{Y_p} \quad (\text{E.7})$$

The state sequence matrices for the deterministic and stochastic states that play an important role in deriving the state space matrices in the sub-space identification algorithms are shown below.

Deterministic state matrix:

$$x_i^d = [x_i^d \quad x_{i+1}^d \quad x_{i+2}^d \quad \dots \quad x_{i+j-1}^d] \quad (\text{E.8})$$

Stochastic state matrix:

$$x_i^s = [x_i^s \ x_{i+1}^s \ x_{i+2}^s \ \dots \ x_{i+j-1}^s] \quad (\text{E.9})$$

The subspace identification algorithm derivation is based on the observability and controllability matrices. It will be shown that through the extraction of these matrices, the state space model will be defined (see ERA algorithm).

The observability matrix is given by the equation:

$$\Gamma_i = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{i-1} \end{bmatrix} \in R^{li \times n} \quad (\text{E.10})$$

and the controllability matrix is given by the equation:

$$\Delta_i = [A^{i-1}B \ A^{i-2}B \ \dots \ AB \ B] \quad (\text{E.11})$$

and finally the lower triangular Toeplitz matrix is defined as follows:

$$H_i^d = \begin{bmatrix} D & 0 & 0 & \dots & 0 \\ CB & D & 0 & \dots & 0 \\ CAB & CB & D & \dots & 0 \\ \dots & \dots & \dots & \ddots & \dots \\ CA^{i-2}B & CA^{i-3}B & CA^{i-4}B & \dots & D \end{bmatrix} \quad (\text{E.12})$$

The subspace system identification is based on the projection of outputs over inputs. It will be proven that by defining this projection, the observability matrix is obtained, from which the A and C matrices of the state space model can be determined. There are two main algorithms on the SID. The N4SID that is based on the oblique projection and the MOESP that is based on orthogonal between the input-output matrices respectively.

The subspace identification algorithm proof begins from the innovation form of a linear discrete state space model:

$$Y_p = \Gamma_i X_p^d + H_i^d U_p + Y_p^s \quad (\text{E.13})$$

$$Y_f = \Gamma_i X_f^d + H_i^d U_f + Y_f^s \quad (\text{E.14})$$

$$X_f^d = A_i X_p^d + \Delta_i^d U_p \quad (\text{E.15})$$

$$(\text{E.16})$$

For the determination of the stochastic terms based on the Kalman filter, the following equation is given:

$$\begin{aligned} Z_i &= Y_f / \begin{pmatrix} \mathbf{W}_p \\ \mathbf{U}_f \end{pmatrix} \\ &= \Gamma_i \hat{X}_i + H_i^d U_f \end{aligned} \quad (\text{E.17})$$

where  $\hat{X}_i$  are the Kalman filter states as derived from the Kalman filter presented in Chapter 2, working in parallel on each of the columns of the block Hankel matrix of past inputs and outputs  $W_p$ .

Solving the state space equation E.13 of  $Y_p$  for  $X_p^d$  with the simplicity of the equations whereas  $Y_p^s$  and  $Y_f^s$  are zero, we get:

$$X_p^d = \Gamma_i^\dagger Y_p - \Gamma_i^\dagger H_i^d U_p \quad (\text{E.18})$$

Solving equation E.15 for  $X_f^d$  and substituting equation E.18, it gives :

$$\begin{aligned} X_f^d &= A_i X_p^d + \Delta_i^d U_p \\ X_f^d &= A_i [\Gamma_i^\dagger Y_p - \Gamma_i^\dagger H_i^d U_p] + \Delta_i^d U_p \\ X_f^d &= L_p W_p \end{aligned} \quad (\text{E.19})$$

where:

$$L_p = \Delta_i^d - A_i \Gamma_i^\dagger H_i^d | A_i \Gamma_i^\dagger \quad \& \quad W_p = \frac{U_p}{Y_p}$$

By taking equation E.14 and substituting in equation E.19:

$$\begin{aligned} Y_f &= \Gamma_i L_p W_p + H_i^d U_f \\ Y_f \Pi_{U_f^\perp} &= \Gamma_i L_p W_p \Pi_{U_f^\perp} + H_i^d U_f \Pi_{U_f^\perp} \\ Y_f \Pi_{U_f^\perp} &= \Gamma_i L_p W_p \Pi_{U_f^\perp} \\ Y_f \Pi_{U_f^\perp} &= \frac{\Gamma_i L_p W_p}{U_f^\perp} \\ [Y_f \Pi_{U_f^\perp}] \left[ \frac{W_p}{U_f^\perp} \right] W_p &= \Gamma_i L_p W_p \end{aligned} \quad (\text{E.20})$$

The above equation can be rephrased as follows:

$$O_i = \Gamma_i X_f^d \quad (\text{E.21})$$

or alternatively :

$$O_i = \begin{bmatrix} Y_f \\ U_f \end{bmatrix} \begin{bmatrix} W_p \\ U_f^\perp \end{bmatrix} W_p$$

$$O_i = Y_{f/U_f} W_p \quad (\text{E.22})$$

The importance of projection is clearly shown. The projection of the row space of a matrix onto the orthogonal complement to the row space of the matrix  $B$ ,  $\Pi_B^\perp$ , a filtering on the inputs  $U_f$  is accomplished. Hence, the subspace system identification is immune to input noise and the extended observability matrix  $\Gamma_i$  can be derived directly from input-output Hankel matrices.

The extended observability matrix  $\Gamma_i$  and the system order can be extracted from the SVD (Singular Value Decomposition) of  $O_i$ . Applying Singular Value Decomposition on  $O_i$ , the observability matrix  $\Gamma_i$  can be determined.

$$W_1 O_i W_2 = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} S_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} \quad (\text{E.23})$$

The user-defined weighted matrices  $W_1$  and  $W_2$  are such that  $W_1$  is of full rank and  $W_2$  obeys:  $W_p = \text{rank}(W_p W_2)$ , where  $W_p$  is the block Hankel matrix containing past inputs and outputs. The selection of the weighted matrices of the SVD in equation E.23 plays an important role in the robustness of the SID algorithm [38,62] and they define which part of the original row space of the  $O_i$  will be obtained for the state space matrices estimation. The following table indicates the weight matrices used in SVD of the projection matrix in each subspace system identification algorithm.

There are two major algorithm that determine different values for the matrices  $W_1$  and  $W_2$ , called the Numerical algorithm for subspace system identification (N4SID) and the Multivariate output error state space MOESP which will be analysed below.

	$W_1$	$W_2$
N4SID	$I_{li}$	$I_j$
MOESP	$I_{li}$	$\Pi_{U_f^\perp}$

MOESP is based on the orthogonal projection of  $U_f$  and  $Y_f$  Hankel matrices. On the other hand, N4SID is based on the oblique projection of these matrices. There is no clear selection criteria of the algorithm that can be used and is user defined depending on the nature of the data.

The order of the state space system is equal to the number of the non negative singular values of SVD. Hence, the extended observability matrix is expressed as:

$$\Gamma_i = W_1 U_1 S \bar{2} \quad (\text{E.24})$$

where  $U_1$  comes from the first  $n$  rows of the SVD, where  $n$  is the order of the system.  $S$  is the diagonal matrix with non negative values greater than zero.

#### A & C Matrices Determination

By determining the  $\Gamma_i$  matrix, the  $A$  and  $C$  matrices can be defined. The combination of the first  $n$  rows of  $\Gamma_i$  matrix in equation E.10 and the definition of the observability matrix in equation E.24 gives:

$$C = \Gamma_i(1 : n) \quad (\text{E.25})$$

The matrix  $A$  is determined from the shift structure of  $\Gamma_i$  based on the determination of the observability matrix in equation E.10. Hence given that:

$$\underline{\Gamma}_i A = \bar{\Gamma}_i \quad (\text{E.26})$$

where  $\underline{\Gamma}_i$  is the matrix  $\Gamma_i$  without the last  $n$  rows and:

$$\underline{\Gamma}_i = \begin{bmatrix} C & CA & CA^2 & \vdots & CA^{i-2} \end{bmatrix} \epsilon R^{li \times n} \quad (\text{E.27})$$

$$\bar{\Gamma}_i = \begin{bmatrix} CA & CA^2 & \vdots & CA^{i-1} \end{bmatrix} \epsilon R^{li \times n} \quad (\text{E.28})$$

The matrix  $A$  is defined such as

$$A = \underline{\Gamma}_i^\dagger \bar{\Gamma}_i \quad (\text{E.29})$$

### B & D Matrices Determination

Given the fundamental equations of a state space model, matrices B and D could be determined from the following equation:

$$\begin{bmatrix} \hat{X}_{i+1} \\ Y_{i|i} \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} \hat{X}_i \\ U_{i|i} \end{bmatrix} + \begin{bmatrix} \rho_w \\ \rho_u \end{bmatrix} \quad (\text{E.30})$$

Unfortunately, the state sequences  $\hat{X}_{i+1}$  and  $\hat{X}_i$  cannot be determined directly from the input-output data, thus they have to be determined following a different process.

Following equation E.10 by shifting the border of the input-output matrices between “past” and “future”, the matrix in equation E.17 can be transformed in the following form :

$$\mathcal{Z}_{i+1} = \Gamma_{i-1} \hat{X}_{i+1} + H_{i-1}^d U_f^- \quad (\text{E.31})$$

where in that case the shifted state  $\hat{X}_{i+1}$  is a Kalman filter state and the equation E.31 is summarized as:

$$\mathcal{Z}_{i+1} = Y_f^- / \begin{pmatrix} W_p^+ \\ U_f^- \end{pmatrix} \quad (\text{E.32})$$

Combining equations E.17 and E.32, the  $\hat{X}_i$  and  $\hat{X}_{i+1}$  can be determined as:

$$\hat{X}_i = \Gamma_i^\dagger \cdot [\mathcal{Z}_i - H_i^d \cdot U_f] \quad (\text{E.33})$$

$$\hat{X}_{i+1} = \Gamma_{i-1}^\dagger \cdot [\mathcal{Z}_{i+1} - H_{i-1}^d \cdot U_f^-] \quad (\text{E.34})$$

Solving these equation in terms of  $\hat{X}_i$  and  $\hat{X}_{i+1}$  respectively, B and D can be determined as follows:

$$\begin{pmatrix} \Gamma_{i-1}^\dagger \mathcal{Z}_{i+1} \\ Y_{i|i} \end{pmatrix} = \begin{pmatrix} A \\ C \end{pmatrix} \Gamma_i^\dagger \mathcal{Z}_i + K U_f + \begin{pmatrix} \rho_w \\ \rho_u \end{pmatrix} \quad (\text{E.35})$$

where K is equal to :

$$K = \left[ \begin{array}{c} \left( B | (\Gamma_{i-1}^\dagger H_{i-1}^d) - A \Gamma_i H_i^d \right) \\ (D|0) - C \Gamma_i^\dagger H_i^d \end{array} \right] \quad (\text{E.36})$$

Moreover, for illustrating purposes, terms of the equation E.35 can be defined as:

$$L = \begin{pmatrix} A \\ C \end{pmatrix} \Gamma_i^\dagger = \begin{pmatrix} L_{1|1} & L_{1|2} & \dots & L_{1|i} \\ L_{2|1} & L_{2|2} & \dots & L_{2|i} \end{pmatrix} \quad (\text{E.37})$$

$$M = \Gamma_{i-1}^\dagger = (M_1 \quad M_2 \quad \dots \quad M_{i-1}) \quad (\text{E.38})$$

$$K = \begin{pmatrix} K_{1|1} & K_{1|2} & \dots & K_{1|i} \\ K_{2|1} & K_{2|2} & \dots & K_{2|i} \end{pmatrix} \quad (\text{E.39})$$

Hence, the combination of equations E.32 - E.39, leads to the following equation:

$$\begin{pmatrix} \mathcal{K}_{1|1} \\ \mathcal{K}_{1|2} \\ \mathcal{K}_{1|3} \\ \vdots \\ \frac{\mathcal{K}_{1|i}}{\mathcal{K}_{2|1}} \\ \mathcal{K}_{2|2} \\ \mathcal{K}_{2|3} \\ \vdots \\ \mathcal{K}_{2|i} \end{pmatrix} = \mathcal{N} \begin{pmatrix} D \\ B \end{pmatrix} \quad (\text{E.40})$$

where

$$N = \frac{\begin{pmatrix} -L_{1|1} & M_1 - L_{1|2} & \dots & M_i - L_{1|i} \\ M_1 - L_{1|2} & M_1 - L_{1|3} & \dots & 0 \\ M_2 - L_{1|3} & M_3 - L_{1|4} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ M_i - L_{1|i} & \dots & \dots & \dots \end{pmatrix}}{\begin{pmatrix} -I_{1|1} - L_{2|1} & L_{2|2} & \dots & L_{2|i-1} & L_{2|i} \\ L_{2|2} & L_{2|3} & \dots & L_{2|i} & 0 \\ L_{2|3} & L_{2|4} & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots \\ L_{2|i} & 0 & \dots & 0 & 0 \end{pmatrix}} \times \begin{pmatrix} I_i & 0 \\ 0 & \Gamma_i - 1 \end{pmatrix} \quad (\text{E.41})$$

The last formula is an overdetermined set of linear equations in the unknowns B and D, which could for instance be solved using least squares.

This approach can lead to poor performance when the input Hankel matrix is badly conditioned due to the introduction of a large correlation in the sample error when multiplying with  $U_f^\dagger$  in order to compute the matrix K.

Hence, an alternative approach for more accurate estimation for general purpose is given by:

$$B, D = \operatorname{argmin} \left\| \begin{pmatrix} \Gamma_{i-1}^\dagger Z_{i+1} \\ Y_{i|i} \end{pmatrix} - \begin{pmatrix} A \\ C \end{pmatrix} \Gamma_i^\dagger Z_i + K(B, D) U_f \right\|_F^2 \quad (\text{E.42})$$

Rewriting equation E.42, we obtain:

$$P = \begin{pmatrix} \Gamma_{i-1}^\dagger Z_{i+1} \\ Y_{i|i} \end{pmatrix} - \begin{pmatrix} A \\ C \end{pmatrix} \Gamma_i^\dagger Z_i$$

$$Q = U_f$$

which are known values.

Hence:

$$B, D = \operatorname{argmin} \left\| \operatorname{vec} P - \sum Q_k^T \otimes N_k \operatorname{vec} \begin{pmatrix} D \\ B \end{pmatrix} \right\|_F^2 \quad (\text{E.43})$$

By solving equation E.43, B and D are estimated via least square method.

$$\operatorname{vec} \begin{pmatrix} D \\ B \end{pmatrix} = \left[ \sum Q_k^T \otimes N_k \right]^\dagger \operatorname{vec} P \quad (\text{E.44})$$



# Appendix F

## ERA

The eigenvalue realization algorithm (ERA) is a fundamental control method for both system identification and model reduction. The ERA is a data driven method that accepts input-output data and estimates a minimal realization of the underlying system dynamics [70].

The ERA originates from the combination of the observability and controllability matrices, creating time lagged data structures, hence it creates Hankel form matrices of the input-output data in order to take advantage of the Hankel matrix properties.

The Markov parameters of an linear time invariant (LTI) system is a Hankel matrix of the multiplication of the observability  $\mathcal{P}_\alpha$  and controllability  $\mathcal{Q}_\beta$  matrix.

$$H_0 = \mathcal{P}_\alpha \mathcal{Q}_\beta \quad (\text{F.1})$$

Substituting, the matrices for  $\mathcal{P}_\alpha$  and  $\mathcal{Q}_\beta$ , the Hankel matrix can be expressed as follows:

$$H_0 = \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{(\alpha-1)} \end{pmatrix} (B \quad AB \quad A^2B \quad \dots \quad A^{(\beta-2)}B \quad A^{(\beta-1)}B)$$

By multiplying the observability and controllability matrices, it follows:

$$H_0 = \begin{pmatrix} CB & CAB & CA^2B & \cdots & CA^{(\beta-2)}B & CA^{(\beta-1)}B \\ CAB & CA^2B & CA^3B & \cdots & CA^{(\beta-1)}B & CA^{(\beta)}B \\ CA^2B & CA^3B & CA^4B & \cdots & CA^{(\beta)}B & CA^{(\beta+1)}B \\ \vdots & \vdots & \vdots & & \vdots & \\ CA^{(\alpha-1)}B & CA^{(\alpha)}B & CA^{(\alpha+1)}B & \cdots & CA^{(\alpha-1+\beta-2)}B & CA^{(\alpha-1+\beta-1)}B \end{pmatrix}$$

The matrix above proves that this Hankel matrix describes the Markov parameters of the input-output data:

$$H_0 = \begin{pmatrix} Y_1 & Y_2 & Y_3 & \cdots & Y_{\beta-1} & Y_{\beta} \\ Y_2 & Y_3 & Y_4 & \cdots & Y_{\beta} & Y_{\beta+1} \\ Y_3 & Y_4 & Y_5 & \cdots & Y_{\beta+1} & Y_{\beta+2} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ Y_{\alpha} & Y_{\alpha+1} & Y_{\alpha+2} & \cdots & Y_{\alpha+\beta-2} & Y_{\alpha+\beta-1} \end{pmatrix}$$

The structure of the latest equation is identical to the Hankel matrix.

The one-step forward shifted form of the previous Hankel form, gives rise to the following equation:

$$H_1 = \mathcal{P}_{\alpha} A \mathcal{Q}_{\beta} \quad (\text{F.2})$$

$$H_1 = \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{(n-1)} \end{pmatrix} A (B \quad AB \quad A^2B \quad \cdots \quad A^{(n-2)}B \quad A^{(n-1)}B)$$

By multiplying the observability and controllability matrices, it derives:

$$H_1 = \begin{pmatrix} CAB & CA^2B & CA^3B & \cdots & CA^{(\beta-1)}B & CA^{(\beta)}B \\ CA^2B & CA^3B & CA^4B & \cdots & CA^{(\beta)}B & CA^{(\beta+1)}B \\ CA^3B & CA^4B & CA^5B & \cdots & CA^{(\beta+1)}B & CA^{(\beta+2)}B \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ CA^{(\alpha)}B & CA^{(\alpha+1)}B & CA^{(\alpha+2)}B & \cdots & CA^{(\alpha+\beta-2)}B & CA^{(\alpha+\beta-1)}B \end{pmatrix}$$

Again, the matrix above describes the Markov parameters of the one step shifted input-output data:

$$H_1 = \begin{pmatrix} Y_2 & Y_3 & Y_4 & \cdots & Y_{n-1} & Y_n \\ Y_3 & Y_4 & Y_5 & \cdots & Y_{n+1} & Y_{n+2} \\ Y_4 & Y_5 & Y_6 & \cdots & Y_{n+2} & Y_{n+3} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ Y_{\alpha+1} & Y_{\alpha+2} & Y_{\alpha+3} & \cdots & Y_{\alpha+\beta-1} & Y_{\alpha+\beta} \end{pmatrix}$$

The controllability and observability matrices are thus computed by the SVD over the equation F.1.

$$H_0 \approx U_n \Sigma_n V_n^T = (U_n \Sigma_n^{(1/2)}) T T^{-1} (\Sigma_n^{(1/2)} V_n^T) = \mathcal{P}_\alpha \mathcal{Q}_\beta \quad (\text{F.3})$$

The matrix  $T$  is an arbitrary unitary transformation matrix.  
The system dynamic matrix  $A$  is derived by the equation F.2.

$$\begin{aligned} H_1 &= \mathcal{P}_\alpha A \mathcal{Q}_\beta \\ H_1 &= U_n \Sigma^{1/2} A \Sigma^{1/2} V_n^T \\ U_n^T H_1 V_n &= \Sigma_n^{1/2} A \Sigma_n^{1/2} \\ \Sigma^{-1/2} U_n^T H_1 V_n \Sigma^{-1/2} &\equiv A_n \end{aligned} \quad (\text{F.4})$$

On the other hand, the estimate of the input matrix  $B_n$  is computed as the first  $n$  columns of  $\mathcal{Q}_\beta$  and the estimate of the output matrix  $C_m$  is recovered as the first  $m$  rows of  $\mathcal{P}_\alpha$ . and the estimate of the feed-through matrix  $D_n$  is recovered from the first  $n$  columns of the sequence of Markov parameters.

The ERA introduces the Hankel matrix form through the combination of the input-output data via the system Markov parameters and estimates a reduced order model using the SVD for capturing the dominant structures. In the following chapters more reduced order model system identification methods will be deployed which are linked to the ERA.



# Appendix G

## 1-D Wavelet Transform

The Fourier analysis is represented by the Fourier transform which originates from the sum of the given signal multiplied by a complex exponential either in continuous or discrete time. The Fourier transform is expressed for both occasions below:

$$\begin{aligned} F(\omega) &= \int_{-\infty}^{\infty} f(x)e^{-j\omega x} dx, \text{ in continuous time} \\ F(\omega) &= \sum_{n=0}^{N-1} f_n \cdot e^{-\frac{i2\pi}{N}kn}, \text{ in discrete time} \\ &= \sum_{n=0}^{N-1} f_n \cdot \left[ \cos\left(\frac{2\pi}{N}kn\right) - i \cdot \sin\left(\frac{2\pi}{N}kn\right) \right] \end{aligned} \quad (\text{G.1})$$

On the other hand, the wavelet transform is the sum of the signal multiplied by scaled, shifted versions of the selected wavelet function, called mother wavelet. A simple continuous time 1-D wavelet transform is described as:

$$\begin{aligned} W(s, \tau) &= \int_{-\infty}^{\infty} f(x)\phi_{s,\tau}^*(\mathbf{x})dx, \text{ in continuous time} \\ W(j, k) &= \sum_k f(x)\phi_{j,k}(x), \text{ in discrete time} \end{aligned} \quad (\text{G.2})$$

For the continuous time wavelet transformation,  $f(x)$  is the input signal and  $\phi_{s,\tau}^*(\mathbf{t})$  is the wavelet analysing function. The output signal  $W(s,\tau)$  is a coefficient matrix that is identified by two terms: dilation-scale ( $s$ ) and translation-shift ( $\tau$ ).

$$\phi_{s,\tau} = \frac{1}{\sqrt{s}} \cdot \phi\left(\frac{x - \tau}{s}\right) \quad (\text{G.3})$$

The term translation indicates the shifting of the corresponding wavelet in time as the signal is analysed.

The term dilation indicates the amount of compression of the wavelet signal in every scale. A low frequency scale (high scale) wavelet is a wavelet stretched out and a high frequency scale (low scale) wavelet is a wavelet compressed. This means that in low frequencies it becomes less accurate to identify the time where a low frequency occurs. High frequencies show better localization in time due to the fact that the wavelet is much shorter. In few words, short scale wavelets can be used to isolate very fine details in a signal, while large scale wavelets can identify coarse details [27].

In every decomposition level, the continuous wavelet decomposition is represented by the convolution of the wavelet function and the given signal. Based on equation G.3 the wavelet transform signal is obtained by :

$$W(s, \tau) = f(x) * \phi_{s,\tau}(x) = \frac{1}{\sqrt{s}} \int_{-\infty}^{\infty} f(x) \phi\left(\frac{x - \tau}{s}\right) du \quad (\text{G.4})$$

where  $s = 2^{-j}$  and  $\tau = k2^{-j}$  and  $j$  represents the scale step and  $k$  is the shift step.

In every scale, the wavelet decomposes the signal into high and low frequency components, called approximation and detail coefficients. The signal is passing through two filters (one high- and one low-pass) and derives two separate signals. In every next level of decomposition, the approximation coefficients pass through the same filters and are decomposed into two new sets of detail and approximation coefficients. The wavelet decomposition process can be repeated, so that the input signal can be broken down into many lower resolution components.

### G.0.0.1 Wavelet tree structure and multiresolution analysis

The wavelet decomposition is analysing signals both in time and frequency where the given dataset is expanded in terms of the basis function which can be scaled in multiple scales of resolution. This multilevel representation is linked and relied on concepts of the multiresolution signal analysis, a tool that was invented by Mallat [32].

A Multiresolutional Analysis (MRA) generated by the function  $\phi(j, k)$ , consists of a sequence of closed subspaces  $V_j$ ,  $j \in \mathbb{Z}$ , of  $L_2(\mathbb{R})$  satisfying :

- (i) (nested)  $V_j \subset V_{j+1} \subset \dots \subset L_2(\mathbb{R})$  for all  $j \in \mathbb{Z}$
- (ii) (density)  $\overline{\cup_{j \in \mathbb{Z}} V_j} = L_2(\mathbb{R})$
- (iii) (separation)  $\cap_{j \in \mathbb{Z}} V_j = \{0\}$
- (iv) (scaling)  $f(x) \in V_j$  if and only if  $f(2x) \in V_{j+1}$  for all  $j \in \mathbb{Z}$
- (v) (Basis) There exists a function  $\phi \in V_0$  such that  $\{\phi(x - k) : k \in \mathbb{Z}\}$  is an orthonormal basis or a Riesz basis for  $V_0$ .

Analysing the multiresolution properties starting from the property number (ii), indicates that for every  $f \in V_j \iff g \in V_{j+1}$ , where  $g(x) = f(2x)$ . Moreover, the function  $\phi(x)$  in property (v) is also called the scaling function associated with the multiresolution analysis. In fact, the multiresolution analysis is also defined with the set  $\{\phi_{0,k}(x) = \phi(x - k), k \in \mathbb{Z}\}$  being a mere Riesz basis of  $V_0$ .

Since  $\phi(x) \in V_0 \subset V_1$ , there exists a sequence,  $\{h_k, k \in \mathbb{Z}\}$ , where  $h_k$  are the signal coefficients, such that:

$$\phi(x) = \sum_k h_k \phi_{1,k}(x) = \sqrt{2} \sum_k h_k \phi(2x - k) \quad (\text{G.5})$$

This equation G.5 is known by several different names, such as the dilation equation, the two-scale difference equation, or the refinement equation. As an expansion to the multiscale decomposition of a  $j$  number of scales, it follows directly that the collection of functions  $\{\phi_{j,k}, k \in \mathbb{Z}\}$  are given by:

$$\phi_{j,k}(x) = 2^{j/2} \phi(2^j x - k), \quad k \in \mathbb{Z} \quad (\text{G.6})$$

and constitutes an orthonormal basis of  $V_j$ . The scaling function  $\phi$  is uniquely defined by its dilation equation G.5 and the normalization that lays :

$$\sum_k \phi(x) dx = 1$$

The equation G.5 contains the low frequency coefficients, called approximation coefficients. A wavelet is a function  $\psi$  such that the collection of functions  $\{\psi(x - k), k \in \mathbb{Z}\}$  constitutes an orthonormal basis of  $W_0$ , where  $V_1 = V_0 \oplus W_0$ . Therefore, the collection of wavelet functions  $\{\psi_{j,k}, j, k \in \mathbb{Z}\}$  is an orthonormal basis of  $L^2(\mathbb{R})$  and contains the high frequency coefficients, called detail coefficients.

The definition of the wavelet function  $\psi_{j,k}$  per scale  $j$  is similar to the definition of  $\phi_{j,k}$  :

$$\psi_{j,k}(x) = 2^{j/2} \psi(2^j x - k) \quad (\text{G.7})$$

The wavelet function  $\psi$  satisfies an equation similar to that of the scaling function  $\phi$ :

$$\psi(x) = \sqrt{2} \sum_k g_k \phi(2x - k) \quad (\text{G.8})$$

where the coefficients  $g_k$  are given by

$$g_k = (-1)^k h_{-k+1}$$

As the scales of decomposition are increased, the scaling and wavelet functions are subsampled by a factor of 2, thus the wavelet function is every next scale is given by:

$$\begin{aligned} \psi(2x - k) &= \sum_{m \in \mathbf{Z}} g_m \sqrt{2} \phi(2(2x - k) - m) \\ &= \sum_{m \in \mathbf{Z}} g_m \sqrt{2} \phi(2^{j+1}x - 2k - m) \end{aligned} \quad (\text{G.9})$$

where  $m$  replaces  $k$  for the next level of decomposition and this transform is continued till a desired level of decomposition is reached.

As opposed, every scaling-approximation function in spaces  $V_{j+1}$ , allows the wavelet  $W_j$  to be its orthogonal complement subspace . An orthogonal complement has the quality that every vector in  $W_j$  is orthogonal to every basis in  $V_j$  . In addition, the space  $V_{j+1}$  is the direct sum of  $W_j$  and  $V_j$  .

This can be summarized to the following two useful properties :

- $\mathcal{V}_j \perp \mathcal{V}_j, \mathcal{V}_j \perp \mathcal{W}_j, \mathcal{W}_1 \perp \mathcal{W}_2 \perp \mathcal{W}_{j-1}$
- $V_{j+1} = V_j \oplus W_j$

The spaces  $W_j$  contain adequate detail information to jump from an approximation at the resolution scale  $j$  to an approximation at the resolution scale  $j + 1$ . Consequently:

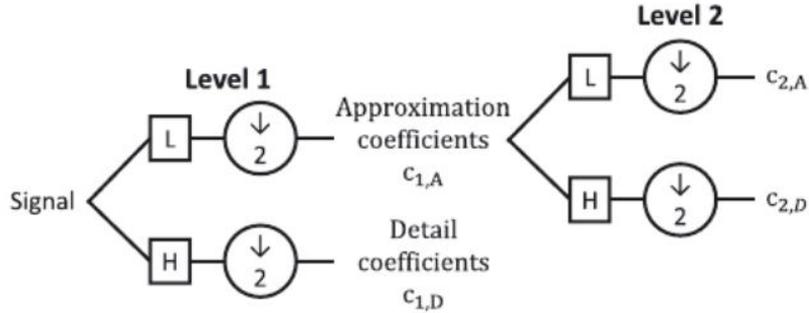


Figure G.1: A two-level signal breakdown into approximation and detail coefficients

$$\bigoplus_j W_j = L^2(R)$$

The wavelet subspaces are perpendicular to each other, meaning that the wavelets are orthogonal too. The orthogonal basis functions allow simple calculation of expansion coefficients and allow a partitioning of the signal energy in the wavelet transform domain. The following properties imply for the wavelet and scaling functions, shaping the wavelet decomposition as a very efficient mathematical tool. In the following section, the wavelet properties will be analysed and proven and will be of main use for the purpose of this thesis.

In a nutshell, the Discrete time wavelet decomposition (DWT) deploys the wavelet transform that uses a discrete set of the wavelet scales and translations. The transform decomposes the signal into mutually orthogonal set of wavelets as mentioned above and are represented by discrete dyadic grids using the scaling parameter  $j$  and the shift parameter  $k$  as introduced for the continuous wavelet transform.

In each level of decomposition, two signals are derived that contain the approximation and the detail coefficients.

$$\begin{aligned} W_\phi(j, k) &= \frac{1}{\sqrt{s}} \sum f(x) \phi_{j,k}(x) \\ W_\psi(j, k) &= \frac{1}{\sqrt{s}} \sum f(x) \psi_{j,k}(x) \end{aligned} \quad (\text{G.10})$$

where  $b = 2^{-j} \cdot n$  is the position of the wavelet (see translation),  $s$  is the scaling factor and  $s = 2^{-j}$ .  $W_\phi(j, k)$  contains the image approximation coefficients in the level  $j$  of decomposition and  $W_\psi(j, k)$  contains the detail (high

frequency ) coefficients in the level  $j$  of decomposition and after employing the refinement equation of  $\phi$  and  $\psi$ , the scaling function  $\phi_{j,k}(x)$  and wavelet function  $\psi_{j,k}(x)$  are defined as :

$$\begin{aligned}\phi_{j,k}(x) &= 2^{-\frac{j}{2}}\phi(2^jx - k) \\ \psi_{j,k}(x) &= 2^{-\frac{j}{2}}\psi(2^jx - k)\end{aligned}\quad (\text{G.11})$$

The scaling function is orthogonal to the wavelet function by construction and hence the discrete wavelets are called orthogonal wavelets. Thus, they decompose signals into well-behaved orthogonal signal spaces. [27]

There are several equivalent viewpoints from which the DWT can be regarded. The wavelet decomposition implementation can be formulated by cascade filters where the low pass and high pass filters are estimating the approximation and detail signal in every scale respectively. The wavelets employ filter banks consisting of a low-pass filter  $h_\phi$  and a high-pass filter  $h_\psi$ . In every level  $j$  of decomposition, the approximation signal  $\phi_{j,k}$  and the detail signal  $\psi_{j,k}$  are derived from a given discrete-time signal  $f[k]$ . The same process is repeated to the approximation signal  $W\phi(j, k)$  in every level of decomposition where a new set of  $W\phi(j + 1, k)$  and  $W\psi(j + 1, k)$  that are subsampled by a factor of 2 are derived till a desired level of decomposition is achieved.

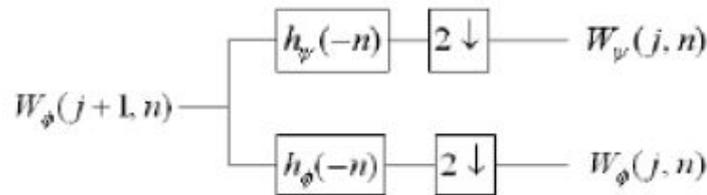


Figure G.2: One Scale Image Wavelet Decomposition

The wavelet function in every level of decomposition is described as:

$$W_\psi(j, k) = \frac{1}{\sqrt{M}} \sum_x f(x) 2^{j/2} \left[ \sum_p g_p \sqrt{2} \phi(2^{2^j+1} - 2k - p) \right] \quad (\text{G.12})$$

which is equal to:

$$W_\psi(j, k) = \sum_x g_p W_\phi(j + 1, k). \quad (\text{G.13})$$

The scaling process creates a set of coefficients in each scale that represent the given signal among the different scales providing full insight of the signal properties and hidden details in respect to the number of selected scales. Moreover, the convolution of the input signal with the corresponding wavelet offers a signal noise filtering in every scale.

In every next level of decomposition, the approximation coefficients are downsampled by a factor of 2 and a new set of approximation and detailed coefficients are obtained. The process is continued till a desired level of decomposition is obtained.

In summary, the wavelet decomposition has the property of being local in space and frequency. They introduce a compact support (localization in space), which is smooth (decay towards high frequencies), and which have vanishing moments (decay towards low frequencies). The order of a wavelet transform is typically given by the number of vanishing moments of the analysis wavelet. More vanishing moments means that the scaling function can represent more complex functions and the wavelet is unbiased. The higher the number of zero moments, the higher the number of zero derivatives making smoother the signal decays from mid frequency to DC in the frequency domain.



# Bibliography



# Bibliography

- [1] Sarkka, S., Chou, K.C., Willsky, A.S. & Nikoukhah, R. (1994). Multiscale Systems, Kalman Filters, and Riccati Equations. *IEEE Transactions On Automatic Control*, Vol.39, No.3.
- [2] Willett, R.M. & Nowak, R.D. (2007). Multiscale Poisson Intensity and Density Estimation. *IEEE Transactions on Information Theory*, Vol.53, No.9, 3171.
- [3] Willsky, A.S. (2002). Multiresolution Markov Models for Signal and Image Processing. *Proceedings of The IEEE*, Vol.90, No.8.
- [4] Weinan, E.(2011). *Principles of Multiscale Modeling*. Cambridge University Press.
- [5] Choi, M.J., Chandrasekaran, V. & Willsky, A.S. (2010). Gaussian Multiresolution Models: Exploiting Sparse Markov and Covariance Structure. *IEEE Transactions on Signal Processing*, Vol.58, No.3.
- [6] Brandt, A. (1977). Multi-Level Adaptive Solutions to Boundary Value Problems. *Math. of Computation*, Vol.31,333-390.
- [7] Greengard, L.,RokhlinA, V. (1987) A fast algorithm for particle simulations. *Computer Science, Journal of Computational Physics*, Vol.73, Issue 2, 325-348.
- [8] Weinan, E., Engquist, B., Xiantao Li, Ren, W. & Vanden-Eijnden, E. (2013). *Heterogeneous Multiscale Methods: A Review*. Cambridge University Press.
- [9] Verfürth, R. (1998). A Review of A Posteriori Error Estimation Techniques for Elasticity Problems. *Studies in Applied Mechanics*, Vol. 47, 257-274.

- 
- [10] Rieger, A., Wriggers, P. (2001). Error Control in Frictional Contact Problems. *Computational Mechanics—New Frontiers for the New Millennium*, 697-706.
- [11] Toselli, A., Widlund, O. (2004). *Domain Decomposition Methods - Algorithms and Theory*. Springer Series in Computational Mathematics, Vol.34, 2004.
- [12] Smith, B., Bjorstad, P., Gropp, W. (1996). *Domain Decomposition, Parallel Multilevel Methods for Elliptic Partial Differential Equations*. Cambridge University Press 1996.
- [13] Mallat, S. (1999). *A Wavelet Tour of Signal Processing*, 2nd Edition.
- [14] Chopard, B., Borgdorff, J., Hoekstra, AG. (2014). A framework for multi-scale modelling. *Phil. Trans. R. Soc. A* 372.
- [15] Stephanopoulos, G., Dyer, M. & Karsligil, O. (1997). Multi-scale modeling, estimation and control of processing systems. *Computers & Chemical Engineering*, 21, S797–S803.
- [16] Willsky, A.S.. Multiresolution Markov models for signal and image processing (2002). *Proceedings of the IEEE*, 90, 1396–1458.
- [17] Timmerman, K.E. & Nowak, R.D. (1999). Multiscale modeling and estimation of Poisson processes with application to photon-limited imaging. *IEEE Transactions on Information Theory*, 45, 846–862.
- [18] Bouman, C.A. & Shapiro, M. (1994). A multiscale random field model for bayesian image segmentation. *IEEE Transactions on Image Processing*, 3, 162–177.
- [19] Kannan, A. , Ostendorf, M., Karl, W.C., Castanon, D.A. & Fish R.K. (2000). ML parameter estimation of a multiscale stochastic process using the EM algorithm. *IEEE Transactions on Signal Processing*, 48, 1836 – 1840.
- [20] Choi, M.J., Chandrasekaran, V., Malioutov, D.M., Johnson, J.K. & Willsky,A.S. (2008). Multiscale stochastic modeling for tractable inference and data assimilation. *Comput. Methods Appl. Mech. Engrg.* 197, 3492–3515.
- [21] Chou, K.C., Willsky, A.S., & Benveniste,A. (1994). Multiscale recursive estimation, data fusion and regularization. *IEEE Transactions on Automatic Control*, 39, 464–478.

- 
- [22] Chou, K.C., Willsky, A.S., & Nikoukhah, R. (1994). Multiscale systems, Kalman filters, and Riccati equations. *IEEE Transactions on Automatic Control*, 39, 479–492.
- [23] Rauch, H.E., Tung F., & Striebel C.T. (1965). Maximum likelihood estimates of linear dynamic systems. *AIAA Journal*, 3, 1445–1450.
- [24] Tustison, B., Foufoula-Georgiou, E. & Harris. D. (2003). Scale-recursive estimation for multisensor quantitative precipitation forecast verification: a preliminary assessment. *Journal of Geophysical Research*, 108, 8377.
- [25] Simone, G., Morabito, F.C. & Farina, A., (2000). Radar image fusion by multiscale Kalman filtering. In *Proceedings of the Third International Conference on Information Fusion Volume 2*.
- [26] Unser, M. & Blu. T. (2003). Wavelet theory demystified. *IEEE Transactions on Signal Processing*, 51, 470–483.
- [27] Prasad, P.M.K. , Prasad, D.Y.V., Sasibhushana Rao, G. (2016). Performance Analysis of Orthogonal and Biorthogonal Wavelets for Edge Detection of X-ray Images. 4th International Conference on Recent Trends in Computer Science & Engineering, *Procedia Computer Science* 87, 116 – 121.
- [28] Ibrahim, M., Jemei, S., Wimmer, G., Steiner, N.Y, Kokonendji, C.C, Hissel, D. (2015). *International journal of hydrogen energy*, 40, 15823-15833.
- [29] Kumar, D., Kaur, J. & Sharma, V. September (2016). Analysis of Various Wavelets Transform Techniques. *International Journal of Wireless Communications and Networking Technologies*, Vol.5, 46-51.
- [30] Antonini, M., Barlaud, M., Mathieu, P. & Daubechies, I. (1992). Image Coding Using Wavelets. *IEEE Transactions on Image Processing*, Vol. I, No 2.
- [31] Hajji, M.A., Melkonian, S., Vaillancourt, R. (2003). Representation of Differential Operators in Wavelet Basis. *Computers and Mathematics with Applications* 47, 1011-1033
- [32] Schneider, K. & Vasilyev, O.V. (2019). Wavelet Methods in Computational Fluid Dynamics. *The Annual Review of Fluid Mechanics*, 42:473–503.

- 
- [33] Ho, B., L., Kalman, R., E. (1965). Effective construction of linear state-variable models from input-output functions. 3rd Ann. Allerton conf. Circuit and System Theory. Vol.14, 545-548.
- [34] Zeiger H., McEwen A. (1974). Approximate linear realizations of given dimension via Ho's algorithm. IEEE Transactions on Automatic Control, Vol.19, 153.
- [35] Ljung L. (1991). Issues in System Identification. IEEE Control Systems, Vol. 11, Vol.1, 25-29.
- [36] Ljung. L. (1991). A Simple Start-Up Procedure for Canonical Form State Space Identification, Based on Subspace Approximation. 30th IEEE Conference on Decision and Control, 1333-1336.
- [37] Vajpayee, V., Mukhopadhyay, S., Pati Tiwari, A. (2018). Multiscale subspace identification of nuclear reactor using wavelet basis function. Annals of Nuclear Energy Vol.111, 280-292.
- [38] Swevers, J. (2006). Introduction to system identification. K.U.Leuven.
- [39] Piltan, F., Tayebi Haghghi, S., Sulaiman, N.B. (2017). Comparative Study between ARX and ARMAX System Identification. International Journal of Intelligent Systems and Applications 2, 25-34.
- [40] Rachad, S., Nsiri, B. & Bensassi, B. (2002) .System Identification of Inventory System Using ARX and ARMAX Models. International Journal of Control and Automation Vol.8, No.12, 283-294.
- [41] Joe Qin,S. (2006). An overview of subspace identification. Computers and Chemical Engineering, 30, 1502–1513.
- [42] Van Overshee, P., De Moor, B. (1996). Subspace System Identification For Linear Systems ,Theory - Implementation - Applications. K.U. Leuven Belgium
- [43] Hachicha, S., Kharrat, M. & Chaari, A. (2014). N4SID and MOESP Algorithms to Highlight the ill-conditioning into Subspace Identification. International Journal of Automation and Computing, 11, 30-38.
- [44] Ruscio, D.D. (1995). Subspace system identification: Theory and applications. 6<sup>th</sup> edition, Telemark University College.
- [45] Movellan, J.R. (2011). Discrete Time Kalman Filters and Smoothers. MPLab Tutorials, University of California San Diego.

- 
- [46] Aravkin, A., Burke, J.V., Lennart Ljung, Lozano A. & Pillonetto G. (2017). Generalized Kalman smoothing: Modelling and algorithms. *Automatica*, 86, 63–86.
- [47] Sarkka, S. (2013). *Bayesian Filtering and Smoothing*. Cambridge University Press, Vol.3.
- [48] Sarkka, S., Vincent Poor, H. (1997). Delta-Operator Based Signal Processing: Fast Algorithms for Rapidly Sampled Data. *Proceedings of the IEEE Conference on Decision and Control*, 872-877.
- [49] Huang, D. & Katayama, T. (2004). A closed-loop subspace identification method for continuous-time systems based on  $\delta$ -operator model. *Asian Journal of Control*, Vol.6, 341-352.
- [50] Miao, Y., Guo, G., & Liu, J. (2021) Closed-loop delta-operator-based subspace identification for continuous-time systems utilising the parity space. *International Journal of Systems Science*, , Vol.52, 3323–3334.
- [51] Verhaegen, M., Westwick, D. & Kearney, R. (1995). The Use of a Bilinear Transformation of the Shift Operator in Subspace Model Identification. *IEEE Transaction on Automatic Control*, Vol.40, No.8.
- [52] Verhaegen, M., & Dewilde, P. (1992). Subspace model identification-Analysis of the elementary output error state-space model identification algorithm. *International Journal of Control*, Vol. 56, No. 5, pp. 1211-1241.
- [53] Verhaegen, M., & Dewilde, P. (1994). Identification of the deterministic part of MIMO state space models given in innovation form from input-output data, *Automatica*, Vol. 30, No. 1, pp. 61-74.
- [54] Kulikov, Yu.G., Kulikova, Maria V. (2016). Square-root Kalman-like filters for estimation of stiff continuous-time stochastic systems with ill-conditioned measurements. *IET Control Theory & Applications*, ISSN, 1751-8644.
- [55] LEE, Y. & Engquist, B. (2013). *Variable Step Size Multiscale Methods for Stiff and Highly Oscillatory Dynamical Systems*. Cornell University.
- [56] Nounou, M.N. (2006). Enhanced State Estimation using Multiscale Kalman Filtering. *Proceedings of the 45th IEEE Conference on Decision & Control WeIP5.7*, 1679-1684.
- [57] Reis, M.S. (2009). A multiscale empirical modeling framework for system identification. *Journal of Process Control*, Vol.19 (9), 1546-1557.

- [58] Kadiramanathan, V. & Anderson, S.R. (2008). An iterative Kalman smoother/least-squares algorithm for the identification of delta-ARX models. *IEEE Transactions on Signal Processing*, Vol.56, No.8, 376.
- [59] Mercieca, J., Kadiramanathan, V. (2016). Estimation and Identification of Spatio-Temporal Models with Applications in Engineering, Healthcare and Social Science. *Annual Reviews in Control*, 42, 285-298.
- [60] Mangion, A.Z. (2012). Modelling from spatiotemporal data: a dynamic systems approach. Thesis (Ph.D.) - University of Sheffield, Department of Automatic Control and Systems Engineering.
- [61] Harlim, J. (2008). Numerical strategies for filtering partially observed stiff stochastic differential equations. *Journal of Computational Physics* 227(10): 5304-5341.
- [62] Vajpayee, V., Mukhopadhyay, S., Pati Tiwari, A. (2018). Multiscale subspace identification of nuclear reactor using wavelet basis function. *Annals of Nuclear Energy* Vol.111, 280-292.
- [63] Saifizi, M., Ab Muin, M.Z., Yaacob, S. & M. S. Mohamad, M.S. (2013). Comparison of ARX and ARMAX model for Thermoelectric Refrigerator. *International Conference on Mechanical Engineering Research*, 338.
- [64] Katayama, t. (2005). *Subspace Methods for System Identification*. Communications and Control Engineering Series, ISSN, 0178-5354.
- [65] Joe Qin, S. (2008). An overview of subspace identification. *AIChE Journal* 54 (7), 1811-1829.
- [66] Goodwin, G.C., Middleton, R.H. & Poor, H.V., *High-speed Digital Signal Processing and Control* (1992). *Proceedings of the IEEE* Vol.80, 240-259.
- [67] Rauch H.E. (1963). Solutions to the linear smoothing problem. *IEEE Transactions on Automatic Control*, Vol.8, 371-372.
- [68] Aravkin A. , Burke J.V., Lennart, L. (2017). Generalized Kalman smoothing: Modeling and algorithms. *Automatica Journal of IFCA*, 86, 43-66.
- [69] Schmid, P., J. & Sesterhenn, J. (2008). Dynamic mode decomposition of numerical and experimental data. In *61st Annual Meeting of the APS Division of Fluid Dynamics*. American Physical Society.

- [70] Jonathan, H., Tu., Rowley, C.W., Luchtenburg, D., M., Brunton, S.L., Kutz, J.N. (2014). On Dynamic Mode Decomposition : Theory and Applications. *Journal of Computational Dynamics*, 1 (2), 391-421.
- [71] Taira, K., Brunton, S.L., Dawson, S.T.M., Rowley, C.W., Colonius, T., McKeon, B.J., Schmidt, O.T., Gordeyev, S., Theofilis, V. & Ukeiley, L.S. (2017). Modal Analysis of Fluid Flows: An Overview. *AIAA Journal* 55(12), 4013–4041.
- [72] Kutz, J., N., Brunton, S., L., Brunton, B. W. & Proctor, J. L. (2016). Dynamic mode decomposition: Data-driven modeling of complex systems. SIAM.
- [73] Schmidt, P. (2010). Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics* , Vol. 656, 5-28.
- [74] Kou, J., Zhang, W. (2017). An improved criterion to select dominant modes from dynamic mode decomposition. *European Journal of Mechanics - B/Fluids* Vol. 62, 109-129.
- [75] Dylewsky, D., Tao, M. & Kutz N. (2019). Dynamic mode decomposition for multiscale nonlinear physics. *Physical review. E*, Vol.99.
- [76] Filho, E., Dos Santos, P.L. (2019). A Dynamic Mode Decomposition approach with Hankel blocks to forecast multi-channel temporal series. *Journal to Conference paper* Vol.3 (3), 739-744.
- [77] Keisuke, F., Takeishi, N., Kibushi, B., Kouzaki, M. & Kawahara, Y. (2019). Data-driven spectral analysis for coordinative structures in periodic human locomotion. *Scientific reports* ) 9, 16755.
- [78] Lumley, J., L. (1967) . The structure of inhomogeneous turbulent flows. *Atmospheric turbulence and radio wave propagation*.
- [79] Weiss, J. (2019). A Tutorial on the Proper Orthogonal Decomposition. AIAA 2019-3333
- [80] Kerschen, G., Golinval, J.C., Vakakis, A.F., Bergman, L.A. (2020). The Method of Proper Orthogonal Decomposition for Dynamical Characterization and Order Reduction of Mechanical Systems: An Overview. *Journal of Applied Nonlinear Dynamics*, 9(1), 71-91.
- [81] Wax, M. and Kailath, T. (1985). Detection of signals by information theoretic criteria. *IEEE Transactions on Acoustics, Speech and Signal Processing*, Vol.33, 387–392.

- [82] Zhao Wua, Laurence, D., Utyuzhnikov, S., Afgana, I. (2019). Proper orthogonal decomposition and dynamic mode decomposition of jet in channel cross-flow. *Nuclear Engineering and Design*, Vol. 344, 54-68.
- [83] Graham, M., D. & Kevrekedis, I., G. (1996). Alternative approaches to the Karhunen–Loeve decomposition for model reduction and data analysis', *Computers and Chemical Engineering*, Vol.20, 495–506.
- [84] Bayly, P., V., Johnson, E., E., Wolf, P., D., Smith, W., M., & Ideker, R., E. (1995). Predicting patterns of epicardial potentials during ventricular fibrillation. *IEEE Transactions on Biomedical Engineering*, Vol.42, 898–907.
- [85] Epureanu, B., I., Hall, K., C., & Dowell, E., H. (2001). Reduced-order models of unsteady viscous flows in turbo-machinery using viscous-inviscid coupling', *Journal of Fluids and Structures* Vol.15, 255–273.
- [86] Barnston, A., G. & Ropelewski, C., F. (1992). Prediction of ENSO episodes using canonical correlation analysis. *Journal of Climate*, Vol.5, 1316–1345.
- [87] Leen, T., K., Rudnick, M. & Hammerstrom, R. (1990). Hebbian feature discovery improves classifier efficiency. *Proceedings of the IJCNN, IEEE, Piscataway*, 51–56.
- [88] Fitzsimons, P., M. & Rui, C. (1993). Determining low dimensional models of distributed systems. *Advances in Robust and Nonlinear Control Systems, ASME DSC*, 53.
- [89] Kreuzer, E. & Kust, O. (1996). Analysis of long torsional strings by proper orthogonal decomposition. *Archive of Applied Mechanics*, Vol.67, 68–80.
- [90] Azeez, M., F., A. & Vakakis, A., F. (2001). Proper orthogonal decomposition of a class of vibroimpact oscillations. *Journal of Sound and Vibration*, Vol.240, 859–889.
- [91] Tumer, I., Y., Wood, K., L. & Busch-Vishniac, I., J. (2000). Monitoring of signals from manufacturing processes using K-L transform, *Mechanical Systems and Signal Processing*, Vol.14, 1011–1026.
- [92] Lenaerts, V., Kerschen, G. & Golinval, J. C. (2003). Identification of a continuous structure with a geometrical non-linearity, Part II: Proper orthogonal decomposition. *Journal of Sound and Vibration*, Vol.262, 907–919.

- [93] Quaranta, G., Mantegazza, P. & Masarati, P. (2004). Assessing the local stability of periodic motions for large multi-body non-linear systems using proper orthogonal decomposition. *Journal of Sound and Vibration*, Vol.271, 1015–1038.
- [94] Ghanem, R. and Spanos, P. (1991). *Stochastic Finite Elements: A Spectral Approach*, Springer.
- [95] Mendez, M.A, Balabane, M. & Buchlin, J.M. (2018). Multi-scale proper orthogonal decomposition (mPOD). *AIP Conference Proceedings* 1978.
- [96] Mendez, M.A, Hess, D., Watz, B.B. & Buchlin, J.M. (2019) . Multi-scale Proper Orthogonal Decomposition (mPOD) of TR-PIV data - a Case Study on Stationary and Transient Cylinder Wake Flows. *CODEN: MSTCEP Measurement science & technology*, Vol.31 (9), 94014.
- [97] Mendez, M.A, Balabane, M. & Buchlin, J.M. (2019). Multi-scale proper orthogonal decomposition of complex fluid flows. *Journal of Fluid Mechanics* 870, 988-1036.
- [98] Smith, T.R., Moehlis, J.; Holmes, F. (2005). Low-Dimensional Modelling of Turbulence Using the Proper Orthogonal Decomposition: A Tutorial. *Nonlinear dynamics*, Vol.41 (1),275-307.
- [99] Brunton, S.L., Kutz, J.N., Fu Xing, Grosek, J. (2016). Dynamic Mode Decomposition for Robust PCA with Applications to Foreground/Background Subtraction in Video Streams and Multi-Resolution Analysis.
- [100] Arbabi, H. & Mezi, I. (2017). Ergodic theory, Dynamic Mode Decomposition and Computation of Spectral Properties of the Koopman operator. *SIAM Journal on Applied Dynamical Systems*, 2096-2126.
- [101] Chen,K.K., Jonathan,H.Tu & Rowley, C.W. (2011). Variants of dynamic mode decomposition: boundary conditions, Koopman, and Fourier analyses. *Journal of Nonlinear Science* Vol.22, 887–915.
- [102] Krake,T., Weiskopf, D. & Eberhardt, B. (2019). *Dynamic Mode Decomposition: Theory and Data Reconstruction*.
- [103] Muld, T., W., Efrainsson, G. & Henningson, D., S. (2012) . Flow structures around a high-speed train extracted using proper orthogonal decomposition and dynamic mode decomposition. *Comput. Fluids*, Vol. 57, 87–97.

- [104] DUKE, D., Honnery, D. & Soria, J. (2012). Experimental investigation of non-linear instabilities in annular liquid sheets. *J. Fluid Mech.*, Vol. 691, 594–604.
- [105] Grilli, M. , Schmid, P., J. , Hickel, S., & Adams, N., A. (2012). Analysis of unsteady behaviour in shock-wave turbulent boundary layer interaction. *J. Fluid Mech.*, Vol. 700, 16–28.
- [106] Schmid, P., J. (2011). Application of the dynamic mode decomposition to experimental data. *Exp.Fluids*, Vol.50(4), 1123–1130.
- [107] Schmid, P., J., Li, L, Juniper, M. P. & Pust, O. (2011). Applications of the dynamic mode decomposition. *Theor. Comp. Fluid Dyn.*, Vol.25(1–4), 249–259.
- [108] Schmid, P., Violato, J., D. & Scarano, F. (2012). Decomposition of time-resolved tomographic PIV. *Exp. Fluids*, Vol.52(6), 1567–1579.
- [109] Semeraro, O., Bellani, G. & Lundell, F. (2012). Analysis of time-resolved PIV measurements of a confined turbulent jet using POD and Koopman modes. *Exp. Fluids*, Vol.53(5), 1203–1220.
- [110] Tu., J., H. & Rowley, C., W. (2012). An improved algorithm for balanced POD through an analytic treatment of impulse response tails. *J. Comput. Phys.*, Vol.231(16), 5317–5333.
- [111] Duke, D., Soria, J. & Honnery, D. (2012). An error analysis of the dynamic mode decomposition. *Exp. Fluids*, Vol.52(2), 529–542.
- [112] Chen, K., K., Tu, J., H. & Rowley, C., W. (2012). Variants of dynamic mode decomposition: Boundary condition, Koopman, and Fourier analyses. *J. Nonlinear Sci.*, Vol.22, 887–915.
- [113] Goulart, P., J., Wynn, A. & Pearson, D. (2012). Optimal mode decomposition for high dimensional systems. In *Proceedings of the 51st IEEE Conference on Decision and Control*.
- [114] Brunton, BW., Johnson, LA., Ojemann, JG., Kutz JN. (2016). Extracting spatial-temporal coherent patterns in large-scale neural recordings using dynamic mode decomposition. *Journal of Neuroscience Methods* Vol.258, 1-15.
- [115] Arzani, A., Dawson, STM. (2021). Data-driven cardiovascular flow modelling: examples and opportunities. *Journal of Royal Society Interface*, Vol.18.