



## **Model Predictive Control for Wake Steering: a Koopman Dynamic Mode Decomposition Approach**

**Nassir Rodrigues Cassamo**

Thesis to obtain the Master of Science Degree in  
**Mechanical Engineering**

Supervisors: Prof. Jan-Willem van Wingerden  
Prof. João Miguel da Costa Sousa

### **Examination Committee**

Chairperson: Prof. Carlos Frederico Neves Bettencourt da Silva  
Supervisor: Prof. Jan-Willem van Wingerden  
Members of the Committee: Prof. Miguel Afonso Dias de Ayala Botto  
Prof. Paulo Jorge Coelho Ramalho Oliveira

**October 2020**



*Seeing small pieces of a larger jigsaw puzzle in isolation, no matter how hi-def the picture, is insufficient to grapple with humanity's greatest challenges*

- David Epstein



# **ACKNOWLEDGMENTS**

---

Hardly is success the work of one person alone. Nobody exists in a vacuum and nobody succeeds in one either. The work in this thesis is no exception. It is the combination of the support of several different individuals, the compounded effect of five years of hard work and serendipitous and fortunate events.

First and foremost, I would like to thank my parents. To my mother, for having taught me and showed me first hand the importance of values such as integrity, generosity and respect. To my father, who has repeatedly gone to great lengths and placed his interests in second so that I could have a worthily education.

Second, to thank all professors and academia personnel for their immeasurable support. To my supervisor and professor Jan Willem for having made the time for all weekly meetings and for always pointing in the right direction. It has been an enormous privilege to conduct research alongside such a brilliant professor. In spite of all obstacles it gave me an immense felling of enjoyment to work on this topic with you. To professor Carlo Bottasso, who sparked my interest in wind farm control initially during my time at the Technical University of Munich. Thank you for your enthusiasm and passion during your lectures, I am sure it resonated with all students in the hall. To my supervisor professor João Sousa for introducing me and facilitating the process of working for an outside department. To Bart, Maarten, Joeri, Daan and all PhDs and personnel in the Delft Department for Systems and Control, for the interesting discussions, feedback and presentations of their work, which made the learning curve even steeper.

Third, to thank all my friends and colleagues. To Henrique, Francisco, Pedro and Nuno, for putting up with my lousy workaholism habits during the last five years. Also, to thank Miguel, with whom I have ridden through the frightening halls of Instituto Superior Técnico in the last years. It was a fun ride.



# RESUMO

---

Turbinas eólicas são usualmente agrupadas em quintas eólicas dadas as vantagens económicas. No entanto, as restantes turbinas a jusante beneficiam de menos energia à medida que a primeira a extrai do vento. As atuais práticas da indústria otimizam o desempenho da turbina eólica ao nível individual, desprezando a interação aerodinâmica, conduzindo a perdas energéticas na quinta. A ênfase no controlo de quintas é cada vez maior. Contudo, esta tarefa evidencia-se desafiante, dado as quintas serem sistemas de elevadas dimensões cuja dinâmica é governada por equações diferenciais parciais não lineares sem solução analítica conhecida. O desenvolvimento de controladores torna-se assim computacionalmente dispendioso e inadequado para implementação em tempo real em simuladores de alta fidelidade. Os modelos de dimensão reduzida constituem uma alternativa para o desenvolvimento de controladores cooperativos. A presente tese faz uso de um algoritmo inovador no contexto da modelação de quintas eólicas - *Input Output Dynamic Mode Decomposition* - e da teoria da dinâmica de sistemas não lineares - *Koopman Operator* - com o fim de encontrar modelos adequados para o desenvolvimento de controladores preditivos. A estratégia de redirecionamento da esteira por desalinhamento do rotor é estudada e constatou-se que um modelo de espaço de estados com 37 estados é capaz de reproduzir a dinâmica da produção energética da segunda turbina, explicando 88% da sua variância e reconstruir a esteira com um erro normalizado de 4%. Controladores são desenvolvidos e implementados num simulador de alta fidelidade para se obter um nível de produção energética pré-determinado.

**Palavras-chave:** Controlo de Quintas Eólicas, Controlo da Direção da Esteira, *Dynamic Mode Decomposition*, Teoria do Operador de Koopman, Controlo Preditivo.



# ABSTRACT

---

Sitting wind turbines together in wind farms is economically advantageous. However, as the first turbine extracts energy from the wind, less power is available for downstream turbines. Current industry practices neglect the aerodynamic interaction, optimizing only at the individual turbine level, which leads to suboptimal behaviour of the total wind farm. Controlling wind farms as a whole is becoming increasingly important. Nevertheless, due to the fact that wind farms are high order systems whose dynamics are governed by nonlinear partial differential equations with no known analytical solution, the design and implementation of numerical optimal controllers in high fidelity simulators becomes computationally expensive and unsuitable for real time usage. Reduce order state models provide a possible route to the design and implementation of practical cooperative wind farm controllers. This thesis makes use of an innovative algorithm in the context of wind farm modelling - Input Output Dynamic Mode Decomposition - and the ideas of non linear dynamical system theory - the Koopman Operator - to find suitable reduced order models to be used for model predictive control. The wind farm control strategy of wake redirection control is studied. It is shown that a reduced state space model with 37 states can accurately reproduce the downstream turbine generator power dynamics with a variance accounted for of 88%, rebuild the upstream turbine wake with an average normalized root mean squared error of 4% and that controllers can be designed and implemented in a high fidelity simulator for a collective power reference tracking problem.

**Keywords:** Wind Farm Control, Wake Redirection Control, Dynamic Mode Decomposition, Koopman Operator Theory, Model Based Predictive Control.



# CONTENTS

---

Acknowledgments . . . . .	v
Resumo . . . . .	vii
Abstract . . . . .	ix
List of Tables . . . . .	xiii
List of Figures . . . . .	xv
Nomenclature . . . . .	xix
Glossary . . . . .	xxii
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Objectives . . . . .	2
1.3 Thesis Outline . . . . .	3
<b>2 Controlling and Modelling</b>	<b>5</b>
2.1 Controlling Wind Turbines and Wind Farms . . . . .	5
2.1.1 Wind Turbines . . . . .	5
2.1.2 Wind Farms . . . . .	9
2.2 Modelling Wind Turbines and Wind Farms . . . . .	14
2.2.1 Wind Farm Models . . . . .	14
2.2.2 Governing Dynamics . . . . .	15
2.2.3 Reduced Order Models . . . . .	16
<b>3 Dynamic Mode Decomposition</b>	<b>19</b>
3.1 Introduction to Dynamic Mode Decomposition . . . . .	19
3.2 Dynamic Mode Decomposition Architecture in Fluid Mechanics . . . . .	21
3.3 Dynamic Mode Decomposition Algorithms . . . . .	23
3.3.1 Dynamic Mode Decomposition . . . . .	23
3.3.2 Dynamic Mode Decomposition with Control . . . . .	25
3.3.3 Input-Output Dynamic Mode Decomposition . . . . .	27
3.3.4 Extended Input-Output Dynamic Mode Decomposition . . . . .	28
3.4 Regularization . . . . .	33

<b>4 Wind Farm Model for Yaw Control</b>	<b>35</b>
4.1 Modelling . . . . .	35
4.1.1 Step One: Simulation Design . . . . .	36
4.1.2 Step Two: Simulation and Data Record . . . . .	38
4.1.3 Step Three: Data Pre-Processing . . . . .	41
4.1.4 Step Four: Model Selection . . . . .	42
4.1.5 Step Five: Criterion of Fit . . . . .	43
4.1.6 Step Six: Model Calculation . . . . .	44
4.1.7 Step Seven: Model Validation . . . . .	45
4.1.8 Step Eight: Final Model to Use . . . . .	53
<b>5 Model Predictive Control</b>	<b>55</b>
5.1 Linear Model Predictive Control Overview . . . . .	55
5.2 Model Predictive Control Tracking Problem Formulation . . . . .	56
5.2.1 Process Model . . . . .	56
5.2.2 Performance Index . . . . .	58
5.2.3 Constraints . . . . .	59
5.2.4 Optimisation . . . . .	60
5.2.5 Receding Horizon Principle . . . . .	61
5.3 Model Predictive Control Implementation in SOWFA . . . . .	62
5.3.1 Model Predictive Control Simulation . . . . .	62
5.3.2 Open-loop Testing in SOWFA . . . . .	64
5.3.3 Closed-loop Testing in SOWFA . . . . .	65
<b>6 Conclusions</b>	<b>69</b>
6.1 Final Conclusions . . . . .	69
6.2 Contributions . . . . .	71
6.3 Recommendations and Future Work . . . . .	71
<b>Bibliography</b>	<b>73</b>
<b>A Axial Induction Control</b>	<b>79</b>
<b>B Mathematical Operators</b>	<b>93</b>
B.1 Linear Algebra . . . . .	93
B.2 Vector Calculus . . . . .	94
<b>C Main Algorithms</b>	<b>95</b>

# LIST OF TABLES

---

4.1	Validation results for IODMD <sub>u</sub> model for wake redirection control . . . . .	46
4.2	Dynamical properties of IODMD <sub>u</sub> model for wake redirection control . . . . .	49
4.3	IODMD models validation using different observables . . . . .	52
A.1	Validation results using IODMD <sub>u</sub> algorithm for axial induction control . . . . .	84
A.2	Dynamical properties of IODMD <sub>u</sub> for axial induction control . . . . .	87
A.3	IODMD models validation using different observables . . . . .	90
A.4	EIODMD models validation by augmenting with different flow field observables . . . . .	91
A.5	EIODMD models validation by augmenting with different turbine observables . . . . .	92



# LIST OF FIGURES

---

2.1	HAWT with three rotor blades . . . . .	5
2.2	HAWT main components . . . . .	5
2.3	HAWT main components and control variables . . . . .	8
2.4	HAWT main components and control variables in detail . . . . .	8
2.5	Wind turbine standard operating regions . . . . .	8
2.6	Wake behaviour travelling downstream in a wind farm . . . . .	9
2.7	The trajectory of an air particle passing through the rotor disc . . . . .	10
2.8	Flow model behind rotating wind turbine blade, incorporating wake rotation . . . . .	10
2.9	Rotor misalignment and consequent thrust force decomposition . . . . .	12
2.10	Individual Pitch Control (IPC) for Wake Redirection Control (WRC) . . . . .	12
2.11	Deflected wake of a yawed turbine and induced velocities . . . . .	13
2.12	The helix approach . . . . .	13
3.1	Schematic illustrating the Koopman operator for nonlinear dynamical systems . . . . .	30
4.1	Wind farm system to be modelled . . . . .	35
4.2	The system identification loop . . . . .	36
4.3	Two turbine set layout in SOWFA . . . . .	37
4.4	Yaw angle variation of upstream turbine during simulation for wake redirection control . . . . .	37
4.5	Generator power variations at turbine and wind farm level using wake redirection control by yaw misalignment . . . . .	38
4.6	Wake deflection dynamics reconstructed from SOWFA simulation for wake redirection control: first snapshots of absolute wake vorticity isosurface of value 5.65 . . . . .	39
4.7	Wake deflection dynamics reconstructed from SOWFA simulation for wake redirection control: second snapshots of absolute wake vorticity isosurface of value 5.65 . . . . .	39
4.8	Wake deflection dynamics reconstructed from SOWFA simulation for wake redirection control: first snapshots of streamwise and spanwise velocities at hub height . . . . .	40
4.9	Wake deflection dynamics reconstructed from SOWFA simulation for wake redirection control: second snapshots of streamwise and spanwise velocities at hub height . . . . .	40
4.10	Input-output measurements to be used for model identification for wake redirection control	42
4.11	Input-output measurements to be used for model validation . . . . .	45

4.12 Fitness of all models computed using IODMD <sub>u</sub> for wake redirection control . . . . .	46
4.13 Best performing model validation results for wake redirection control . . . . .	47
4.14 Comparison of wake reconstructed by IODMD <sub>u</sub> model and real wake from SOWFA for wake redirection control . . . . .	47
4.15 Comparison of wake reconstructed by IODMD <sub>u</sub> model and real wake from SOWFA for Wake Redirection Control: first snapshots . . . . .	48
4.16 Comparison of wake reconstructed by IODMD <sub>u</sub> model and real wake from SOWFA for Wake Redirection Control: second snapshots . . . . .	48
4.17 DMD eigenvalues of the IODMD <sub>u</sub> for wake redirection control . . . . .	50
4.18 Bode magnitude plot of IODMD <sub>u</sub> model for wake redirection control . . . . .	50
4.19 DMD modes #1, #2 and #3, as in representations of the high dimension eigenvectors of the state matrix <b>A</b> . . . . .	51
4.20 DMD modes #7, #8 and #9, as in representations of the high dimension eigenvectors of the state matrix <b>A</b> . . . . .	51
4.21 DMD modes #11, #15 and #20, as in representations of the high dimension eigenvectors of the state matrix <b>A</b> . . . . .	52
4.22 Comparison of frequency response of different IODMD models for wake redirection control	53
 5.1 The 'Moving horizon' in predictive control . . . . .	61
5.2 Predictive controller performance for collective power reference tracking . . . . .	62
5.3 Predictive controller performance for collective power reference tracking under different controller settings . . . . .	63
5.4 Open-loop predictive controller performance for collective power reference tracking implemented in SOWFA . . . . .	64
5.5 Open-loop implementation of MPC in SOWFA for tracking problem of a collective power reference . . . . .	65
5.6 MPC closed loop implementation schematics . . . . .	66
5.7 Closed-loop predictive controller performance for collective power reference tracking implemented in SOWFA . . . . .	67
5.8 Closed-loop predictive controller performance for collective power reference tracking implemented in SOWFA: analysis at the turbine level . . . . .	67
5.9 Closed-loop predictive controller performance regarding control action effort . . . . .	68
5.10 Closed-loop predictive controller performance for collective power reference tracking implemented in SOWFA: analysis at the turbine level with further restrictions on yaw rate . . . . .	68
 A.1 Collective pitch angle variation of upstream turbine during simulation . . . . .	81
A.2 Generator power variations at turbine and wind farm level using axial induction control by collectively pitching the blades . . . . .	81
A.3 Wake deflection dynamics reconstructed from SOWFA simulation for axial induction control: first snapshots of absolute wake vorticity isosurface . . . . .	82

A.4	Wake deflection dynamics reconstructed from SOWFA simulation for axial induction control: second snapshots of absolute wake vorticity isosurface . . . . .	82
A.5	Wake deflection dynamics reconstructed from SOWFA simulation for axial induction control: snapshots velocity field at downstream turbine rotor plane at different time instants . . . . .	83
A.6	Input-output measurements to be used for model identification for axial induction control .	83
A.7	Input-output measurements to be used for model validation for axial induction control . .	84
A.8	Fitness of all models computed using $\text{IODMD}_u$ for axial induction control . . . . .	85
A.9	Best performing model validation results for axial induction control . . . . .	85
A.10	Comparison of wake reconstructed by $\text{IODMD}_u$ model and real wake from SOWFA for axial induction control: first snapshots . . . . .	86
A.11	Comparison of wake reconstructed by $\text{IODMD}_u$ model and real wake from SOWFA for axial induction control: second snapshots . . . . .	86
A.12	DMD eigenvalues of the $\text{IODMD}_u$ model for axial induction control . . . . .	88
A.13	Bode magnitude plot of LTI $\text{IODMD}_u$ model for axial induction control . . . . .	88
A.14	DMD modes #1, #2 and #4 as in representations of the high dimensional eigenvectors of the state matrix <b>A</b> . . . . .	89
A.15	DMD modes #6, #7 and #8 as in representations of the high dimensional eigenvectors of the state matrix <b>A</b> . . . . .	89
A.16	DMD modes #11, #15 and #16 as in representations of the high dimensional eigenvectors of the state matrix <b>A</b> . . . . .	89
A.17	Comparison of $\text{IODMD}_u$ and $\text{EIODMD}_{u,w}$ model's performance using VAF for axial induc- tion control . . . . .	91
A.18	Comparison of $\text{IODMD}_u$ and $\text{EIODMD}_{u,w}$ model's response in the frequency domain for axial induction control . . . . .	92
C.1	MALTAB program structure . . . . .	98



# NOMENCLATURE

---

## Dynamic Mode Decomposition

$\Lambda$	Eigenvalues of reduced order state matrix
$\Omega$	State and control snapshot matrix
$\Phi$	DMD mode
$\Sigma$	Matrix of singular values of $\mathbf{X}$
$\Theta$	Least square solution of subspace identification problem
$\Upsilon$	Control input snapshot matrix
$u$	Input vector
$v$	Koopman mode
$x$	High dimension state vector
$y$	Output vector
$\hat{\Sigma}$	Matrix of singular values of $\mathbf{X}'$
$\hat{\mathbf{U}}$	Left singular vectors of $\mathbf{X}'$
$\hat{\mathbf{V}}$	Right singular vectors of $\mathbf{X}'$
$\mathcal{K}$	n operator
$\mathbf{W}$	Eigenvectors of reduced order state matrix
$\tilde{\Sigma}$	Matrix of singular values of $\Omega$
$\tilde{x}$	Low dimension state vector
$\tilde{\mathbf{A}}$	Discrete time reduced order state matrix
$\tilde{\mathbf{B}}$	Discrete time reduced order input matrix
$\tilde{\mathbf{C}}$	Discrete time reduced order output matrix
$\tilde{\mathbf{U}}$	Left singular vectors of $\Omega$

$\tilde{\mathbf{V}}$	Right singular vectors of $\Omega$
$\varphi$	Koopman eigenfunction
$g$	Observable function
$l$	Dimension of the output vector
$n$	Dimension of the high order state vector
$q$	Dimension of the input vector
$r$	Dimension of the reduced order state vector
<b>A</b>	Discrete time high order state matrix
<b>B</b>	Discrete time high order input matrix
<b>C</b>	Discrete time high order output matrix
<b>D</b>	Discrete time feedthrough matrix
<b>G</b>	Linear dynamics matrix representation of states and inputs
<b>Q</b>	Projection subspace
<b>U</b>	Left singular vectors of $\mathbf{X}$
<b>V</b>	Right singular vectors of $\mathbf{X}$
<b>X'</b>	Time shifted snapshot matrix
<b>X</b>	Snapshot matrix
<b>Y</b>	Output snapshot matrix

### Fluid Dynamics

$\bar{u}$	Mean streamwise velocity
$\bar{v}$	Mean spanwise velocity
$\bar{w}$	Mean vertical velocity
$\tau_{ij}$	Viscous stress tensor
$\mathbf{g}$	Gravitational acceleration vector
$\mathbf{V}$	Velocity vector
$\mu$	Viscosity coefficient
$p$	Hydrostatic pressure
$u$	Streamwise velocity

$u'$	Streamwise velocity fluctuation
$v$	Spanwise velocity
$v'$	Spanwise velocity fluctuation
$w$	Vertical velocity
$w'$	Vertical velocity fluctuation

## Mathematics

*	Conjugate transpose of matrix
$\nabla \cdot$	Vector divergent operator
$\nabla$	Vector gradient operator
$\dagger$	Moore Penrose matrix inverse
$T$	Transpose of matrix

## Model Predictive Control

$\bar{\mathbf{P}}$	Predicted outputs over predictiton horizon
$\bar{\mathbf{U}}$	Predicted input over control horizon
$\Lambda$	Linear inequality constraints vector
$x^e$	Extended state vector
$\Delta u_k$	Incremental input vector
$\Delta u_k^e$	Extended incremental input vector
$\hat{\mathbf{P}}$	Collective predicted power over predictiton horizon
$f$	Quadratic objective function vector
$\mathcal{H}$	Quadratic objective function hessian matrix
$\mathbf{A}_e$	Discrete time extended state matrix
$\mathbf{B}_e$	Discrete time extended input matrix
$\mathbf{C}_e$	Dicrete time extended output matrix
$\mathbf{M}$	Linear inequality constraints matrix
$\mathbf{Q}$	Reference tracking penalty
$\mathbf{R}$	Control action penalty
$H_c$	Control horizon

$H_p$  Prediction horizon

$J$  Cost function

### Systems Identification

$\hat{x}$  Predicted state

$\hat{y}$  Predicted output

$\omega_n$  Natural frequency

$\theta$  Model parameters

$\xi$  Damping ratio

$N$  Number of samples

$T_s$  Sampling rate

$x$  State

$y$  Measured output

### Wind Turbines

$\gamma$  Yaw angle

$\rho$  Air density

$\tau$  Generator torque

$\theta$  Blade pitch angle

$A$  Rotor disk area

$a$  Axial induction factor

$C_p$  Power coefficient

$C_t$  Thrust coefficient

$P$  Rotor power

$T$  Thrust

$U$  Wind velocity

$U_r$  Wind speed at the rotor

$U_\infty$  Free wind velocity

# ACRONYMS

---

<b>ABL</b>	Atmospheric Boundary Layer.
<b>ADM</b>	Actuator Disk Model.
<b>AIC</b>	Axial Induction Control.
<b>ALM</b>	Actuator Line Model.
<b>CFD</b>	Computational Fluid Dynamics.
<b>DIC</b>	Dynamic Induction Control.
<b>DIO</b>	Direct Input Output.
<b>DIPC</b>	Dynamic Individual Pitch Control.
<b>DMD</b>	Dynamic Mode Decomposition.
<b>DMDc</b>	Dynamic Mode Decomposition with Control.
<b>DOF</b>	Degrees Of Freedom.
<b>EDMD</b>	Extended Dynamic Mode Decomposition.
<b>EIODMD</b>	Extended Input Output Dynamic Mode Decomposition.
<b>FAST</b>	Fatigue Aerodynamics Structures and Turbulence.
<b>HAWT</b>	Horizontal Axis Wind Turbine.
<b>IIO</b>	Increment Input Output.
<b>IODMD</b>	Input-Output Dynamic Mode Decomposition.
<b>IPC</b>	Individual Pitch Control.
<b>IPCC</b>	Intergovernmental Panel on Climate Change.
<b>LCOE</b>	Levelized Cost of Electricity.

<b>LES</b>	Large Eddy Simulation.
<b>LPV</b>	Linear Parameter Varying.
<b>LQG</b>	Linear Quadratic Gaussian.
<b>LQR</b>	Linear Quadratic Regulator.
<b>LTI</b>	Linear Time Invariant.
<b>MBPC</b>	Model Based Predictive Control.
<b>MPC</b>	Model Predictive Control.
<b>NPDE</b>	Nonlinear Partial Differential Equations.
<b>NREL</b>	National Renewable Energy Laboratory.
<b>NRMSE</b>	Normalized Root Mean Squared Error.
<b>NS</b>	Navier Stokes.
<b>POD</b>	Proper Orthogonal Decomposition.
<b>QP</b>	Quadratic Programming.
<b>RANS</b>	Reynold Averaged Navier Stokes.
<b>RBS</b>	Randomized Binary Signals.
<b>ROM</b>	Reduced Order Model.
<b>SOWFA</b>	Simulator for Offshore Wind Farm Applications.
<b>SVD</b>	Singular Value Decomposition.
<b>TEM</b>	Topical Expert Meeting.
<b>VAF</b>	Variance Accounted For.
<b>WECS</b>	Wind Energy Conversion System.
<b>WRC</b>	Wake Redirection Control.

# GLOSSARY

---

**coherent structure** repetitive pattern seen in a turbulent fluid flow, maintaining the same size/shape through time, in spite of the whole flow being unsteady and not repeating itself in detail.

**computational fluid dynamics** (or CFD) is a branch of fluid mechanics that uses numerical methods and algorithms to solve problems that involve fluid flows.

**dynamic mode** (or DMD mode) is an eigenvector of the best-fit DMD operator  $\mathbf{A}$ . These modes are spatially coherent and oscillate in time at a fixed frequency and a growth or decay rate.

**dynamic mode decomposition** (or DMD) corresponds to the best fit linear operator that propagates a data matrix into its future correspondent. The eigenvectors of the linear operator are referred to as dynamical or DMD modes and the corresponding eigenvalues determine the dynamics of these modes such as growth or decay rates.

**dynamical system** mathematical model for the dynamic evolution of a system. It is often formulated in terms of ordinary differential equations on a state-space. The resulting equations may be linear or nonlinear and may also include the effect of actuation inputs and represent outputs as sensor measurements of the state.

**equation free modelling** (or EFM) consists in characterizing a dynamical system using solely data measurements, as opposed to starting with the knowledge of the governing equations.

**kalman filter** an estimator that reconstructs the full state of a dynamical system from measurements of a time series of the sensor outputs and actuation inputs. A Kalman filter is itself a dynamical system that is constructed for observable systems to stably converge to the true state of the system and is optimal for linear systems with Gaussian process and measurement noise of a known magnitude.

**koopman operator** an infinite dimensional linear operator that propagates measurement functions from an infinite-dimensional space through a dynamical system.

**least squares regression** a regression technique where a best-fit line or vectors is found by minimizing the sum of squares of the error between the model and the data.

**model based predictive control** advanced control strategy that computes the optimal control action, at each time instant, to be applied to a system, given a certain objective. The computation of the

optimal control action relies on dynamical model of the process, most often linear empirical models obtained by systems identification.

**offshore wind** use of wind farms constructed in bodies of water, usually on the high seas, to harvest wind energy to generate electricity. Higher and more constant wind speeds are available offshore compared to on land, so offshore wind power's electricity generation is higher per amount of capacity installed.

**reduced order model** (or ROM) is a computationally inexpensive mathematical representations of high fidelity complex models which capture the behaviour of these source models so that a system's dominant effects may be quickly studied using minimal computational resources, offering the potential for near real-time analysis.

**simulator for offshore wind farm applications** (or SOWFA) is a high fidelity simulator for the interaction between wind turbine dynamics and the fluid flow in a plant developed by the National Renewable Energy Laboratory in the United States. It allows for the implementation of the majority of wind plant control strategies proposed in the literature.

**state space model** type of mathematical model which represents the dynamics of a system as a set of coupled first-order differential equations in a set of internal variables known as state variables, together with a set of algebraic equations that combine the state variables into physical output variables. The state variable is a quantity which, in the absence of external excitation, completely determines the evolution of the system.

**wake** flow downstream of a wind turbine whose characteristic are altered, presenting reduced flow velocity due to the extraction of energy from the flow by the turbine and increased turbulence intensity due to obstruction of the flow by the turbine.

**wake redirection control** wind farm control strategy which changes the direction of the wake so that the overlap of the wake with downstream rotors can be avoided.

# CHAPTER 1

## INTRODUCTION

---

### 1.1 Motivation

Wind energy offers a potential to reduce carbon emissions and mitigate worldwide climate change. As this technology is harvested throughout the world, many challenges arise when attempting to extract the most benefit out of wind turbines. The motivation to study power optimisation problems in wind farms (aggregation of wind turbines) emerges as a means to provide answers to the current challenges. The motivation is threefold:

#### **1. Offshore wind energy production is rapidly growing and technologies are developing.**

Renewable energy production has increased substantially in the last years, with a 5 to 6 fold increase since the 1960s, according to [59]. Wind energy is one of the fast growing technologies, representing 19% of the total renewable energy production in 2018, compared to 8% in 2010 [59]. The total installed wind power capacity in Europe has been steadily increasing since 2009, as shown by Wind Europe, a European agency [39]. Statistics indicate an **increase from 77 GW** (75 onshore and 2 offshore) **of cumulative wind power capacity in 2009 to 205 GW** (183 onshore and 22 offshore) in 2019. Last year alone, Europe installed 15.4 GW of new wind power capacity, 28% more than 2018. Onshore wind was 76% of the new installations with 11.7 GW and offshore wind installations were a record 3.6 GW. The trend for going offshore is explained by its advantages, namely the high availability of resources - wind speeds are higher - improved social acceptability, lower environmental impacts (noise) and potential for reduced Levelized Cost of Electricity (LCOE).<sup>1</sup>

#### **2. Wind energy production is a pivotal step in preventing global warming complications.**

The Summary for Policymakers [46] presents the key findings of the Special Report produced in 2018 by the Intergovernmental Panel on Climate Change (IPCC). The later begins by noting that human activities are estimated to have caused approximately 1.0°C of global warming above pre-industrial levels and that **it is likely to reach 1.5°C between 2030 and 2052 - if it continues to increase at the current rate**. The knowledge that global warming presents a serious issue has been around for more

<sup>1</sup>Represents the average revenue per unit of electricity generated that would be required to recover the costs of building and operating a generating plant (during an assumed financial life and duty cycle).

than two decades. The Kyoto Protocol, launched in 1995, was the first international tool focused on greenhouse gas mitigation with initially 37 countries pledging to reduce their emissions by 5% compared to the 1990 baseline in the period 2008-2012 [49]. The Paris Agreement is a more recent effort, bringing all nations together with a clear aim: to strengthen the global response to the threat of climate change by keeping the global temperature rise this century well below 2°C above pre-industrial levels and to pursue the efforts to limit temperature increase even further to 1.5°C. These values are based on the Special Report findings, which highlights that **a number of climate changes could be avoided by limiting global warming to 1.5°C compared to 2°C, or more**. In order to meet the agreements' targets and circumvent disastrous scenarios, countries started to adopt new measures, which **place great emphasis on energy policies**. The Kyoto Protocol, for example, states in Article 2 that countries shall implement and/or further elaborate policies such as *research on, and promotion, development and increased use of new and renewable forms of energy, of carbon dioxide sequestration technologies and of advanced and innovative environmentally sound technologies* [54].

### **3. It is an active and promising research topic which tackles wind farms intrinsic challenges.**

In spite of the economic advantages of sitting wind turbines together in wind farms, **the aerodynamic interaction between turbines may have a negative effect on the total electrical power production** and may increase the loads experienced by turbines. The aerodynamic interaction effects are caused by the turbine wakes, which are the flow structures that form behind each turbine. Mitigation of these effects can be achieved either by optimizing the wind farm layout and/or by using **wind plant control techniques** during the operation of the wind plant. The later concept aims at improving the performance of the wind plant as a whole by **maximizing the power output of the plant instead of each individual turbine** (which can most often be suboptimal). It was firstly introduced in 1988 in [65], showing that slight increases in the overall energy capture are possible. Since then, various control strategies have been studied in depth, resorting to simulations, wind tunnel experiments and real wind farm tests. More recently, **dynamic wind farm control** has gained interest in the scientific field. The later is based on varying the control input through time, which differs from steady-state optimal control. The underlying hypothesis is that time-varying inputs are able to purposely influence the inherently dynamic nature of the wind. The potential of such strategies has been shown in high-fidelity simulations and more recently, in 2019, wind tunnel experiments [26]. Moreover, new control strategies are being developed based on these new novel concepts, such as the helix approach, where the wake presents a helicoidal shape [25]. Understanding optimal dynamical controlling techniques is still an on-going research topic and a fundamental one in order to achieve meaningful power production levels.

## **1.2 Objectives**

The research objective of this thesis is to **model how the power production varies in a set of two turbines interacting given a certain wind farm control strategy**, so that advanced controllers can be designed, making possible the computation of the optimal control action which maximizes the power

production at the wind farm level at each instant. Following this strategy, no restriction regarding steady state control is imposed, thus allowing time varying inputs. Three clear objectives can be noted:

**Thesis objective 1:** Model the wind turbines power dynamics **using a novel technique in the context of wind energy, namely Input Output Dynamic Mode Decomposition.** This method and its variants allow both to derive an accurate mapping for input-output information and perform the reconstruction of the full flow field, shedding light on the dynamical properties of the fluid - the wind - when excited by a certain control strategy.

**Thesis objective 2:** Design **model predictive controllers** and test them in a high fidelity simulator in order to achieve power production gains at the wind farm level, thus evaluating the ability of using pure data driven techniques - dynamic mode decomposition in this work - to obtain models suitable for control. **The controllers are tested for a reference tracking problem** so that 2 turbines work together by taking into account the aerodynamic interaction to achieve a pre-specified collective power production level.

**Thesis objective 3:** Develop a series of algorithms and a MATLAB program that is **versatile enough to handle different data sets corresponding to different wind farm control strategies.** In this way, models can be derived straightforwardly so as to gain insights on the dynamics and/or design controllers. The usefulness of these algorithms is shown in detail using data from high fidelity simulations where wind farm control strategies are used, with a focus on yaw control.

## 1.3 Thesis Outline

This thesis is divided into six distinct chapters:

1. **Chapter 1 - Introduction:** the motivations underpinning the study of wind farm power optimisation are presented, along with the objectives this work strives to achieve.
2. **Chapter 2 - Controlling and Modelling:** the task of controlling wind turbines and wind farms is introduced. The main control parameters used to maximize power production at the wind turbine level are explained and how these can be used to devise control strategies that focus on maximizing the collective power production of wind turbines, as opposed to the individual level, are presented. The chapter focuses also on the task of modelling, by presenting the available mathematical tools to capture the dynamical behaviour of a wind farm which have been famously used for the last decades.
3. **Chapter 3 - Dynamic Mode Decomposition:** the cornerstone method to derive wind farm models in the context of this work is presented. The standard Dynamic Mode Decomposition (DMD), conceived in 2008, is introduced, as well as its adaptions for the purposes of control, namely Dynamic Mode Decomposition with Control (DMDc) and Input-Output Dynamic Mode Decomposition (IODMD), developed in 2015 and 2016. The mathematical connection to non linear dynamical system theory is briefly described.

4. **Chapter 4 - Wind Farm Model for Yaw Control:** the model for wind farm control is presented by using the system identification loop. A high fidelity simulation is performed to gather input-output data, the models are derived according to the DMD framework and then validated using pre-defined criteria. The final model is then studied in depth so as to better understand the dynamical properties of the system.
5. **Chapter 5 - Model Predictive Control:** a model based predictive controller is designed. The various parameters are fine tuned and simulations performed. An observer is conjointly designed so that in the final setup the states are corrected based on the estimated output and real output. The final controller is tested in a high fidelity simulator and the results for the collective power reference tracking presented.
6. **Chapter 6 - Conclusions:** conclusions based on the results are presented, and further research pathways devised.

As this dissertation combines concepts from different domains of science - linear algebra, systems identification, optimal control, dynamical systems theory, fluid mechanics, wind turbines and wind farm control - a glossary with key concepts has been provided with straightforward summaries of the most important ones. Furthermore the most relevant mathematical tools are described in Annex B. In addition, short footnotes are occasional used for simpler concepts.

The following notation is used throughout this thesis: vectors and matrices are printed in a boldface font (e.g.,  $\mathbf{A}$ ,  $\Sigma$ ). Vectors are usually denoted by lower cases (e.g.,  $g$ ) and matrices by a capital letter (e.g.,  $\mathbf{G}$ ).

The work developed within the context of this thesis will lead to the submission of the results to a conference [17] and journal [18]. In [17] focus is given to the development of a predictive controller based on a reduced order model and implementation in a high fidelity simulator of offshore wind farms so that two turbines can work in a cooperative fashion to produce a pre specified level of power. In [18] the results concerning the using of concepts from non linear dynamical systems theory to improve on the reduced order model's capabilities to reproduce the non linear dynamics within a wind farm are emphasized.

An extensive use of the computing software MATLAB has been made use of for the purposes of this thesis. The majority of the pictures here present were obtained by codes developed in MATLAB by using data from simulations. The algorithms implemented were also written in this coding language. In order for others to use these tools, all codes along with detailed explanations have been made available in [https://github.com/nassircassamo/DMD\\_SOWFA](https://github.com/nassircassamo/DMD_SOWFA).

Whether the reader is a wind farm control enthusiast attempting to gain insights into wind farm control strategies, or a curious aficionado of the latest data driven techniques, the open disclosure of this material intends to allow for reproducible research and promote the learning and implementation of this recent mathematical techniques.

# CHAPTER 2

## CONTROLLING AND MODELLING

---

### 2.1 Controlling Wind Turbines and Wind Farms

#### 2.1.1 Wind Turbines

##### 2.1.1.1 Wind turbine components and mechanisms

A wind turbine consists of a **mechanical device specifically designed to extract kinetic energy from the wind and convert part of it into useful mechanical energy**. The most commonly produced and used wind turbine is the Horizontal Axis Wind Turbine (HAWT), represented in Figure 2.1. The main component of a HAWT is its rotor, most often equipped with three rotor blades, which is attached to the generator through a drivetrain. The generator and drivetrain are housed in the nacelle, which is supported by a tower, as schematically represented in Figure 2.2 .

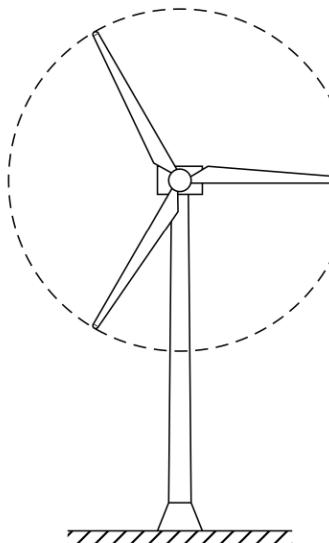


Figure 2.1: HAWT with three rotor blades, originally in [9].

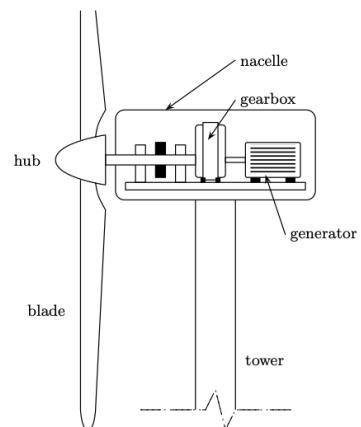


Figure 2.2: HAWT main components, originally in [9].

### 2.1.1.2 Wind turbine power extraction

The rotor blades convert the momentum of a wind field passing the rotor plane into aerodynamic forces that drive the rotor. The drivetrain transfers the aerodynamic torque from the rotor to the generator shaft, directly or through a gearbox. The initial kinetic energy in the wind is then converted to electrical power.

The available power in the wind due to its kinetic energy can be derived by evaluating the mass of wind flowing through an imaginary rotor disk of area  $A$ , which is swept by the rotor blades, and it is given by a well-known expression [34]:

$$P_w = \frac{1}{2} \rho A U^3 \quad (2.1)$$

where  $P_w$  [W] is the instantaneous kinetic power of the wind available at surface of area  $A$  [ $\text{m}^2$ ],  $\rho$  [ $\text{kg} \cdot \text{m}^{-3}$ ] is the air density which is assumed to be constant and  $U$  [ $\text{m} \cdot \text{s}^{-1}$ ] the wind velocity.

To evaluate the rotor performance in extracting energy from wind, the power coefficient  $C_P$  is used. It is defined as the dimensionless ratio of the power extracted by the rotor to the kinetic available power in the wind:

$$C_P = \frac{P}{\frac{1}{2} \rho A U^3} \quad (2.2)$$

As the air passes through the rotor disc it proceeds downstream with reduced speed, a natural consequence of the extraction of energy from the wind. **This region of the flow downstream of the turbine is called the wake.**

### 2.1.1.3 The axial induction factor

A measure of interest to characterise the drop in wind velocity between the free stream and rotor plane is the *axial induction factor*  $a$ , which is defined as the ratio of the difference between the free wind velocity  $U_\infty$  and the wind velocity at the rotor  $U_r$  to  $U_\infty$ :

$$a = \frac{U_\infty - U_r}{U_\infty} \quad (2.3)$$

allowing to relate the wind speed at the rotor  $U_r$  with the free wind velocity  $U_\infty$ :

$$U_r = U_\infty(1 - a) \quad (2.4)$$

The power coefficient can be written in terms of the axial induction factor by means of a simple model based on linear momentum theory generally attributed to Albert Betz in 1926. The model assumptions and derivation can be analysed in [35]. The power coefficient is then rewritten as:

$$C_P = 4a(1 - a)^2 \quad (2.5)$$

Taking the derivative of equation (2.5) with respect to  $a$  and equalling to zero, the maximum value of the power coefficient is found. In light of the underlying assumptions of the model, the maximum value of  $C_P$  occurs for  $a = \frac{1}{3}$  and is equal to  $C_{P_{\max}} = \frac{16}{27} = 0.5926$ . This limit is famously known as the

Betz limit. It states that, given the laws of physics, **there exists a physical limit for the power a wind turbine can capture**. Furthermore, the relation in (2.4) and the result for  $a$  of the Betz limit indicate that **if an ideal rotor were designed and operated such that the wind speed at the rotor were 2/3 of the free stream wind speed, then it would be operating at the point of maximum power production**. The task is then to modify the wind turbine's settings such that the operating point of maximum power production is reached.

#### 2.1.1.4 Wind turbine loading

The model based on linear momentum theory used for the analysis of power production can also be used to evaluate the force  $T$  of the wind on the wind turbine. This is referred to as the thrust force, and it can be non-adimensionalized to give a coefficient of thrust  $C_T$ :

$$C_T = \frac{T}{\frac{1}{2}\rho AU_\infty^2} \quad (2.6)$$

and expressed by the axial induction factor (for full derivation, please see [34]):  $C_T = 4a(1 - a)$ . The thrust force is a variable of interest, as one is not only interested in maximizing power production, but also in minimizing loading to increase the wind turbine's lifetime.

#### 2.1.1.5 Wind turbine control objectives and mechanisms

A Wind Energy Conversion System (WECS) connected to electric power networks - where a HAWT is included - must be designed to minimise the cost of supplied energy, whilst simultaneously ensuring safety, noise and power quality standards. This task, in turn, involves balancing a series of objectives, which must be pursued collectively. They include [9]:

1. **Energy capture:** maximisation of energy capture taking into account constraints, for example, on maximum rotor speed due to safety issues.
2. **Mechanical loads:** preventing WECS from excessive dynamical loads, both static (related to wind mean wind speed) and dynamic (related to wind speed spatio temporal distribution over the rotor area).
3. **Power quality:** conditioning the generated power complies with quality standards.

To guarantee that the above objectives are met, it is essential to control power production and loads exerted on the wind turbine. For such purpose, a number of Degrees Of Freedom (DOF) (control variables) are typically available:

1. **Blade pitch angle  $\theta$**  : the rotors blades can rotate, with their axis of rotation aligned with the blades, using hydraulic actuators or servo pitch motors.
2. **Generator torque  $\tau$**  : torque control is used to control the power capture. Generator torque is regulated for that purpose either by design characteristics of the grid-connected generator or independently controlled with the use of power electronic converters.

3. **Yaw angle  $\gamma$** : the nacelle can rotate, with its axis of rotation aligned with the tower, by using a yaw motor. The yaw angle is defined as the angle between the axial rotor axis and the incoming wind direction. Yaw control allows to vary the angle between the rotor plane and the incoming wind direction.

The three control variables can be seen in Figures 2.3 and in more detail in Figure 2.4.

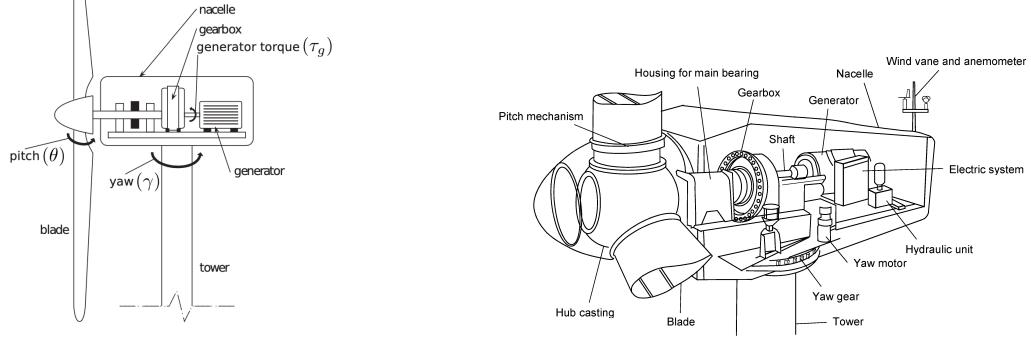


Figure 2.3: HAWT main components and control variables, taken from [10].

Figure 2.4: HAWT main components and control variables in detail.

### 2.1.1.6 Wind turbine operating regions

The different control variables play a fundamental role and their usage defines distinct operating regions for the wind turbine functioning. For single wind turbines, three regions are distinguished - represented in Figure 2.5 - each with its control strategy (the rating [maximum power production point] of the turbine arises from a compromise between available energy and manufacturing costs):

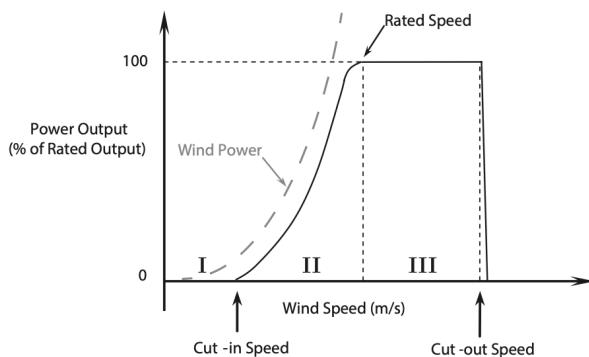


Figure 2.5: Typical wind turbine power curve and control strategies, originally in [10].

**Region 1:** Below cut-in speed, the available wind energy is too low to compensate for the operation costs and losses. **Region 2:** Below rated speed torque control is employed, and the blades pitch angle remain constant. **Region 3:** Above rated speed, pitch control is employed. The rotor's blades pitch angle is varied so that an optimal rotor speed  $\Omega$  is maintained and constant power is achieved. Above cut-out wind speed, the turbine is shut down to prevent structural overload.

## 2.1.2 Wind Farms

Wind turbines are often sited together in wind farms due to a number of advantages, such as reduced deployment costs of turbines, reduced costs of the electricity grid and reduced operation and maintenance costs. In spite of these benefits, the grouping of turbines poses a number of complications that often significantly affect their performance. Such complications arise due to the presence of the wake.

### 2.1.2.1 The wake

The **wake** refers to the altered wind flow downstream of the wind turbine and its properties depend on *time* - subject to the varying operational setting of a turbine and varying surround flow - space - the behaviour of the wake is different depending on how far downstream it is from the rotor - and *external variables* - such as temperature, which influences the behaviour of the wake. The typical wake behaviour when travelling downstream of a wind turbine is represented in Figure 2.6, and its most relevant characteristics are highlighted:

- **Reduced flow velocity:** natural consequence caused by the extraction of energy from the flow by the upstream turbine.
- **Expansion of cross sectional area:** explained by the law of mass conservation and the assumption that air density does not change (incompressibility).
- **Increased turbulence intensity:** caused by the obstruction of the flow by the turbine, and the resulting velocity gradients in the flow (shear).

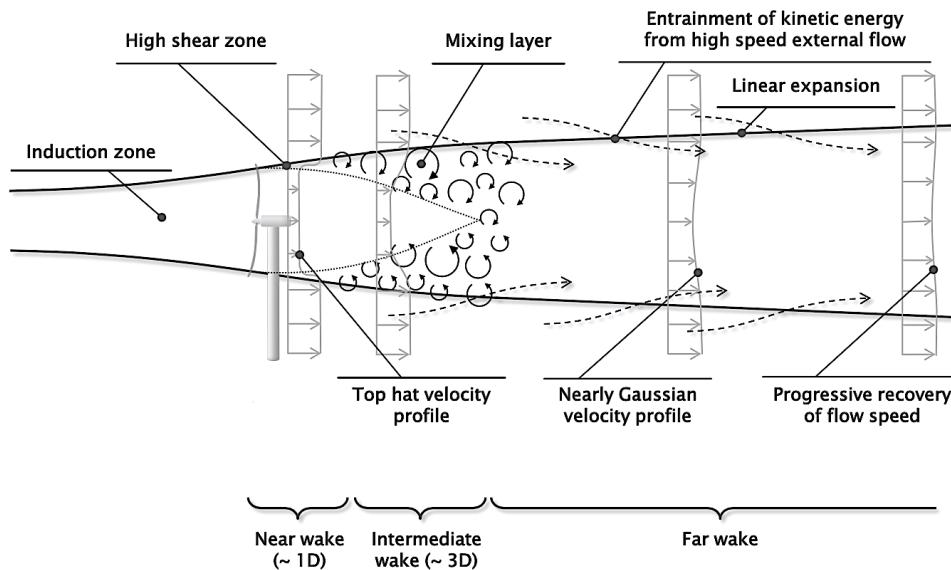


Figure 2.6: Wake behaviour when travelling downstream. Adapted from [11].

The amount of wake interaction is also dependent on the distance between turbines, as the flow in the wake will recover to the surrounding flow conditions through convection and diffusion. Such processes are promoted by the turbulence, intensifying **wake recovery by mixing of the flow in the wake and the surrounding stream**, as depicted on the right side of Figure 2.6 (progressive recovery of flow speed).

Moreover, **the wake rotates in the opposite direction to the rotor**, as represented in Figure 2.7 and Figure 2.8. Such characteristic is explained by Newton's third law of motion<sup>1</sup>: as the air passes through the rotor disk and exerts a torque, an equal and opposite torque is imposed by the rotor upon the downstream air.

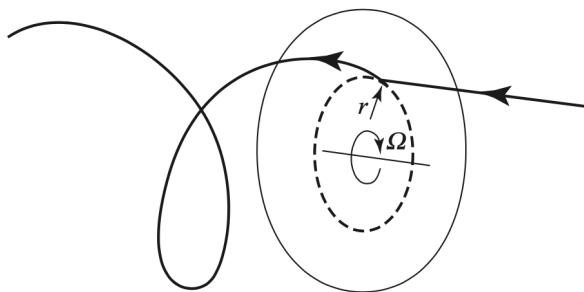


Figure 2.7: The trajectory of an air particle passing through the rotor disc [35].

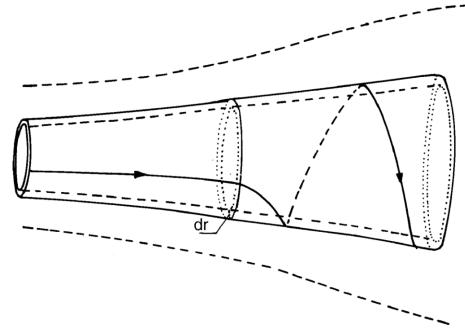


Figure 2.8: Flow model behind rotating wind turbine blade, incorporating wake rotation [34].

Different operating settings might induce other wake characteristics. As an example, in [7], for a yawed turbine (rotor is not oriented perpendicular with respect to the incoming wind), the **wake is deflected**, a phenomena where the complete wake is diverging in the perpendicular direction from the rotor centre. Moreover, the mentioned deflection increases with an increase in yaw angle.

### 2.1.2.2 Wind turbine interaction

It is through the wake that an upwind turbine can influence the performance of downwind turbines. As a result of the downstream wind velocity deficit, **downstream turbines will capture less power when compared to free-stream conditions**. Moreover, the **increased turbulence in the wake can lead to increased loading, resulting in the wind turbine shorter lifetime**.

The purpose of wind farm control is to go one step further than individual turbine control and take into account the interaction of the wake with the turbine to **ensure a specific level of performance of the wind farm**. This is achieved with a supervisory controller that acts on information provided by each individual turbine to reach a global objective.

**Wind farm control consists of finding control inputs using measurements to increase the performance of a wind farm**, thus improving quality or minimizing the cost of wind energy. The objectives of wind farm control are similar to the ones of single turbine control: (1) maximize power production, (2) minimize structural degradation and (3) improve energy quality. However, it is now more important to focus on the effects of the wake.

Currently, most wind farms are operated using individually optimal wind turbine control settings, referred to as *greedy control* as a result of neglecting wake effects [10]. **The assumption that wind farm performance can be increased by operating turbines in the farm at configurations different from their individual optimal control** settings has become an active research topic since the 1990s.

---

<sup>1</sup>Newton's third law states that for every action (force) in nature there is an equal and opposite reaction.

Two general control methods exist for this purpose: Axial Induction Control (AIC) and Wake Redirection Control (WRC), which are thoroughly discussed in the next subsections.

### 2.1.2.3 Axial Induction Control

The reasoning underpinning AIC is that by **changing the axial induction** - making use of the control variables available, blade pitch angle  $\theta$  and generator torque  $\tau$  - **downstream turbines can generate more power**. The control mechanisms influence the axial induction of the rotor and therefore also affect the velocity deficit in the wake the rotor generates. AIC is worthwhile as long as reduced power production of the upwind turbine (the first of a series of turbines) is compensated for by the downwind turbines. A natural consequence is the reduction in thrust force, which leads to reduced turbulent wake mixing and thereby reduced wake recovery. There are thus two counteracting effects: increased velocity in the near wake but reduced recovery downstream.

Despite initial promising results, recent studies with high-fidelity simulation models and scaled wind tunnel experiments indicate that **the power gain that can be achieved with steady state induction control is limited to non existing**. In [3] it is shown that the control settings that yield gain at the wind plant level are dependent on particular atmospheric conditions, the wind plant configuration and turbine characteristics. Simulations in Simulator for Offshore Wind Farm Applications (SOWFA)<sup>2</sup>, a simulator of high fidelity, further suggest the existence of certain circumstances where this strategy becomes infeasible and that the power lost on the first turbine is not regained at the second turbine. Wind tunnel tests point again in the same direction: in [16], only very limited power gains are shown, showing a modest increase in total power of circa 1% when two turbines are considered.

Alternative approaches have emerged, which focus on **time-varying control inputs which purposely influence the inherently dynamic nature of the wake**. Recent studies, namely [50, 51], have shown that the so called Dynamic Induction Control (DIC), where the induction factor is varied over time, improving power production in small to medium-sized wind farms by generating a turbulent wind flow which enables enhanced wake recovery. In the former, power gains for wind farms with optimal coordinated control in the order of 8-21% compared with a greedy controlled case are found. In the later, these results are explored in more depth, where the simpler approach of varying the induction sinusoidally is taken. A best parameter for the sinusoidal variations is found, corresponding to an amplitude  $A^*$  of 1.5 and a non-dimensional frequency  $St^*$ <sup>3</sup> of 0.25 , proving to be robust to both wind-turbine spacing and turbulence intensity.

Wind tunnels experiments with DIC detailed in [26] prove the potential of such control strategy in real-world experiments, showing a potential increase in power production up to 4% in a 3 scaled wind turbine setup.

---

<sup>2</sup>A thorough explanation of SOWFA is provided in Chapter 4.

<sup>3</sup>The adimensionalized frequency is quantified by the Strouhal number, defined as  $St = fD/U_\infty$  for a given frequency  $f$ , rotor diameter  $D$  and inflow velocity  $U_\infty$ .

#### 2.1.2.4 Wake Redirection Control

The WRC strategy attempts to **change the direction of the wake so that the overlap of the wake with downstream rotors can be avoided**. It can be achieved by yaw misalignment and Individual Pitch Control (IPC), although the benefits of the later are in terms of enhancing *wake mixing*<sup>4</sup>. Another possibility for WRC is by tilting the rotor, meaning to lean the rotor forward or backwards, although it is not a controllable feature in present-day wind turbines. A simplified physical explanation for each WRC strategy is illustrated in Figures 2.9 and 2.10.

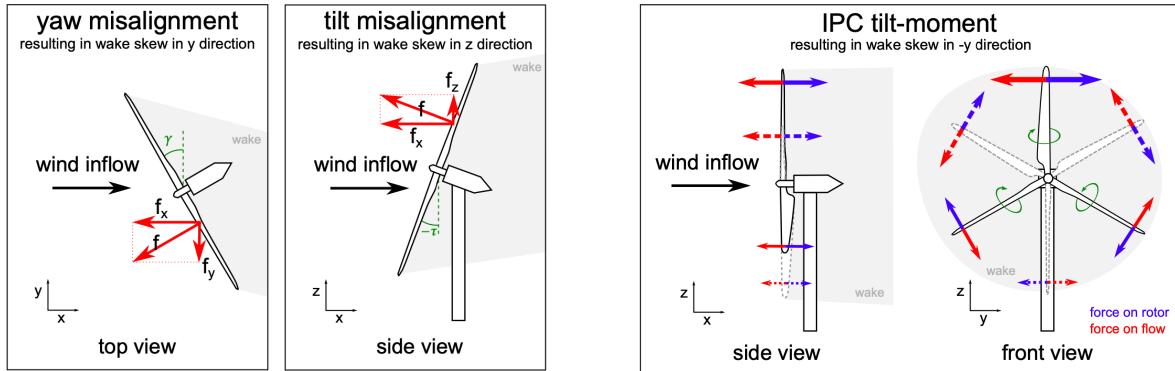


Figure 2.9: Rotor misalignment and thrust force decomposition applies a force to the flow which causes wake redirection, taken from [28].

Figure 2.10: IPC generates an asymmetric rotor torque which can yield a force perpendicular to the flow direction, taken from [28].

On the left, in figure 2.9, the thrust force  $f$  that the rotor exerts on the flow is shown along the axis of the rotor. When the wind inflow is at an angle to this direction due to misalignment, the thrust can be divided into its components  $f_x$  and  $f_y$ . The component  $f_x$  is parallel to the flow and slows the wind, **while  $f_y$  is perpendicular and applies the force that causes wake redirection**. Figure 2.11 depicts a schematic illustration of the wake being deflected behind the first turbine due to the component  $f_y$ .

On the right, IPC creates an uneven distribution of thrust forces on the rotor blades over the course of a rotation (side view) and uneven blade torques over the course of a rotor rotation as well (front view). The former will not cause significant redirection of the flow behind the turbine, as the thrust reaction forces on the flow are directed parallel to the inflow direction. The latter, however, **will cause the flow to be redirected and the wake structure to be skewed, as the turbine applies a net force on the flow perpendicular to the thrust direction**. These unbalances cause moments at the rotor centre, the tilt  $M_{tilt}$  and yaw  $M_{yaw}$  moments.

The dynamical concepts used in the context of AIC have also been leveraged in IPC. Such approach is labelled the Dynamic Individual Pitch Control (DIPC), where the goal is to dynamically vary the tilt and yaw moments such that the wake of the turbine is manipulated in vertical and/or horizontal direction, over time. The completely novel concept published in 2019 is described in [25]. The resulting wake propagates through space in a helical fashion, as shown in Figure 2.12 and the strategy is referred to as the *helix strategy*. The article shows it can be effective as a wind farm control strategy, improving wind energy

<sup>4</sup>Upstream turbines are dynamically uprated and downrated on short timescales to induce additional wake recovering, minimizing wake losses further downstream [25, 69].

capture and decreasing power and thrust variations due to enhanced wake mixing.

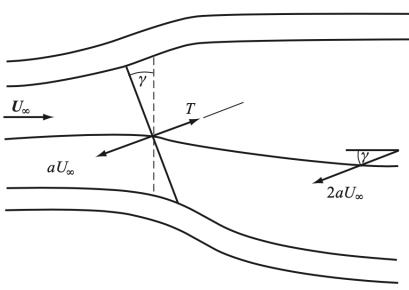


Figure 2.11: Deflected wake of a yawed turbine and induced velocities, originally in [34].

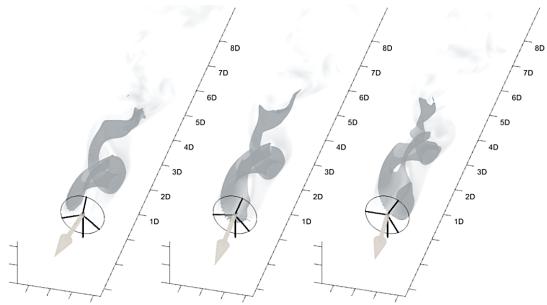


Figure 2.12: The helix approach, using DIPC, evolving through time. Adapted from [25].

**The focus of this work is on WRC using rotor misalignment by yawing the turbine.** This strategy has been proved to yield good results in simulations, wind tunnel experiments and, more recently, field tests in offshore wind farms. Moreover, wake behaviour as a result of yaw actuation is an actively research topic in order to predict wake flows and provide valuable physical insights on the behaviour of turbine wakes. For a thorough discussion of the physical phenomena of the wake characteristic of a yawed wind turbine, the reader is referred to [7].

High fidelity simulations in SOWFA [23] have showed significant redirection effects coupled with reductions in loading. Wind tunnel tests have reproduced promising results as well. In [16], 2 scaled wind turbines with a rotor diameter of 1.9 meters are placed apart, with the second one located at a distance of 4 rotor diameters downstream. It is then shown that **power increases of up to 21% are achievable for a partial wake condition with a lateral offset of  $\pm 0.45$  rotor diameters.**

Field tests in commercial wind farms have also been performed to better assess the benefits of wake steering. In [32], wake steering is demonstrated at an onshore six-turbine array, showing significant gains in power production for particular situations. The **control scheme developed increased the power production for wind speeds near the site annual average between 7% and 13%** and decreased variability by up to 72% for selected wind directions at night. Nevertheless, the net gain over annual production is reported to be insignificant when compared to baseline operation. Additional field campaign results exist, with an approximate 14% increase in energy being measured on the downstream turbine for two closely spaced turbines in [22]. The combined upstream-downstream turbine pair in the later article registered a 4% increase in energy production. Another example is [24], whose results indicate the ability of a wake-steering controller to successfully increase power capture by amounts similar to those predicted from engineering models.

Furthermore, wake steering is further demonstrated in a complex onshore wind farm in Italy [21], containing different turbine types and more complicated wake interaction. **The measurements show increases of up to 35% for two-turbine interactions and up to 16% for three turbine interactions. However, losses in power production are seen for various regions of wind directions.**

In [52], dynamic strategies for yaw are also analysed and it is found that dynamic yawing provides significant benefits over static yaw control in turbulent flow environments, whereas that is not the case for

an uniform inflow profile.

Wind farm control is an active and growing field of research. Consensus of research findings, best practices and identification of open research questions are needed, as mentioned in [69], a recent expert elicitation on the field. The later points towards reaching common standard definition of concepts, metrics, and tools, as recommended by survey results and Topical Expert Meeting (TEM) discussions.

## 2.2 Modelling Wind Turbines and Wind Farms

### 2.2.1 Wind Farm Models

The task of modelling, broadly speaking, focuses on describing the relationships between variables of a given system. **The system represents an object of interest, in which variables of different kinds interact and produce observed signals. The system of interest in this thesis is a small wind farm**, more specifically, a set of two turbines which interact due to the wake. The relevant observed signals are referred as *outputs* and the external signals manipulated by the observer which serve as a stimulus to the system are called *inputs*. The role of the model is then to incorporate these relations, so the user may perform simulations and predictions, in an attempt to better understand the behaviour of the system of interest. **The study of wind farm control algorithms requires an accurate description of the wake behaviour and its interaction with wind turbines.** This information is contained in what is referred to as the *wind farm model*.

#### 2.2.1.1 Turbine and flow models

Although wind farm models may differ, three main components can always be distinguished, as described in [28]:

- **A model describing the flow characteristics in the wind field around the turbines:** flow models generally fall in 2 categories - parametric and Computational Fluid Dynamics (CFD) models. Parametric models are simplified analytical formulations, where mathematical expressions describing key properties of the fluid - velocity, turbulence - are used. The expressions are a function of a limited number of input parameters. CFD models rely on solving the equations that describe the motion of a fluid in two or three dimensions. CFD models are more sophisticated, but the resulting computation time can be in the order of days or weeks.
- **A model of the forces of the wind turbine on the flow and the reactive forces of the flow on the wind turbine:** two models usually used are the Actuator Disk Model (ADM) - the rotor is modelled as a disk of distributed forces exerted on the flow - and the Actuator Line Model (ALM) - a more detailed representation of the rotor where each blade is represented individually in the flow as a distribution of forces along a rotating line. Using the indicated models has significant advantages in terms of computational cost, as using an exact representation of the blade bodies becomes too costly for simulating wind plants.

- **A model describing the effect of the flow forces on the performance of the wind turbine system:** estimations on power capture by wind turbines are contemplated in the ADM and ALM [10].

A more elaborate turbine model is the Fatigue Aerodynamics Structures and Turbulence (FAST), developed by the National Renewable Energy Laboratory (NREL). It contains the ALM and calculates the loads, power and rotor speed of each turbine.

## 2.2.2 Governing Dynamics

The motion of a fluid - and thus the wake - is governed by three basic differential equations: the continuity equation, the momentum equation and the energy equation (which is omitted for simplicity). The derivation can be seen in [53]:

$$\text{Continuity: } \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{V}) = 0 \quad (2.7)$$

where  $\cdot \nabla$  is the vector divergence operator<sup>5</sup>,  $\rho$  the fluid's density and  $\mathbf{V}$  the velocity vector. The equation in (2.7) expresses the conservation of mass (within a certain volume).

$$\text{Momentum: } \rho \frac{d\mathbf{V}}{dt} = \rho \mathbf{g} - \nabla p + \nabla \cdot \boldsymbol{\tau}_{ij} \quad (2.8)$$

where  $\nabla$  is the vector gradient operator,  $\mathbf{g}$  is the gravitational acceleration vector,  $p$  is the hydrostatic pressure and  $\boldsymbol{\tau}_{ij}$  the viscous stress tensor. The equation in (2.8) expresses the conservation of momentum. For an incompressible<sup>6</sup> newtonian<sup>7</sup> fluid it simplifies to the famous Navier Stokes (NS) equations, which, written in full, illustrate the inherent mathematical difficulties:

$$\begin{aligned} \rho g_x - \frac{\partial p}{\partial x} + \mu \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right) &= \rho \frac{du}{dt} \\ \rho g_y - \frac{\partial p}{\partial y} + \mu \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2} \right) &= \rho \frac{dv}{dt} \\ \rho g_z - \frac{\partial p}{\partial z} + \mu \left( \frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2} \right) &= \rho \frac{dw}{dt} \end{aligned} \quad (2.9)$$

where  $\mu$  is the viscosity coefficient and  $(u, v, w)$  the velocity components in the cartesian coordinate system  $(x, y, z)$ . The triplet of equations along with the incompressible form of the continuity equation (2.7) form four equations in the four unknowns  $p, u, v$  and  $w$ . The NS equations are second-order Nonlinear Partial Differential Equations (NPDE) and have only a limited number of known analytical solutions. **The inherent mathematical complexity of these equations are the driving force behind the using of other methodologies, such as data driven and equation free techniques, for the purposes of modelling wind farms.**

The flow may exhibit smooth and steady (laminar) or fluctuating and agitated behaviour (turbulent). Turbulence is characterized by the rapid and random fluctuations in the flow. Every pressure and velocity

---

<sup>5</sup>The divergence and gradient operators are detailed in Annex B.

<sup>6</sup>An incompressible fluid is defined as one whose density does not change as a result of pressure changes.

<sup>7</sup>A newtonian fluid is defined as one with constant viscosity due to the linear relation between applied shear and resulting strain rate.

terms in equations (2.7) and (2.8) is a rapidly varying function of time and space, leading to the unsteady NS equations. **The dynamical behaviour of a wind turbine wake is governed by the 3D unsteady NS equations.**

The above mentioned equations can be rewritten in terms of time-averaged turbulent variables, leading to the famous Reynold Averaged Navier Stokes (RANS). The velocity components are separated into their mean and fluctuating parts:  $u = \bar{u} + u'$ ,  $v = \bar{v} + v'$  and  $w = \bar{w} + w'$ . For  $x$ , the most relevant direction, the equation takes the form (for a thorough derivation, please refer to [53]):

$$\rho \frac{d\bar{u}}{dt} = -\frac{\partial \bar{p}}{\partial x} + \rho g_x + \frac{\partial}{\partial x} \left( \mu \frac{\partial \bar{u}}{\partial x} - \rho \bar{u}'^2 \right) + \frac{\partial}{\partial y} \left( \mu \frac{\partial \bar{u}}{\partial y} - \rho \bar{u}' \bar{v}' \right) + \frac{\partial}{\partial z} \left( \mu \frac{\partial \bar{u}}{\partial z} - \rho \bar{u}'^2 \right) \quad (2.10)$$

where the terms  $-\rho \bar{u}'^2$ ,  $-\rho \bar{u}' \bar{v}'$  and  $-\rho \bar{u}' \bar{w}'$  are called *turbulent stresses*. The several terms mentioned are of interest to design the model in Chapter 4.

### 2.2.3 Reduced Order Models

A typical high fidelity wind farm simulation might contain  $10^6$  or more states - points where the equations are numerically solved and the flow field information retrieved. However, building models with such complexity might prove computationally unfeasible. **An alternate solution is to perform model order reduction techniques, thus diminishing the model's complexity whilst still benefiting from the accuracy of high fidelity simulations.** Furthermore, the flow fields obtained with the lower-order model with that of a high fidelity flow solver can be compared.

Examples of such techniques are Proper Orthogonal Decomposition (POD) and Dynamic Mode Decomposition (DMD). These provide a **possible route to real-time implementation of optimal controllers**, as mentioned in [50], considering that simulation-based optimal controllers using the complete flow field information have high computational costs and are hence not suitable for designing practical cooperative wind-farm controllers. In addition, Reduced Order Model (ROM) - the final model resulting from applying model order reduction techniques - can be **pure data driven, meaning no prior knowledge of the governing equations is necessary, which is of significant advantage given the complexity of the unsteady NS equations**. Nevertheless, it should be noted that these techniques rely on specific operating conditions, meaning they provide linear models for a specific operating point and are only valid within small deviations from this point [10].

#### 2.2.3.1 Proper Orthogonal Decomposition

POD is used to **analyze experimental data with a view of extracting dominant features and trends** - in particular, the so called *coherent structures*. A coherent structure is a type of repetitive pattern seen in a turbulent fluid flow, maintaining the same size/shape through time, in spite of the whole flow being unsteady and not repeating itself in detail. An example of these structures are the wakes and pair of counter-rotating vortices formed on the surface of flowing water behind obstacles, long ago sketched by Leonardo Da Vinci in his notebooks [31]. Therefore, POD, in the context of turbulence and other complex spatio-temporal fields, will typically extract patterns in space and time.

At the heart of the POD is the Singular Value Decomposition (SVD) of the experimental data matrix  $\mathbf{X}$  (further elaborated in the next chapter). The SVD is a matrix factorization<sup>8</sup>, and applying it to  $\mathbf{X}$  yields:

$$\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^* \quad (2.11)$$

POD separates the space and time correlations into the matrices  $\mathbf{U}$  and  $\mathbf{V}$ , respectively. The matrix  $\mathbf{U}$  only contains the spacial correlations in the data, whereas all temporal information is found in  $\mathbf{V}$ . The columns of  $\mathbf{U}$  are called POD modes, which provide an orthogonal basis of spatially correlated modes. Its properties suggest that it is the preferred basis to use in various applications [31]. The singular values  $\Sigma$  represent the relative energy in each mode. In this way, this method determines the most energetic structures.

Nevertheless, drawbacks from employing this technique arise and are pointed out in [63]: the energy may not in all circumstances be the correct measure to rank the flow structures. This shortcoming has been widely recognized and serves as an explanation for the existence of dynamically highly relevant but zero energy modes. DMD emerges as a **decomposition method based on data which is able to yield fluid structures that accurately describe the motion of the flow**, overcoming the shortcoming of POD. This technique produces structures of a linear tangent approximation to the underlying flow, thus describing fluid elements that express the dominant dynamic behaviour captured in the data sequence [63].

This chapter addresses the aerodynamic interaction between wind turbines when grouped in wind farms which lead to collective suboptimal power production, as well as the wind farm control strategies developed to tackle this issue. The control mechanisms used for individual turbine control and how these can be leveraged to developed with farm control strategies are reviewed. Furthermore, the equations governing the dynamics of a wind farm are introduced, as well as the mathematical tools available to model such high dimensional non linear system in an equation free approach.

In the next chapter, the core method used to model the wind farm system in this work, DMD, is thoroughly discussed.

---

<sup>8</sup>For more details on the SVD , please refer to Annex B.



# CHAPTER 3

## DYNAMIC MODE DECOMPOSITION

---

### 3.1 Introduction to Dynamic Mode Decomposition

DMD is included in the group of techniques commonly referred to as data-driven techniques. Such modelling tools **approximate the nonlinear dynamics of a system in an equation free-way**. Equation free-strategies seek to relax the assumption that the underlying non linear governing evolution equations are known, by exploiting sampling and data collection in the complex system. Therefore, DMD reverses the process and **makes use of the abundance of data collected to reverse-engineer the governing equations of the system from data**, as [43] elegantly summarizes.

Modern systems of interest, such as epidemiological systems or financial markets, may be characterized as being high dimensional, non linear systems which exhibit rich multi-scale phenomena in space and time. DMD role is to **extract meaningful spatio-temporal patterns that dominate dynamic activity in such systems**. The unprecedented availability of high fidelity measurements from numerical simulations and experimental data, the abundance of computational resources and capacity for digital storage and the *big data regime*<sup>1</sup> have made for a fertile ground for the rapid development of such algorithms. Moreover, they provide, more than ever, a tremendous incentive to make use such of algorithms.

The method originated in the fluid dynamics community as a means to decompose complex flows into a simple representation and it was first proposed by [63] and later published in [62]. In the latter, DMD is validated and applied for three different flow cases - the first data set is generated numerically in a simulation and the last two based on data extracted from experimental flow measurements. As a result, Peter Schmid, author of the articles, **proves DMD as a robust and reliable algorithm to extract spatio-temporal coherent structures from the data**.

DMD has been applied to a wide variety of flow geometries and it has proven its value as a new quantitative flow analysis tool when compared to other methods. It is shown in [64] whereas POD concentrates on the more energetic structures of the flow, DMD isolates the less energetic but more unstable axial vorticity patterns close to the outer wall of the cylinder, which may be dynamically more relevant.

---

<sup>1</sup>The big data regime refers to the gargantuan bulk of data which cannot be dealt with by traditional data-handling techniques, weather in terms of volume, velocity or variety of data [61].

Since then, different variations of DMD have surged. In [30] DMD is generalized to a larger class of datasets, including nonsequential time series. Furthermore, DMD has been leveraged in other domains of science. In the context of epidemiology [56], DMD allows for epidemiological interpretation of large-scale dynamic patterns such as how a pattern of disease infection spreads across a spatial domain. The insights uncovered by DMD are of the highest interest for planning annual resource allocation of vaccines and improving other decision making processes. Within the context of neuroscience [14], DMD is used to extract coherent patterns in large scale neural recordings to decipher structure's oscillatory behaviour generated in the brain during sleep, thus helping to tackle a major challenge in the field.

DMD has numerous uses and the above mentioned examples lay in one of the three primary tasks of DMD - Diagnostics. The three primary tasks are, as [43] elaborates:

1. **Diagnostics:** extracting key spatio-temporal features of high dimensional systems, allowing for physically interpretable results in terms of spatial structures and their associated temporal responses. The user can then increase his understanding of the system being studied.
2. **State estimation and future-state prediction:** anticipating the behaviour of the system where no measurements were made. This more sophisticated use of the DMD algorithm can be achieved by using the dominant spatio-temporal identified structures to construct dynamical models of the underlying processes observed.
3. **Control:** enabling control strategies is the ultimate and most challenging goal of the DMD algorithm. The intrinsic difficulty is understandable as the final product of DMD - a linear dynamical model - is used to predict the future of a non linear dynamical system. It is reasonably expectable that perhaps only a limited short time window in the future where the two models will actually agree exists.

**The work in this thesis addresses the three tasks:** it (1) capitalizes DMD to understand the dynamics of the system being studied, (2) builds a model with the most relevant dynamic spatio-temporal structures and (3) performs predictive control with the chosen model.

The applications of DMD for the purposes of wind farm modelling are limited. One of the first instances in the literature, dating from 2015, is [33], where a DMD model is embedded in a Kalman filter<sup>2</sup> and shows that the entire wake flow can be predicted with a global accuracy of about 4%. The application, however, is only for a single turbine. A two turbine array setting appears then in [4], where the findings indicate that a ROM can be obtained and used as a predictive model for a similar dataset, where forced input is now considered.

Furthermore, in [1], a Linear Quadratic Gaussian (LQG) controller<sup>3</sup> based on a ROM is designed in order to minimize velocity fluctuations at a specified downstream location and it is shown that the controller is able to reduce significantly fluctuations by varying the thrust force.

---

<sup>2</sup>Estimator that reconstructs the state of a dynamical system from measurements of a time series of the sensor outputs and actuation inputs. For more information please see glossary.

<sup>3</sup>A LQG controller combines a Kalman filter and a Linear Quadratic Regulator (LQR) which minimizes a quadratic cost function.

## 3.2 Dynamic Mode Decomposition Architecture in Fluid Mechanics

In the DMD framework, data is collected from a dynamical system

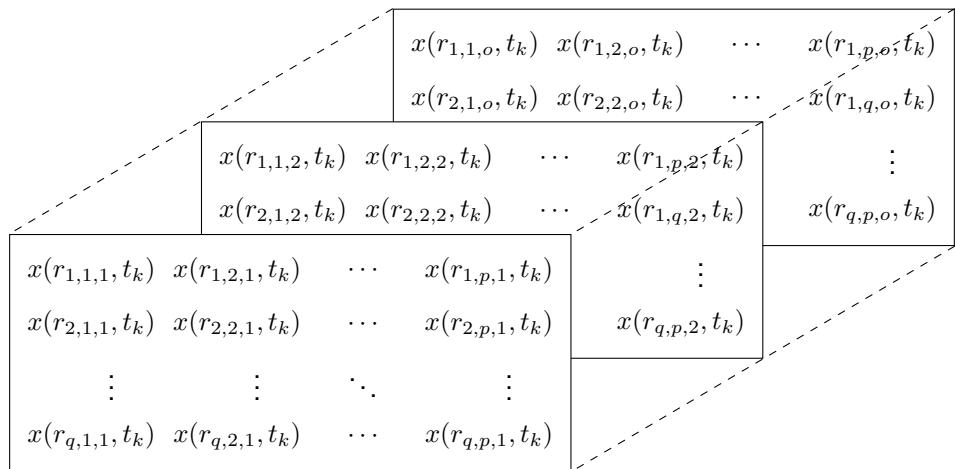
$$\frac{dx}{dt} = \mathbf{f}(x; t; \mu) \quad (3.1)$$

where  $x(t) \in \mathbb{R}^n$  is a vector representing the state<sup>4</sup> of the dynamical system at time  $t$ ,  $\mu$  contains parameters of the system and  $\mathbf{f}(\cdot)$  represents the dynamics. Resorting to the equation free perspective, data measurements of the system alone are used to approximate the dynamics. The approximated linear dynamical system can be constructed in discrete-time, sampling the analogous continuous time system every  $\Delta t$ :

$$x_{k+1} = \mathbf{A}x_k \quad (3.2)$$

The relation in (3.2) puts in evidence the inherent assumption that a linear operator  $\mathbf{A} \in \mathbb{R}^{n \times n}$  is responsible for the evolution of the system at each time instant. This operator  $\mathbf{A}$  is found such that it optimally fits the measured trajectory  $x_k$  for  $k = 1, 2, \dots, m$ . For this purpose, the *snapshot methodology* is used.

When analysing a time series of data containing the velocity field at a grid of spatial locations - **as is the case for a wind farm** - it is beneficial to take the three-dimensional vector field data at time  $k$  and flatten into a single column vector. Let the following example adapted from [43] be considered. For the three dimension vector field,  $x$  denotes the flow variable of interest,  $r$  the spatial coordinate and the index  $k$  the  $k$ th time step. For a generalized grid with  $q, p$  and  $o$  points in the three Cartesian directions:



The flattened resulting column vector  $x_k \in \mathbb{R}^n$  containing the exact same information is referred to as

---

<sup>4</sup>The states of a system are the variables which determine the evolution of the system. This concept, as well as the type of model used to represent the dynamical system, are thoroughly discussed in Chapter 4.

a *snapshot* of data.

$$\mathbf{x}_k = \begin{bmatrix} x(r_{1,1,1}, t_k) \\ x(r_{1,2,1}, t_k) \\ \vdots \\ x(r_{2,1,1}, t_k) \\ x(r_{q,p,1}, t_k) \\ \vdots \\ x(r_{q,p,o}, t_k) \end{bmatrix} \quad (3.3)$$

The snapshots at each time instant are then combined to form the *data matrix*  $\mathbf{X}_{full}$ :

$$\mathbf{X}_{full} = \begin{bmatrix} | & | & & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_m \\ | & | & & | \end{bmatrix} \quad (3.4)$$

The number of rows  $n$  of the data matrix is equal to the number of spatial points saved per time snapshot. The number of columns  $m$  of the data matrix is equal to the number of snapshots taken, which in turn corresponds to the number of instants the flow field is sampled. The data matrix is then divided into two matrices in order to approximate the operator  $\mathbf{A}$  in equation (3.2), where  $\mathbf{X}'$  is the time shifted snapshot matrix of  $\mathbf{X}$ :

$$\mathbf{X} = \begin{bmatrix} | & | & & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_{m-1} \\ | & | & & | \end{bmatrix} \quad \mathbf{X}' = \begin{bmatrix} | & | & & | \\ \mathbf{x}_2 & \mathbf{x}_3 & \cdots & \mathbf{x}_m \\ | & | & & | \end{bmatrix} \quad (3.5)$$

Equation (3.2) can now be rewritten in terms of the data snapshots:

$$\mathbf{X}' = \mathbf{AX} \quad (3.6)$$

The best fit  $\mathbf{A}$  matrix is given by

$$\mathbf{A} = \mathbf{X}'\mathbf{X}^\dagger \quad (3.7)$$

where  $\dagger$  is the Moore-Penrose pseudoinverse. This solution minimizes the error  $\|\mathbf{X}' - \mathbf{AX}\|_F$ , where  $\|\cdot\|_F$  is the Frobenius norm [29], described in Annex B. Hence, DMD can be thought of as a least-square fit<sup>5</sup>, or regression, of a best fit linear dynamical system  $\mathbf{x}_{k+1} = \mathbf{Ax}_k$  for the data sampled. Matrix  $\mathbf{A}$  is constructed so that  $\|\mathbf{x}_{k+1} - \mathbf{Ax}_k\|_2$  is minimized across all snapshots, where  $\|\cdot\|_2$  is the  $l_2$  norm of a vector, described in Annex B.

In practice, due to the large state dimension  $n$ , matrix  $\mathbf{A}$  becomes intractable to analyse directly, so DMD attempts to fit the snapshots in time using a low rank matrix which captures the dynamics of the

---

<sup>5</sup>A least-squares regression is a regression technique where a best-fit line or vector is found by minimizing the sum of squares of the error between the model and the data.

data set. A low-order representation of the states  $x_k$  can be written as:

$$\tilde{x}_k = \mathbf{Q}^* x_k \quad (3.8)$$

where  $\mathbf{Q}$  is a generic orthonormal basis that forms the projection subspace and  $*$  designates the conjugate transpose. A thorough explanation of the several steps that this task encompasses are provided in the following subsections.

### 3.3 Dynamic Mode Decomposition Algorithms

#### 3.3.1 Dynamic Mode Decomposition

In order to find the best fit linear operator  $\mathbf{A}$  and its corresponding low rank matrix, the standard DMD algorithm described in [30, 43, 58], proceeds as follows :

##### 1. Singular Value Decomposition (SVD) of $\mathbf{X}$ :

$$\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^* = \begin{bmatrix} \tilde{\mathbf{U}} & \tilde{\mathbf{U}}_{\text{rem}} \end{bmatrix} \begin{bmatrix} \tilde{\Sigma} & 0 \\ 0 & \tilde{\Sigma}_{\text{rem}} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{V}}^* \\ \tilde{\mathbf{V}}_{\text{rem}}^* \end{bmatrix} \quad (3.9)$$

where:  $\mathbf{U} \in \mathbb{R}^{n \times n}$ ,  $\Sigma \in \mathbb{R}^{n \times m-1}$ ,  $\mathbf{V}^* \in \mathbb{R}^{m-1 \times m-1}$  and  $\tilde{\mathbf{U}} \in \mathbb{R}^{n \times r}$ ,  $\tilde{\Sigma} \in \mathbb{R}^{r \times r}$ ,  $\tilde{\mathbf{V}}^* \in \mathbb{R}^{r \times m-1}$ . The subscript  $_{\text{rem}}$  indicates the remaining  $m - 1 - r$  singular values. By appropriately choosing a truncation value  $r$  of the number of singular values to retain, it is possible to eliminate the remaining  $_{\text{rem}}$  terms and find a reduced SVD approximation of  $\mathbf{X}$ .

$$\mathbf{X} \approx \tilde{\mathbf{U}}\tilde{\Sigma}\tilde{\mathbf{V}}^* \quad (3.10)$$

For simplicity purposes, the truncated SVD approximation of  $\mathbf{X}$  will be referred simply as:

$$\mathbf{X} \approx \mathbf{U}\Sigma\mathbf{V}^* \quad (3.11)$$

##### 2. Compute matrix $\mathbf{A}$ in (3.6) by substituting the SVD of $\mathbf{X}$ (3.11) in (3.7):

The pseudoinverse can now be accomplished since  $\Sigma$  is squared. By using the known result of the pseudoinverse of the SVD:

$$\mathbf{A} \approx \mathbf{X}'(\mathbf{U}\Sigma\mathbf{V}^*)^\dagger \approx \mathbf{X}'\mathbf{V}\Sigma^{-1}\mathbf{U}^* \quad (3.12)$$

##### 3. Project matrix $\mathbf{A}$ from (3.12) onto subspace and obtain a low dimension linear model $\tilde{\mathbf{A}}$ :

Matrix  $\mathbf{A}$  in (3.12) is high dimensional of size  $n \times n$ , and an eigenvalue analysis to uncover the underlying dynamics may be computational prohibitively expensive. Therefore, a more compact and computationally efficient model can be found, assuming that  $r \ll n$ , by projecting the full state onto a linear subspace of dimension  $r$ , as mentioned in the end of Section 3.2. A convenient transformation

has already been computed via the SVD of  $\mathbf{X}$  and the projection subspace  $\mathbf{Q}$  corresponds to the left singular vectors  $\mathbf{U}$ , the POD modes.

$$\mathbf{A} \approx \tilde{\mathbf{A}} = \mathbf{U}^* \mathbf{A} \mathbf{U} = \mathbf{U}^* \mathbf{X}' \mathbf{V} \Sigma^{-1} \mathbf{U}^* \mathbf{U} \quad (3.13)$$

The columns of  $\mathbf{U}$  are orthonormal due to the properties of the SVD, which means that  $\mathbf{U}^* \mathbf{U} = \mathbf{I}$ . Taking this into account, (3.13) reduces to:

$$\tilde{\mathbf{A}} = \mathbf{U}^* \mathbf{X}' \mathbf{V} \Sigma^{-1} \quad (3.14)$$

Matrix  $\tilde{\mathbf{A}}$  defines the low-dimensional linear model of the dynamical system on POD coordinates:

$$\tilde{x}_{k+1} = \tilde{\mathbf{A}} \tilde{x}_k \quad (3.15)$$

The high dimensional states  $x_k$  can be reconstructed from the low dimension states  $\tilde{x}_k$  by  $x_k = \mathbf{U} \tilde{x}_k$ .

#### 4. Compute the eigendecomposition of $\tilde{\mathbf{A}}$ :

$$\tilde{\mathbf{A}} \mathbf{W} = \mathbf{W} \Lambda \quad (3.16)$$

where columns of  $\mathbf{W}$  are eigenvectors and  $\Lambda$  is a diagonal matrix containing the corresponding eigenvalues  $\lambda_k$ . These can be investigated for fundamental properties of the underlying system such as growth modes and resonance frequencies.

#### 5. Reconstruct eigendecomposition of $\mathbf{A}$ :

To approximate the eigenvalues and eigenvectors of  $\mathbf{A}$  without its explicit computation, an approximation is calculated based on the so called *DMD modes*. The eigenvalues of  $\mathbf{A}$  are given by  $\Lambda$  and the eigenvectors are approximated by the dynamic mode, which correspond to the columns of  $\Phi$  in (3.17), as originally proposed in [62] and figuring in the standard DMD algorithm in [30]:

$$\Phi = \mathbf{U} \mathbf{W} \quad (3.17)$$

These modes in (3.17) are referred to as *projected DMD modes*. Another formulation is used to compute the DMD modes:

$$\Phi = \mathbf{X}' \mathbf{V} \Sigma^{-1} \mathbf{W} \quad (3.18)$$

The modes in (3.18) are often called *exact DMD modes*, as it was proven in [30] that these are the exact eigenvectors of the matrix  $\mathbf{A}$ . The projected DMD modes will tend to converge if  $\mathbf{X}$  and  $\mathbf{X}'$  have the same column space [43].

In the context of fluid mechanics, **the modes are spatial fields that often identify coherent structures in the flow**. The correspondent eigenvalues define growth or decay rates and oscillation frequencies for each mode. Taken together, the DMD modes and eigenvalues describe the dynamics observed in the

time series in terms of oscillatory components.

### 3.3.2 Dynamic Mode Decomposition with Control

Dynamic Mode Decomposition with control (DMDc) modifies the basic assumption of DMD, in the sense that the linear dynamical system connecting the future state  $x_{k+1}$  now relies not only on the current state  $x_k$ , as expressed in (3.2) but also on the current control  $u_k$ , given by the following relation:

$$x_{k+1} = \mathbf{A}x_k + \mathbf{B}u_k \quad (3.19)$$

where  $x \in \mathbb{R}^n$ ,  $u \in \mathbb{R}^q$ ,  $\mathbf{A} \in \mathbb{R}^{n \times n}$  and  $\mathbf{B} \in \mathbb{R}^{n \times q}$ . DMDc allows to understand the characteristics of complex systems by helping discovering the underlying dynamics without the confounding effect of external control. In addition, the method also quantifies the effect of control inputs on the state of the system [58].

The DMDc algorithm operates in a similar fashion as the standard DMD. Besides collecting the snapshot matrices as in (3.5), the sequence of control input snapshots is assembled:

$$\Upsilon = \begin{bmatrix} | & | & & | \\ u_1 & u_2 & \dots & u_{m-1} \\ | & | & & | \end{bmatrix} \quad (3.20)$$

The relation (3.19) can be rewritten in terms of the snapshot matrices:

$$\mathbf{X}' \approx \mathbf{A}\mathbf{X} + \mathbf{B}\Upsilon \quad (3.21)$$

Assuming both operators  $\mathbf{A}$  and  $\mathbf{B}$  to be unknown - meaning there exists no knowledge about how the control affects the system, (3.21) is manipulated giving the following representation:

$$\mathbf{X}' \approx [\mathbf{A} \quad \mathbf{B}] \begin{bmatrix} \mathbf{X} \\ \Upsilon \end{bmatrix} = \mathbf{G}\Omega \quad (3.22)$$

where  $\Omega$  contains both the state and control snapshot information. Similarly to the standard DMD, let the DMDc be presented in an algorithmic fashion, where the goal is to seek the low order best fit solution of operator  $\mathbf{G}$ , as described in [58]:

1. **Compute the SVD of the input space  $\Omega$ :**

The SVD of  $\Omega$  is taken with a certain truncation value  $p$ :

$$\Omega \approx \tilde{\mathbf{U}}\tilde{\Sigma}\tilde{\mathbf{V}}^* \quad (3.23)$$

An approximation for operator  $\mathbf{G}$  is now found by taking the pseudoinverse of the SVD of  $\Omega$ :  $\mathbf{G} \approx \mathbf{X}'\tilde{\mathbf{V}}\tilde{\Sigma}^{-1}\tilde{\mathbf{U}}^*$ , where  $\mathbf{G} \in \mathbb{R}^{n \times (n+q)}$ . By breaking the linear operator  $\tilde{\mathbf{U}}$  in two separate components,

an explicit approximation of the operators  $\mathbf{A}$  and  $\mathbf{B}$  is found:

$$[\mathbf{A}, \mathbf{B}] \approx [\mathbf{X}' \tilde{\mathbf{V}} \tilde{\Sigma}^{-1} \tilde{\mathbf{U}}_1^*, \mathbf{X}' \tilde{\mathbf{V}} \tilde{\Sigma}^{-1} \tilde{\mathbf{U}}_2^*] \quad (3.24)$$

where  $\tilde{\mathbf{U}}_1 \in \mathbb{R}^{n \times p}$ ,  $\tilde{\mathbf{U}}_2 \in \mathbb{R}^{q \times p}$ ,  $\tilde{\mathbf{U}} = \begin{bmatrix} \tilde{\mathbf{U}}_1^* & \tilde{\mathbf{U}}_2^* \end{bmatrix}^T$  and  $p$  is the truncation value of the SVD in (3.23).

## 2. Compute the SVD of the output space $\mathbf{X}'$ :

The SVD of  $\mathbf{X}'$  is needed to find a projection subspace matrix  $\mathbf{Q}$  to compute the low order model. DMD makes use of the truncated left singular values  $\mathbf{U}$  of  $\mathbf{X}$  for such purpose. However, DMDc cannot make use of the left singular vectors  $\tilde{\mathbf{U}}$  of  $\Omega$  as it is defined for the input space, which now includes both the state measurements and the exogenous inputs [43, 58]. Consequently, the data matrix of the output space  $\mathbf{X}'$  is used:  $\mathbf{X}' \approx \hat{\mathbf{U}} \hat{\Sigma} \hat{\mathbf{V}}^*$  where  $r$  is the truncation value,  $\hat{\mathbf{U}} \in \mathbb{C}^{n \times r}$ ,  $\hat{\Sigma} \in \mathbb{C}^{r \times r}$ ,  $\hat{\mathbf{V}}^* \in \mathbb{C}^{r \times m-1}$ . For a tractable low order model to be found, usually  $r < p$ .

## 3. Compute the approximation of the low order operator $\mathbf{G}$ :

The reduced order approximation of operators  $\mathbf{A}$  and  $\mathbf{B}$  is then found to be:

$$\begin{aligned} \tilde{\mathbf{A}} &= \hat{\mathbf{U}}^* \mathbf{A} \hat{\mathbf{U}} = \hat{\mathbf{U}}^* \mathbf{X}' \tilde{\mathbf{V}} \tilde{\Sigma}^{-1} \tilde{\mathbf{U}}_1^* \hat{\mathbf{U}} \\ \tilde{\mathbf{B}} &= \hat{\mathbf{U}}^* \mathbf{B} = \hat{\mathbf{U}}^* \mathbf{X}' \tilde{\mathbf{V}} \tilde{\Sigma}^{-1} \tilde{\mathbf{U}}_2^* \end{aligned} \quad (3.25)$$

where  $\tilde{\mathbf{A}} \in \mathbb{R}^{r \times r}$  and  $\tilde{\mathbf{B}} \in \mathbb{R}^{r \times q}$ . The reduced order model can now be written as:

$$\tilde{\mathbf{x}}_{k+1} = \tilde{\mathbf{A}} \tilde{\mathbf{x}}_k + \tilde{\mathbf{B}} \mathbf{u}_k \quad (3.26)$$

## 4. Perform eigenvalue decomposition of $\tilde{\mathbf{A}}$ :

$$\tilde{\mathbf{A}} \mathbf{W} = \mathbf{W} \Lambda \quad (3.27)$$

where columns of  $\mathbf{W}$  are eigenvectors and  $\Lambda$  is a diagonal matrix containing the corresponding eigenvalues  $\lambda_k$ .

## 5. Reconstruct the dynamic modes of operator $\mathbf{A}$ :

The transformation from eigenvectors to dynamic modes of  $\mathbf{A}$  is slightly modified and is given by:

$$\Phi = \mathbf{X}' \tilde{\mathbf{V}} \tilde{\Sigma}^{-1} \tilde{\mathbf{U}}_1^* \hat{\mathbf{U}} \mathbf{W} \quad (3.28)$$

The theoretical mathematical framework of DMDc proves its value in terms of producing accurate reduced-order models from complex systems with exogenous forcing. Nevertheless, it is developed for a specific regime, where the number of state measurements is much larger than the rank of the underlying system [43]. For many engineering applications, the number of measurements is typically smaller than the number of states to describe the input-output models. Such challenge is addressed in the next variation of the DMD algorithm.

Despite the mentioned feature incapacitating DMDc of being exploited in certain engineering applications, it lays the grounds for the development of better suited algorithms. Furthermore, its applicability is wide. In the context of epidemiology, for example, the extension of DMD to DMDc helps planning eradication campaigns by identifying how such inputs drive the disease propagation [56].

### 3.3.3 Input-Output Dynamic Mode Decomposition

In this section, an extension of the DMD approach to include inputs and outputs is introduced, based on the work in [2, 4], where reduced order models are build to model the wind farm flow based on a medium-fidelity and a high-fidelity simulation, respectively.

In Section 3.3.2, DMD is studied in the context of control. This approach projects the full-order model onto the output subspace, where the inputs are accounted for when fitting the data to a linear system. **Input-Output Dynamic Mode Decomposition (IODMD)**, in its turn, combines DMD with standard subspace identification, often used in the controls literature [71], to obtain an input-output characterization of the system.

In addition to the relation in (3.19), a second is added to account for the mapping between the state measurement and the outputs:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k \\ \mathbf{y}_k &= \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k \end{aligned} \quad (3.29)$$

where  $\mathbf{x}_k \in \mathbb{R}^n$ ,  $\mathbf{u}_k \in \mathbb{R}^q$  and  $\mathbf{y}_k \in \mathbb{R}^l$  are the state, input and output vectors respectively. Matrices  $\mathbf{A} \in \mathbb{R}^{n \times n}$ ,  $\mathbf{B} \in \mathbb{R}^{n \times q}$ ,  $\mathbf{C} \in \mathbb{R}^{l \times n}$  and  $\mathbf{D} \in \mathbb{R}^{l \times q}$  are the state matrices that together form the *state space representation*, a common mathematical model of a physical system used in control engineering, further explored in Section 4.

The sequence of output snapshots is now assembled, as well as the input and states snapshots - also previously gathered in the DMDc and DMD methodologies.

$$\mathbf{Y} = \begin{bmatrix} | & | & & | \\ \mathbf{y}_1 & \mathbf{y}_2 & \cdots & \mathbf{y}_{m-1} \\ | & | & & | \end{bmatrix} \quad (3.30)$$

Rewriting (3.29) in terms of the snapshot matrices:

$$\begin{aligned} \mathbf{X}' &\approx \mathbf{AX} + \mathbf{BY} \\ \mathbf{Y} &\approx \mathbf{CX} + \mathbf{DY} \end{aligned} \quad (3.31)$$

The relation in (3.31) can again be rewritten in a more compact format, which allows to find the low order best fit operators of  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$ ,  $\mathbf{D}$ :

$$\begin{bmatrix} \mathbf{X}' \\ \mathbf{Y} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \end{bmatrix} \quad (3.32)$$

The state is then projected onto the subspace defined by an orthonormal basis specified by the columns of matrix  $\mathbf{Q}$ , as evidenced in (3.8). The low order state matrices are then  $\tilde{\mathbf{A}} = \mathbf{Q}^* \mathbf{A} \mathbf{Q}$ ,  $\tilde{\mathbf{B}} = \mathbf{Q}^* \mathbf{B}$ ,  $\tilde{\mathbf{C}} = \mathbf{C} \mathbf{Q}$  and  $\tilde{\mathbf{D}} = \mathbf{D}$ . These equivalences can be rewritten in terms of the high order matrices  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$  and  $\mathbf{D}$ :  $\mathbf{A} = \mathbf{Q} \tilde{\mathbf{A}} \mathbf{Q}^*$ ,  $\mathbf{B} = \mathbf{Q} \tilde{\mathbf{B}}$ ,  $\mathbf{C} = \tilde{\mathbf{C}} \mathbf{Q}^*$  and  $\mathbf{D} = \tilde{\mathbf{D}}$ . Substituting in (3.32) and rearranging:

$$\begin{bmatrix} \mathbf{x}' \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{A}} & \tilde{\mathbf{B}} \\ \tilde{\mathbf{C}} & \tilde{\mathbf{D}} \end{bmatrix} \begin{bmatrix} \mathbf{Q}^* & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{r} \end{bmatrix} \quad (3.33)$$

A suboptimal but useful choice for the projection space is given by the POD modes of  $\mathbf{X}$  [2], as in standard DMD. By taking the SVD of  $\mathbf{X}$  given by  $\mathbf{X} \approx \mathbf{U} \Sigma \mathbf{V}^*$  with a truncation value  $r$ ,  $\mathbf{U}$  is chosen as  $\mathbf{Q}$ .

The reduced order state matrices are obtained by minimizing the error of the Frobenius norm:

$$\min_{\begin{bmatrix} \tilde{\mathbf{A}} & \tilde{\mathbf{B}} \\ \tilde{\mathbf{C}} & \tilde{\mathbf{D}} \end{bmatrix}} \left\| \begin{bmatrix} \mathbf{x}' \\ \mathbf{y} \end{bmatrix} - \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{A}} & \tilde{\mathbf{B}} \\ \tilde{\mathbf{C}} & \tilde{\mathbf{D}} \end{bmatrix} \begin{bmatrix} \mathbf{U}^* & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{r} \end{bmatrix} \right\|_F^2 \quad (3.34)$$

where  $\tilde{\mathbf{A}} \in \mathbb{R}^{r \times r}$ ,  $\tilde{\mathbf{B}} \in \mathbb{R}^{r \times q}$ ,  $\tilde{\mathbf{C}} \in \mathbb{R}^{l \times r}$ , and  $\tilde{\mathbf{D}} \in \mathbb{R}^{l \times q}$ . The reduced order state matrices are then obtained by least squares, taking into account that the reduced order representation of  $\mathbf{X}$  and  $\mathbf{x}'$  are respectively  $\mathbf{U}^* \mathbf{x}'$  and  $\Sigma \mathbf{V}$ :

$$\Theta_{\text{IODMD}} = \begin{bmatrix} \tilde{\mathbf{A}} & \tilde{\mathbf{B}} \\ \tilde{\mathbf{C}} & \tilde{\mathbf{D}} \end{bmatrix} = \begin{bmatrix} \mathbf{U}^* \mathbf{x}' \\ \mathbf{y} \end{bmatrix} \begin{bmatrix} \Sigma \mathbf{V} \end{bmatrix}^\dagger \quad (3.35)$$

This corresponds to a tractable direct N4SID subspace method for estimating the state matrices which can be applied to complex multi-variable dynamical systems directly from data [71].

Using the definition of the pseudo-inverse for a general matrix  $\mathcal{X}$  and under the assumption that  $\mathcal{X}$  has linearly independent rows,  $\mathcal{X}^\dagger$  can be written as  $\mathcal{X}^* (\mathcal{X}^* \mathcal{X})^{-1}$ . This result allows to rewrite equation (3.35) in a more explicit format, which will be of use later in this chapter in Section 3.4:

$$\Theta_{\text{IODMD}} = \begin{bmatrix} \tilde{\mathbf{A}} & \tilde{\mathbf{B}} \\ \tilde{\mathbf{C}} & \tilde{\mathbf{D}} \end{bmatrix} = \begin{bmatrix} \mathbf{U}^* \mathbf{x}' \\ \mathbf{y} \end{bmatrix} \begin{bmatrix} \Sigma \mathbf{V} \end{bmatrix}^T \begin{bmatrix} \Sigma \mathbf{V} \Sigma^T \mathbf{V}^T & \Sigma \mathbf{V} \mathbf{r}^T \\ \mathbf{r} \Sigma^T \mathbf{V}^T & \mathbf{r} \mathbf{r}^T \end{bmatrix}^{-1} \quad (3.36)$$

The reduced order model can now be written as:

$$\begin{aligned} \tilde{\mathbf{x}}_{k+1} &= \tilde{\mathbf{A}} \tilde{\mathbf{x}}_k + \tilde{\mathbf{B}} \mathbf{u}_k \\ \mathbf{y}_k &= \tilde{\mathbf{C}} \tilde{\mathbf{x}}_k + \tilde{\mathbf{D}} \mathbf{u}_k \end{aligned} \quad (3.37)$$

It is important to emphasize a point stated in [4]: the ROM arrived at (3.37) is not a *black box model*<sup>6</sup> as the state of the ROM can be used to approximate the full order state, by  $\mathbf{x}_k = \mathbf{U} \tilde{\mathbf{x}}_k$ .

### 3.3.4 Extended Input-Output Dynamic Mode Decomposition

It has been shown that a nonlinear system can be described by superposition of modes whose dynamics are governed by eigenvalues. However, at a first glance, it may seem dubious that such description of a

---

<sup>6</sup>System which can be solemnly viewed in terms of its inputs and outputs, without any knowledge of its internal working.

complex nonlinear system provides accurate information and that DMD is indeed applicable to nonlinear systems. Nonetheless, evidence exists supporting DMD applicability due to its connections to what is termed the *Koopman operator*, as shown in [60]. In this subsection, as a result of its relevance, this operator and its connections to DMD are explored along with variations it incites to the already seen frameworks.

### 3.3.4.1 Koopman Operator Theory

The Koopman operator was introduced in [40] and recently resurged in the context of modern data-driven dynamical systems. Succinctly, it is defined as an infinite-dimension linear operator that evolves fields of physical observables - scalar functions - defined on a state space - the set of all possible states - of a dynamical system.

More explicitly, let the generic dynamical system in equation (3.1) be again considered, and defined on a state space  $\mathbf{M}$ . This induces a discrete-time dynamical system given by the flow map  $\mathbf{F}_t : \mathbf{M} \rightarrow \mathbf{M}$ , represented in Figure 3.1.

$$\mathbf{x}_{k+1} = \mathbf{F}_t(\mathbf{x}_k) \quad (3.38)$$

A function defined as  $g : \mathbf{M} \rightarrow \mathbb{R}$  is called an *observable* of the system, as represented in Figure (3.1). The Koopman operator  $\mathcal{K}_t$  acts on observable functions  $g$  as:

$$\mathcal{K}_t = g \circ \mathbf{F}_t \quad (3.39)$$

where  $\circ$  is the composition operator, so that:

$$\mathcal{K}_t g(\mathbf{x}_k) = g(\mathbf{F}_t(\mathbf{x}_k)) = g(\mathbf{x}_{k+1}) \quad (3.40)$$

$$g(\mathbf{x}_{k+1}) = \mathcal{K}_t g(\mathbf{x}_k) \quad (3.41)$$

It is now evident in (3.41) that the observation of the state, or, in other words, a function of the measured state variable, is advanced to the next time step by the Koopman operator. This is true for *any* observable function and for *any* point  $\mathbf{x}_k \in \mathbf{M}$ , hence the infinite-dimension. The linearity property follows from its definition in (3.39).

The spectral decomposition<sup>7</sup> of the Koopman operator is instrumental in representing solutions to a dynamical system of interest. Considering the eigenvalue problem:

$$\mathcal{K}\varphi_k = \lambda_k \varphi_k \quad (3.42)$$

The functions  $\varphi_k(\mathbf{x})$  are called Koopman eigenfunctions and they **define a set of intrinsic measurement coordinates, on which it is possible to advance these measurements with a linear dynamical system**. The new measurement space formed by a vector of observables  $g(\mathbf{x})$  can be expressed in

---

<sup>7</sup>Spectral decomposition and eigendecomposition are two terms used almost interchangeably. Spectral theory simply refers to a broader set of theories which extend the eigenvector and eigenvalue theories to a broader theory concerning the structure of operator in various mathematical spaces.

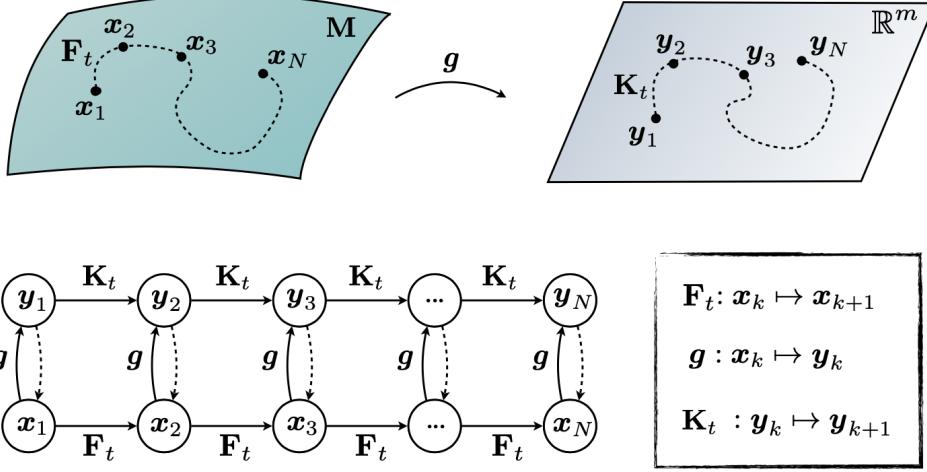


Figure 3.1: Schematic illustrating the Koopman operator for nonlinear dynamical systems, taken from [15]. On the left the direct evolution of the states  $\mathbf{x} \in \mathbf{M}$  governed by  $\mathbf{F}$  is shown. On the right, the evolution of observables  $\mathbf{y} \in \mathbb{R}^m$  governed by the Koopman Operator  $\mathcal{K}$  is shown. The connection between the states and observables is through the observable function  $g$ .

terms of the Koopman eigenfunctions:

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} g_1(\mathbf{x}) \\ g_2(\mathbf{x}) \\ \vdots \\ g_p(\mathbf{x}) \end{bmatrix} = \sum_{k=1}^{\infty} \varphi_k(\mathbf{x}) \mathbf{v}_k \quad (3.43)$$

where  $\mathbf{v}_k$  is the  $k$ th Koopman mode of the map  $\mathbf{F}$  corresponding to the observable  $g$ , associated with the  $k$ th Koopman eigenfunction, i.e., the weighting of each observable on the eigenfunction. The expression in (3.43) can be interpreted as a linear combination of the eigenfunctions  $\varphi_k$  of  $\mathcal{K}$  where  $\mathbf{v}_k$  is the coefficient in the expansion. The nonlinear system can either be evolved in the original state space or in the measurement space as in (3.43).

$$\mathcal{K}\mathbf{g}(\mathbf{x}) = \mathcal{K} \sum_{k=1}^{\infty} \varphi_k(\mathbf{x}) \mathbf{v}_k = \sum_{k=1}^{\infty} \mathcal{K}\varphi_k(\mathbf{x}) \mathbf{v}_k = \sum_{k=1}^{\infty} \lambda_k \varphi_k(\mathbf{x}) \mathbf{v}_k \quad (3.44)$$

Future solutions can be computed by multiplication with the Koopman eigenvalues,  $\lambda_k \in \mathbb{C}$ , which characterize the temporal behaviour of the corresponding Koopman mode  $\mathbf{v}_k$ .

### 3.3.4.2 Koopman Operator Theory and Dynamic Mode Decomposition

The connections of DMD to the underlying non linear dynamics of a system through Koopman operator theory were firstly established in [60], where it is shown that **DMD produces approximations to eigenvalues of the Koopman operator, and their corresponding modes**. A thorough mathematical definition of the Koopman operator and its properties as well as an overview of the rapidly growing work using both these concepts can be seen in [47].

DMD is thus a manifestation of Koopman theory for the specific case when the observable functions

are the identity or a linear transformation of the underlying state space:

$$g_{\text{DMD}}(x) = x \quad (3.45)$$

**DMD provides a tractable computational alternative, as the finite-dimensional matrices it yields are an approximation to the infinite-dimensional operator.** In other words, DMD transforms the useful but abstract mathematical conception to a readily implementable computation.

The connections with the Koopman operator have given birth to variations of DMD, that attempt to better approximate the Koopman eigenvalues and modes and hence improve the approximation of the full nonlinear dynamics. **The key idea is to apply DMD on the space of observables.** For that purpose, the **snapshot matrices in (3.5) are generalized and assembled with observables.**

### 3.3.4.3 Extended Dynamic Mode Decomposition

The Extended Dynamic Mode Decomposition (EDMD) was introduced in [72], **augmenting DMD to include a richer set of nonlinear observables** and as a consequence providing the ability of effectively analysing nonlinear systems. For high-dimensional systems where such extension might become too expensive, techniques from machine learning can be used as shown in [38], thus benefiting from the richness of the non linear observables while maintaining an acceptable computational cost.

The major challenge this approach faces concerns the choosing of the nonlinear observables, as shown in [15, 42]. Furthermore, in the latter other drawbacks of EDMD are discussed, such as the lost of interpretability (common in machine learning schemes) and high variability of results (reiterating the importance of cross validation).

An alternative to the standard EDMD using machine learning techniques is to **select a parsimonious set of observables based upon knowledge of the physics of the system.** This means that **knowledge of nonlinear terms figuring in the NPDE that govern the dynamics of complex systems is used to guide the selection of observables** (analogous to the selection of *features* in the context of machine learning guided by expert knowledge). An extensive overview of such approach is given in [42], where compelling evidence is presented: by judiciously selecting observables based on nonlinear terms figuring in the Schrödinger NPDE, the deviations between DMD's approximation and true solution dramatically decreases (in about 3 orders of magnitude). Additionally, in [45], it is also shown that when combined with adequate choice of observables, the Koopman operator is able to map the nonlinear underlying dynamics with the linear observable space, where the DMD can be implemented with good accuracy (and efficiency).

The linear representation provided by the Koopman operator and its computational implementable correspondent DMD, are useful not only for prediction but also for control, as explored in the works of [6, 41, 57]. In the last two, a structured approach is given to derive linear models suited for predictive control based on the ideas of the Koopman operator. It boils down to a nonlinear transformation of the data - termed the *lifting* - and a linear least squares problem in the lifted space.

For the purpose of this thesis, **these ideas are explored by using the Extended Input Output**

**Dynamic Mode Decomposition (EIODMD)**, which combines the benefits of using nonlinear observables with the subspace identification methods and least squares regression exploited in IODMD, allowing to derive tractable models for predictive control that embed the relevant dynamics.

The formulation of the problem is similar to IODMD, where the goal is to find the ROM that best fits the gathered data. The snapshot matrices are then lifted, (with a slight different notation to match the current literature, albeit maintaining its original meaning):

$$\mathbf{X} = \begin{bmatrix} | & | & & | \\ g(\mathbf{x}_1) & g(\mathbf{x}_2) & \cdots & g(\mathbf{x}_m) \\ | & | & & | \end{bmatrix} \quad \mathbf{X}' = \begin{bmatrix} | & | & & | \\ g(\mathbf{x}'_1) & g(\mathbf{x}'_2) & \cdots & g(\mathbf{x}'_m) \\ | & | & & | \end{bmatrix} \quad (3.46)$$

where  $\mathbf{g} = [g_1(x) \ g_2(x) \ \cdots \ g_p(x)]^T$ .

The algorithm proceeds in a similar fashion as before. The projection subspace  $\mathbf{Q}$  corresponds to the left singular vectors resulting from the SVD of  $\mathbf{X}$ :

$$\min_{\begin{bmatrix} \tilde{\mathbf{A}} & \tilde{\mathbf{B}} \\ \tilde{\mathbf{C}} & \tilde{\mathbf{D}} \end{bmatrix}} \left\| \begin{bmatrix} \mathbf{X}' \\ \mathbf{Y} \end{bmatrix} - \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{A}} & \tilde{\mathbf{B}} \\ \tilde{\mathbf{C}} & \tilde{\mathbf{D}} \end{bmatrix} \begin{bmatrix} \mathbf{U}^* & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \end{bmatrix} \right\|_F^2 \quad (3.47)$$

By defining the first observable  $g_1(\mathbf{x}) = \mathbf{x}$  as in (3.45), it is possible to retain the state of interest, and still benefit from the DMD extension by a careful choosing of the remaining nonlinear observables.

In the methodology defined in [6], functions  $g$  may not only represent non linear functions of the state but as well physical measurements, such as outputs. In the case of a high dimensional system, the number of outputs  $l$  might be significantly smaller than the number of states  $n$ . Therefore, there may not be the need for SVD and projection in to low order subspace. The relation in (3.47) can be changed to accommodate physical measurements, by defining what are termed *deterministic states*:  $X_d \in \mathbb{R}^d$ . Assembling the snapshot matrices of such variables and inserting them into (3.47), the least squares problem boils down to:

$$\min_{\begin{bmatrix} \tilde{\mathbf{A}} & \tilde{\mathbf{B}} \\ \tilde{\mathbf{C}} & \tilde{\mathbf{D}} \end{bmatrix}} \left\| \begin{bmatrix} \mathbf{X}'_d \\ \mathbf{X}' \\ \mathbf{Y} \end{bmatrix} - \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{U} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{A}}_d & \mathbf{A}_{nd} & \tilde{\mathbf{B}}_d \\ \mathbf{A}_d & \tilde{\mathbf{A}}_{nd} & \tilde{\mathbf{B}}_{nd} \\ \tilde{\mathbf{C}}_d & \tilde{\mathbf{C}}_{nd} & \tilde{\mathbf{D}} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{U}^* & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{X}_d \\ \mathbf{X} \\ \mathbf{Y} \end{bmatrix} \right\|_F^2 \quad (3.48)$$

where the subscript  $d$  refers to a term coupled with a deterministic state and the subscript  $nd$  refers to a term coupled with a non deterministic state. The reduced order model is then computed by least squares:

$$\Theta_{\text{EIODMD}} = \begin{bmatrix} \mathbf{X}'_d \\ \mathbf{U}^* \mathbf{X}' \\ \mathbf{Y} \end{bmatrix} \begin{bmatrix} \mathbf{X}_d \\ \Sigma \mathbf{V} \\ \mathbf{Y} \end{bmatrix}^\dagger \quad (3.49)$$

The reduced order matrices are then:

$$\tilde{\mathbf{A}} = \begin{bmatrix} \tilde{\mathbf{A}}_d & \mathbf{A}_{nd} \\ \mathbf{A}_d & \tilde{\mathbf{A}}_{nd} \end{bmatrix}, \quad \tilde{\mathbf{B}} = \begin{bmatrix} \tilde{\mathbf{B}}_d \\ \tilde{\mathbf{B}}_{nd} \end{bmatrix}, \quad \tilde{\mathbf{C}} = \begin{bmatrix} \tilde{\mathbf{C}}_d & \tilde{\mathbf{C}}_{nd} \end{bmatrix}, \quad \tilde{\mathbf{D}} = \tilde{\mathbf{D}} \quad (3.50)$$

where  $\tilde{\mathbf{A}} \in \mathbb{R}^{(d+r) \times (d+r)}$ ,  $\tilde{\mathbf{B}} \in \mathbb{R}^{(d+r) \times q}$ ,  $\tilde{\mathbf{C}} \in \mathbb{R}^{l \times (d+r)}$  and  $\tilde{\mathbf{D}} \in \mathbb{R}^{l \times q}$ .

This extension of IODMD attempts to **increase state reconstruction accuracy**: the generalization of DMD to include non linear observables, having as a starting point an analysis of the nonlinear terms of the governing non linear equations, attempts to improve the embedding of the non linear dynamics in the linear model, thus improving the predictability of the model's states.

## 3.4 Regularization

DMD-based identification methods, such as IODMD and its extended version, **do not guarantee that its resulting models are stable**. To enforce stability, an additional post-processing step is required. In [8] it is shown that directly solving a nonsmooth constrained optimization problem can be a viable approach for stabilizing IODMD-derived systems while retaining the salient properties for approximating the output response. This is achieved by using an open-source MATLAB code, GRANSO (Gradient-based Algorithm Non-Smooth Optimization).

A simpler, although model dependent post processing tool is here developed and used in the context of this thesis, based on the ideas in [68]. In the former, stability of an estimated system - from least squares and a finite number of data points - is imposed by adding a regularization term to the least squares cost function. Given the least squares problem to identify the system matrices ( $\hat{\mathbf{A}}$ ,  $\hat{\mathbf{B}}$ ,  $\hat{\mathbf{C}}$ ,  $\hat{\mathbf{D}}$ ):

$$\min_{\hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{C}}, \hat{\mathbf{D}}} \left\| \begin{bmatrix} \hat{\mathbf{X}}_{i+1} \\ \mathbf{Y}_{i|i} \end{bmatrix} - \begin{bmatrix} \hat{\mathbf{A}} & \hat{\mathbf{B}} \\ \hat{\mathbf{C}} & \hat{\mathbf{D}} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{X}}_i \\ \mathbf{U}_{i|i} \end{bmatrix} \right\|_F^2 \quad (3.51)$$

its solution can be presented as two solutions for independent least squares problems:

$$\begin{aligned} \begin{bmatrix} \hat{\mathbf{A}} & \hat{\mathbf{B}} \end{bmatrix} &= \hat{\mathbf{X}}_{i+1} \begin{bmatrix} \hat{\mathbf{X}}_i^T & \mathbf{U}_{i|i}^T \end{bmatrix} \hat{\Sigma}_{XU}^{-1} \\ \begin{bmatrix} \hat{\mathbf{C}} & \hat{\mathbf{D}} \end{bmatrix} &= \mathbf{Y}_{i|i} \begin{bmatrix} \hat{\mathbf{X}}_i^T & \mathbf{U}_{i|i}^T \end{bmatrix} \hat{\Sigma}_{XU}^{-1} \end{aligned} \quad (3.52)$$

where

$$\hat{\Sigma}_{XU} = \begin{bmatrix} \hat{\mathbf{X}}_i \hat{\mathbf{X}}_i^T & \hat{\mathbf{X}}_i \mathbf{U}_{i|i}^T \\ \mathbf{U}_{i|i} \hat{\mathbf{X}}_i^T & \mathbf{U}_{i|i} \mathbf{U}_{i|i}^T \end{bmatrix} \quad (3.53)$$

Stability of the model is then imposed by adding a regularization term to the cost function from which (3.52) is determined. For a full explanation on the cost function formulation and its rationale the reader is

referred to [68]. The optimal solution is then given by:

$$\begin{aligned} \begin{bmatrix} \tilde{\mathbf{A}}_c & \tilde{\mathbf{B}}_c \end{bmatrix} &= \begin{bmatrix} \hat{\mathbf{X}}_{i+1} \hat{\mathbf{X}}_i^T & \hat{\mathbf{X}}_{i+1} \mathbf{U}_{i|i}^T \end{bmatrix} \left( \hat{\Sigma}_{XU} + c \mathbf{W}_e^{-1} \right) \\ &= \begin{bmatrix} \hat{\mathbf{A}} & \hat{\mathbf{B}} \end{bmatrix} \hat{\Sigma}_{XU} \left( \hat{\Sigma}_{XU} + c \mathbf{W}_e \right)^{-1} \end{aligned} \quad (3.54)$$

where  $\hat{\Sigma}_{XU}$  is defined in (3.53) and  $\mathbf{W}_e$  as:

$$\mathbf{W}_e = \begin{bmatrix} \mathbf{W} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \quad (3.55)$$

The amount of regularization is characterized by the positive real scalar  $c$  and by the positive definite matrix  $\mathbf{W}$ . The same principle applied to subspace identification methods can be used in the IODMD models derivations. The analogous regularized solution in (3.54), in the context of IODMD, using the full explicit formulation in (3.36), is the following:

$$\begin{bmatrix} \tilde{\mathbf{A}}_c & \tilde{\mathbf{B}}_c \\ \tilde{\mathbf{C}}_c & \tilde{\mathbf{D}}_c \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{A}} & \tilde{\mathbf{B}} \\ \tilde{\mathbf{C}} & \tilde{\mathbf{D}} \end{bmatrix} \begin{bmatrix} \Sigma \mathbf{V} \Sigma^T \mathbf{V}^T & \Sigma \mathbf{V} \mathbf{R}^T \\ \mathbf{R} \Sigma^T \mathbf{V}^T & \mathbf{R} \mathbf{R}^T \end{bmatrix} \left( \begin{bmatrix} \Sigma \mathbf{V} \Sigma^T \mathbf{V}^T & \Sigma \mathbf{V} \mathbf{R}^T \\ \mathbf{R} \Sigma^T \mathbf{V}^T & \mathbf{R} \mathbf{R}^T \end{bmatrix} + c \begin{bmatrix} \mathbf{W} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \right)^{-1} \quad (3.56)$$

Matrices  $\tilde{\mathbf{A}}_c$ ,  $\tilde{\mathbf{B}}_c$ ,  $\tilde{\mathbf{C}}_c$  and  $\tilde{\mathbf{D}}_c$  correspond to the state space matrices after the post processing step.

**Although the least squares solution might be a stable one, in the event that it is not, using (3.56) ensures IODMD and EIODMD stability whilst having little or no effect in the outputs and states.**

In this chapter, the data driven techniques appropriate for the modelling of high dimensional non linear systems are reviewed, along with the derivations suited for control. The connections of DMD to the Koopman Operator are discussed and evidence of how a simple linear system is able to provide an accurate description of a complex system is given.

In the next chapter, IODMD is used to build a model suitable for the design of controllers and the ideas of the Koopman Operator are further explored to increase the model's accuracy.

# CHAPTER 4

## WIND FARM MODEL FOR YAW CONTROL

---

### 4.1 Modelling

The system of interest in this work is a small wind farm, where two turbines are placed apart, and it is schematically represented in Figure 4.1.

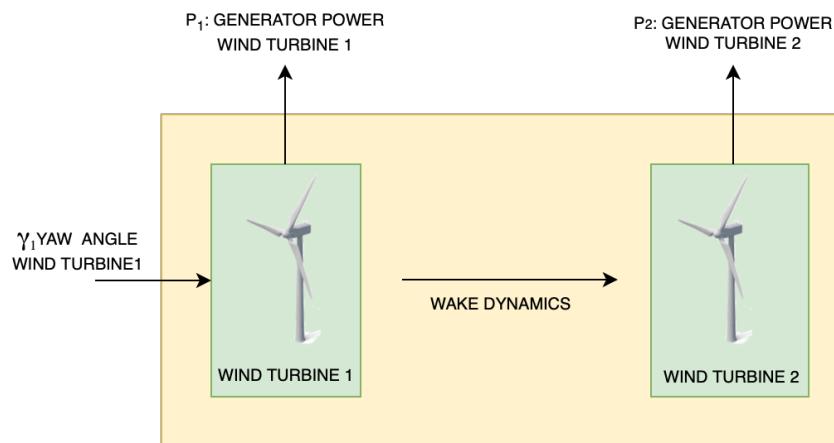


Figure 4.1: Wind Farm system with 1 input - first turbine yaw angle - and 2 outputs - generator power of both turbines.

Several variables within the wind farm are subject to variations, such as the (1) power output of each turbine (2) the loads on each turbine (3) the wind characteristics at different spatial points, amongst others. The model aims to relate and explain such variations. **The directly manipulated variable - the input - will be the yaw angle - and the outputs the generator powers of both turbines.**

Various routes exist for the task of model building. For the purpose of this thesis the route of *system identification* is used. This methodology is focused on experimentation, where input and output signals from the system are recorded and subjected to data analysis so as to infer a model. System Identification is a methodology by which one can generate a set of models based on the observed input-output data from the system and choose the best model(s) in accordance to a defined criteria. Therefore, the problem of system identification is specified by three elements [37, 44]: (1) a data set  $\mathcal{D}$  obtained by input-output measurements (2) a set of candidate models  $\mathcal{M}$  and (2) a criterion  $\mathcal{L}$  to evaluate candidate models and select the best model(s).

A flow diagram of the typical system identification procedure adopted is represented in Figure 4.2. For each step of the process, a further breakdown is provided in the upcoming sections and the main results are presented.

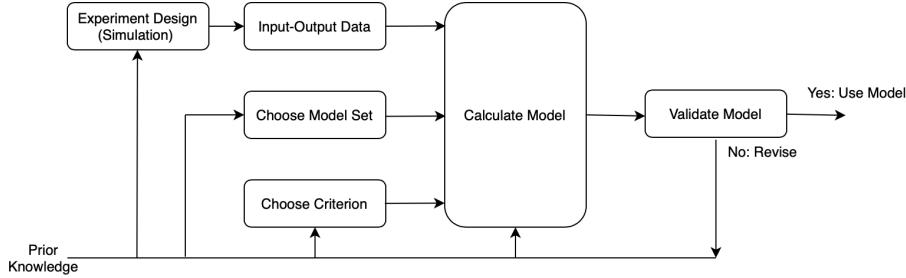


Figure 4.2: The system identification loop, adapted from [37, 44].

#### 4.1.1 Step One: Simulation Design

The input output data set  $\mathcal{D}$  is obtained by performing a CFD simulation in SOWFA. SOWFA is a Large Eddy Simulation (LES) of the three-dimensional wind flow around one or more turbine rotors in the Atmospheric Boundary Layer (ABL)<sup>1</sup> developed by the United States NREL. In the LES method, in order to limit the computational cost of the simulation, the larger scales of the flow field are resolved by solving the temporally and spatially unsteady Navier-Stokes equations (with additional stress terms that empirically model the effect of the turbulence in the smaller unresolved scales), as described in [28]. The rotating rotor blades are modelled through an actuator line approach. The actuator lines are coupled with the FAST turbine aeroelastic simulator tool which calculates the loads, power and rotor speed of each turbine. SOWFA is a widely used tool in the literature to obtain accurate data. More information can be found in [20].

In this work, wind with a uniform inflow profile is used to demonstrate the working principles of DMD. The case of a wind farm with two turbines is studied, to assess the gain in energy in the second turbine. The second turbine is situated 5 rotor diameters (5D) behind the upstream turbine. The turbine used for the simulation is the DTU 10MW turbine, with a rotor diameter of 178.3 [m]. Detailed design characteristics of the turbine can be found in [73].

The domain size used for IODMD has dimensions of approximate 1.71 [km] in the  $x$  direction (lengthwise), 343 [m] in the  $y$  direction and 300 [m] in the  $z$  direction. The domain is represented in Figure 4.3. The dimensions have been adimensionalized by the rotor diameter. A total of 550 points in  $x$ , 111 points in  $y$  and 99 points in  $z$  are sampled during the simulation, which means the complete flow field is solved in each of these points. Thus, a total of 6 million points, each one containing information on the  $(u, v, w)$  velocity components is taken.

Due to the large volume of information generated during the simulation, a pre processing step of the data is performed in a cluster, aiming to retrieve only information at each fourth point. This decreases the total number of points from 6 million to roughly 100 thousand samples, decreasing computational effort.

<sup>1</sup> Lowest region of the atmosphere where Earth's surface strongly influences temperature and wind through the turbulent transfer of air mass.

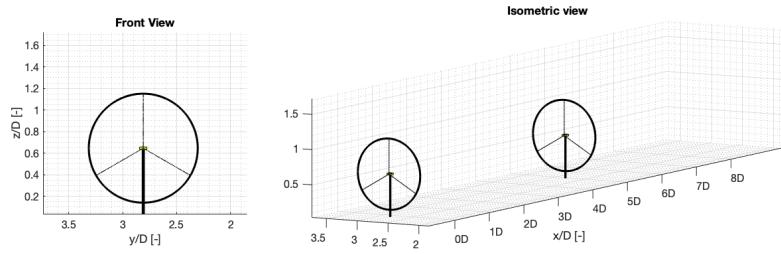


Figure 4.3: Two turbine set layout in SOWFA. On the left, front view of upstream turbine is shown. On the right, a three dimensional perspective is visible, depicting the distance of five rotor diameters (890 metres) between upstream and downstream turbine.

Flow field information is sampled at each 2 seconds, which corresponds to a sampling frequency of 0.5 Hz. According to [1, 4], where AIC is analysed, the majority of the frequency content within a wake 4 rotor diameters downstream of the upstream turbine has a lower frequency than 0.5 Hz, concentrating around 0.25 Hz. Higher frequencies are related with the presence of a shear layer. The decision of using a sampling period  $T_s$  of 2 seconds is also related with computational effort, as using a lower one would increase the total number of data points and using IODMD could become infeasible on a normal computer. In summary, the **sampling period chosen is a trade-off in terms of capturing the highest frequencies and avoiding extreme computational costs**, although it is believed the most relevant frequencies are captured and the Nyquist criterion<sup>2</sup> satisfied.

In addition to flow field, a variety of data on the turbines is also saved. More specifically, information of the blades, nacelle, rotor, tower and generator. These information is sampled at each 0.2 seconds.

A simulation of 2000 seconds (approximately 33 minutes of real time simulated) is performed where Randomized Binary Signals (RBS)<sup>3</sup> provided as a setpoint for the yaw angle of the first turbine  $\gamma_{t1}$  serve as inputs. The procedure is depicted in Figure<sup>4</sup> 4.4. The yaw angle is first set to -10 degrees, and then varied to -30 degrees randomly throughout the simulation. **The yaw of the second turbine  $\gamma_{t2}$  remains the same throughout the simulation.**

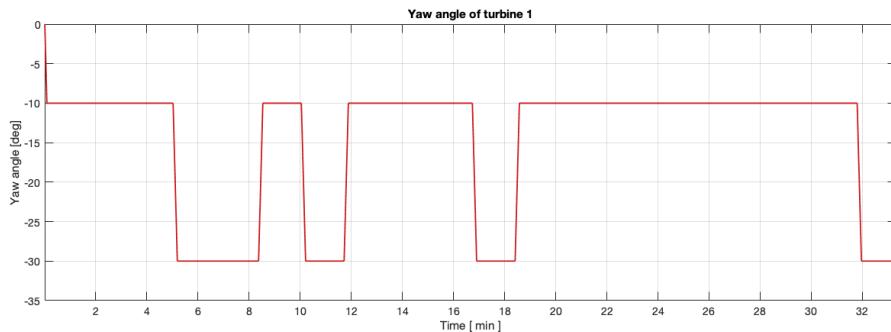


Figure 4.4: Yaw angle variation of upstream turbine  $\gamma_{t1}$  during simulation. The yaw angle is first set to -10 degrees at the beginning of the simulation and then randomly varied to -30 degrees

<sup>2</sup>The Nyquist criterion states that a repetitive waveform can be correctly reconstructed provided that the sampling frequency is greater than double the highest frequency to be sampled.

<sup>3</sup>Random process that can assume one of two possible values at any time.

<sup>4</sup>Turbine information is saved in files automatically generated by SOWFA, which are then read using MATLAB functions developed by the NREL.

### 4.1.2 Step Two: Simulation and Data Record

The resulting generator power of both turbines during the simulation varies as a result of the yaw angle and wake deflection. It is intended to model these dynamics in order to accurately predict the power generated in each turbine, which is illustrated in Figure 4.5.

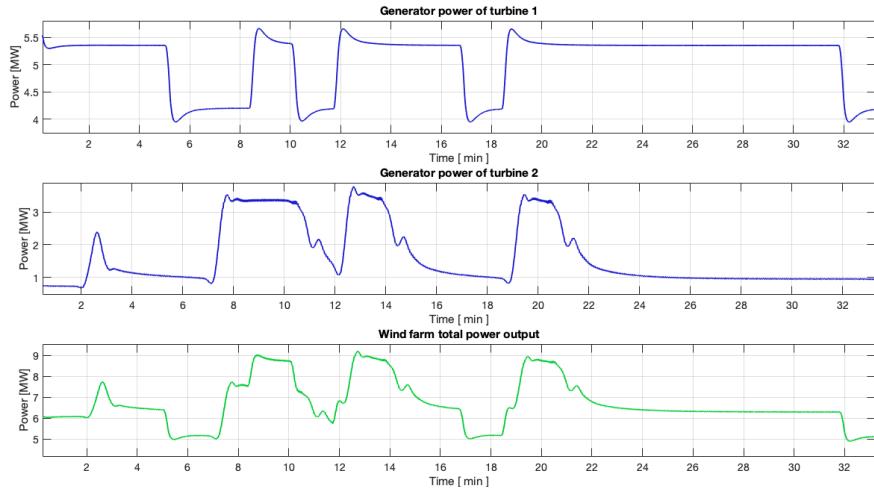


Figure 4.5: Generator power variations of both turbines (in blue) during simulation and total generated power at wind farm level (in green).

In the first five minutes of the simulation, the first turbine yaw angle is set to -10 degrees. These procedure has direct impact on the power produced in the downstream turbine: as the wake travels, a spike in power appears at 2 minutes and 30 seconds. This time interval serves the purpose of parameter setting and ensures that the wake is fully developed. The RBS used for system identification starts acting after the initial five minutes.

In the last graph of Figure 4.5, **the effectiveness of the wind farm control strategy can be observed**. Using greedy control ( $\gamma_{t1}$  set to zero) turbine 1 captures 5.49 [MW] and turbine 2 0.77 [MW]. Using an offset of -10 degrees leads to turbine 1 capturing 5.35 [MW] and turbine 2 0.96 [MW] - a total collective power increase of 0.8% from 6.26 [MW] to 6.31 [MW]. By further yawing the upstream turbine -20 deg, the last graph puts in evidence the rationale of the WRC strategy. Turbine 1 is sacrificed, generating 4.20 [MW] in the fully yawed setting. Turbine 2, on the other hand, compensates for the lost 1.29 [MW], by capturing 3.36 [MW] - an additional 2.6 [MW] - with relation to the greedy control scenario. This corresponds to a **power production increase of 20.77% from 6.26 [MW] to 7.56 [MW] at the wind farm level**.

The observed dynamics can be explained by inspecting the wake's behaviour as it propagates downstream. Figures 4.6 and 4.7<sup>5</sup> are reconstructions from the simulation and allow to visually comprehend the relation between the wake's characteristics and the resulting generator power. In blue the isosurface of the absolute vorticity field with value 5.65 is represented, and it becomes clear how **the deflection of the wake leads to increased power**. Yellow arrows represent the thrust force direction.

<sup>5</sup>These snapshots were obtained using the function `wake_vorticity_deflection_yaw.m`, developed in the context of this work.

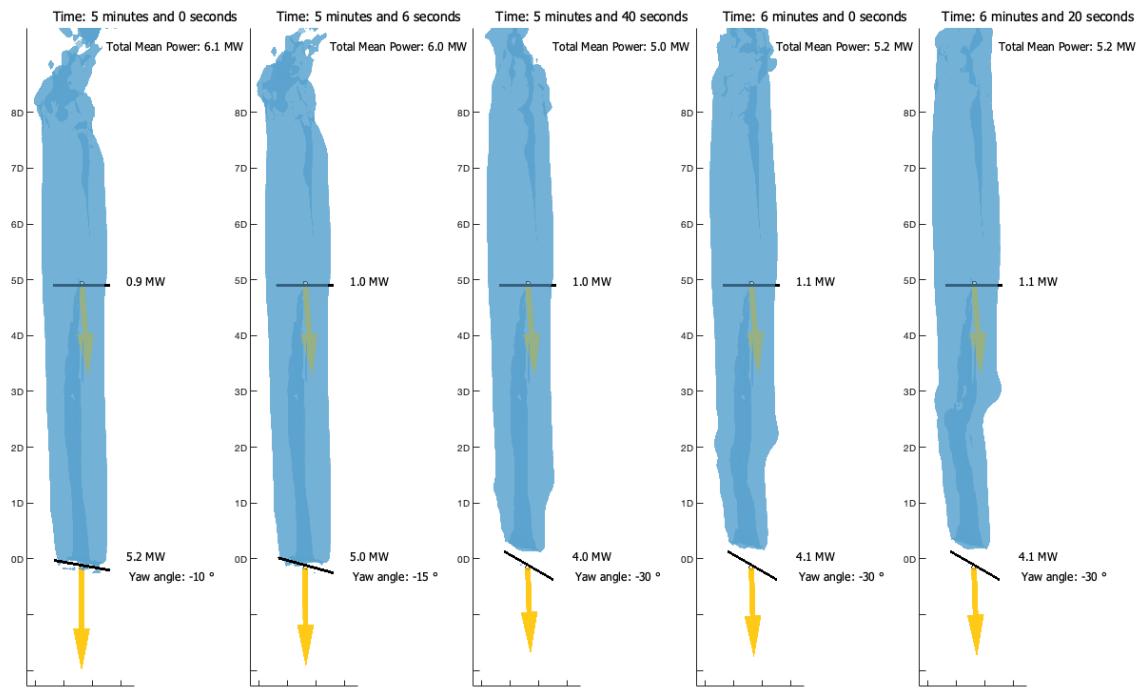


Figure 4.6: Wake deflection dynamics: (1) turbine starts yawing from -10 degrees (2) continues to yaw until reaching fully yawed setting of -30 degrees (3) rotor is fully yawed and wake shows initial signs of deflection to the left (4) wake starts propagating downstream (5) wake continues to propagate, reaching a distance of 3 rotor diameters in more than 1 minute after being fully yawed.

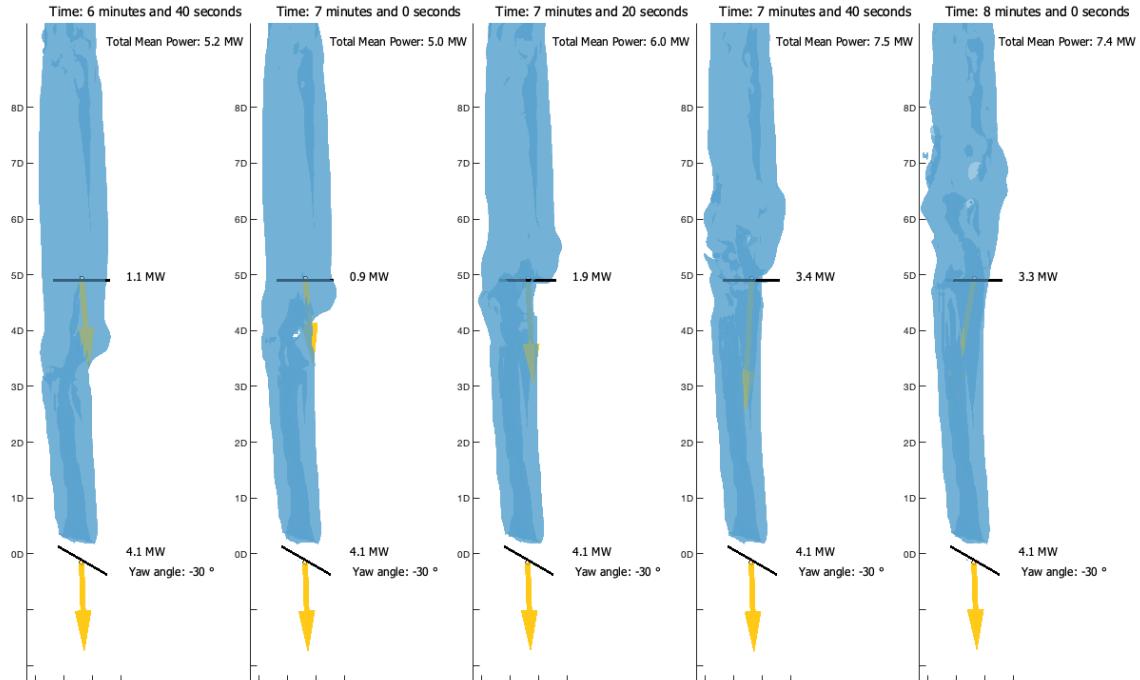


Figure 4.7: Wake deflection dynamics: (1) and (2) wake continues to propagate and deflect (3) wake reaches downstream turbine rotor, and increase of 1 [MW] is registered (4) turbine 2 power output increases to 3.3 [MW], as deflected wake overlaps downstream rotor less (5) wake continues to propagate behind turbine 2 and power output maintains a maximum of approximately 3.3 [MW].

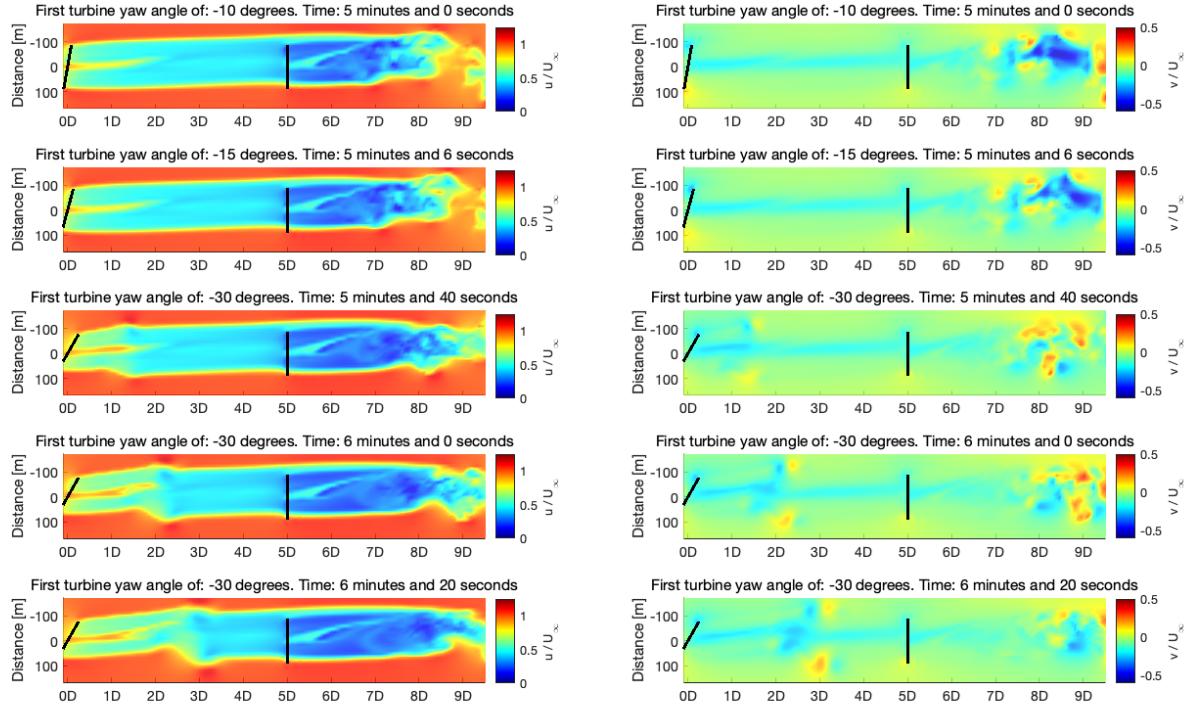


Figure 4.8: Wake deflection dynamics (normalised streamwise and spanwise velocities): (1) and (2) first turbine starts yawing (3) turbine is fully yawed -30 degrees and shows signs of initial deflection (4) and (5) wake continues to propagate downstream. Wake edges represented in yellow - normalized streamwise velocity of 0.75 - mark the inside velocity deficit and the outer undisturbed flow field.

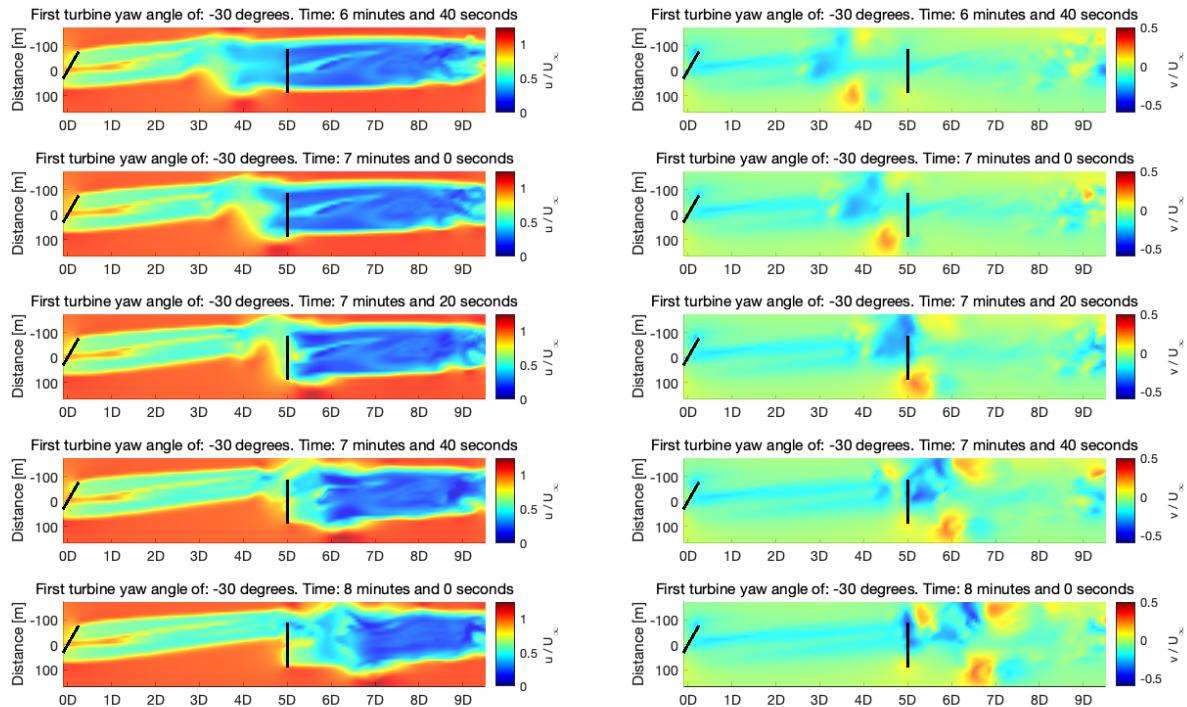


Figure 4.9: Wake deflection dynamics (normalised streamwise and spanwise velocities): (1), (2) and (3) wake continues to propagate downstream, reaching second turbine's rotor two minutes and thirty seconds later (4) and (5) deflected wake is fully developed, and velocity deficit affecting downstream turbine is noticeably reduced.

Figures 4.8 and 4.9<sup>6</sup> provide a quantitative overview on the wake's characteristics under the control strategy being analyzed. The snapshots are taken at hub height<sup>7</sup> and both the streamwise  $u$  and spanwise  $v$  velocities, normalized by the upstream velocity  $U_\infty$ , are represented. These two components play a major role in the dynamics of the wake.

### 4.1.3 Step Three: Data Pre-Processing

The data requires to be preprocessed before being used to calculate candidate models. This process includes choosing only the most relevant information, resampling turbine information and removing trends and offsets from data.

1. **Use flow field information between first and second turbines:** the domain size used for the simulation in SOWFA captures the flow information behind the second turbine. Nevertheless, considering that the focus is on the wake dynamics between the first and second turbine - which explains power variations on the second - only these data is used to build the models. Furthermore, the initial time interval where the first turbine is being set to -10 degrees is removed.
2. **Resample turbine information:** velocity data is sampled at every 2 seconds, whereas turbine data is sampled at every 0.2 second. To compute the model, turbine data is resampled, so that both sampling frequencies are equal.
3. **Remove trends and offsets from input-output information:** a common procedure in systems identification is to remove offsets and trends from the data, so that the model focuses on the true intrinsic dynamics of the system. For this reason, turbine data used for input-output measurements is detrended by removing the mean value. The mean value corresponds to the power produced at each turbine when the first is yawed -10 degrees and the wake is fully developed. For the upstream turbine, this value is equal to 5.35 [MW]. For the downstream turbine, the value is equal to 0.95 [MW]. The model is build using the input output turbine measurements in Figure 4.10.

The third rationale may be applied to the flow field information. In the POD methodology, it is common to subtract the mean. However, subtracting a mean of the data reduces DMD to the temporal discrete Fourier transform, which may be restrictive and undesirable, as indicated in [19]. An alternative is to **subtract an equilibrium point of the dynamics**. This is often the case for fluid flows, where the equilibrium solution is a steady flow - commonly referred to as *base flow*  $x_b$ . As the author in the previous article notes, subtracting the equilibrium appears to have significant advantages over using the mean. Three quantities are defined to distinguish the data to be used:  $(u, v, w)_k$  refers to the velocity vector from the simulation at each time instant,  $(\bar{u}, \bar{v}, \bar{w})$  to the base flow, and  $(u', v', w')_k$  to the difference between the velocity vector at each time instant and the base flow. The last may be interpreted as an unsteady flow, somewhat similar to the terms in the RANS equations. The choosing of the base flow is subjective. In the context of this dissertation, it corresponds to the wake's characteristics for a yaw angle  $\gamma$  of -10 degrees.

---

<sup>6</sup>These snapshots were obtained using the function *hubheightcut\_yaw.m*, developed in the context of this thesis

<sup>7</sup>Distance from the turbine platform to the rotor of an installed wind turbine, indicating how high the turbine stands above the ground (not including the length of the turbine blades)

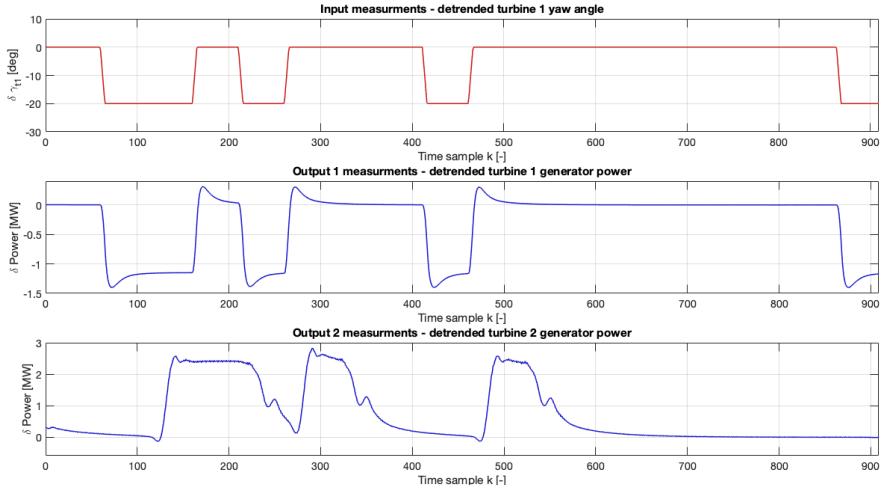


Figure 4.10: Input-output measurements to be used for systems identification: detrended and resampled turbine 1 yaw angle (orange) and generator power of both turbines during identification simulation (blue)

#### 4.1.4 Step Four: Model Selection

The candidate models are obtained in the well-known ***state space representation***, often used in control engineering. The ***state space description provides the dynamics as a set of coupled first-order differential equations in a set of internal variables known as state variables***, together with a set of algebraic equations that combine the state variables into physical output variables. The state variable is a physical quantity which, in the absence of external excitation, completely determines the evolution of the system. The set of  $n$  coupled first-order differential equations, known as state equations, follow the general form:

$$\begin{aligned} \dot{x}_1 &= f_1(\mathbf{x}, \mathbf{u}, t) \\ \dot{x}_2 &= f_2(\mathbf{x}, \mathbf{u}, t) \\ &\vdots = \vdots \\ \dot{x}_n &= f_n(\mathbf{x}, \mathbf{u}, t) \end{aligned} \tag{4.1}$$

where  $\dot{x}_i = dx_i/dt$  and each of the functions may be a general nonlinear, time varying function of the state variables  $\mathbf{x}$  and the system input  $\mathbf{u}$ . State equations are commonly written in vector form, with the set of  $n$  state variables written as a ***state vector***  $\mathbf{x}(t) = [x_1(t), x_2(t), \dots, x_n(t)]^T$ , and the set of  $q$  inputs written as an input vector  $\mathbf{u}(t) = [u_1(t), u_2(t), \dots, u_q(t)]^T$ . The models used in this work are Linear Time Invariant (LTI), that is the system is described by linear differential equations with constant coefficients. For a LTI system of order  $n$  and with  $q$  inputs, equations in (4.1) become a set of  $n$  first order linear differential equations with constant coefficients:

$$\begin{aligned} \dot{x}_1 &= a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n + b_{11}u_1 + \dots + b_{1r}u_q \\ \dot{x}_2 &= a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n + b_{21}u_1 + \dots + b_{2r}u_q \\ &\vdots = \vdots \\ \dot{x}_n &= a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n + b_{n1}u_1 + \dots + b_{nr}u_q \end{aligned} \tag{4.2}$$

where the coefficients  $a_{ij}$  and  $b_{ij}$  are constants that describe the system. This set of  $n$  equations defines the derivatives of the state variables to be a weighted sum of the state variables and the system inputs. Equations (4.2) may be written compactly in a matrix form:

$$\frac{d}{dt} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} b_{11} & \cdots & b_{1q} \\ b_{21} & \cdots & b_{2q} \\ \vdots & \ddots & \vdots \\ b_{n1} & \cdots & b_{nq} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_q \end{bmatrix} \quad (4.3)$$

which may be summarized as:

$$\dot{x} = \mathbf{Ax} + \mathbf{Bu} \quad (4.4)$$

where the state vector  $x$  is a column vector of length  $n$ , the input vector  $u$  is a column vector of length  $q$ ,  $\mathbf{A}$  is a  $n \times n$  square matrix of the constant coefficients  $a_{ij}$  and  $\mathbf{B}$  is a  $n \times q$  matrix of the coefficients  $b_{ij}$  that weigh the inputs.

The system output, defined to be any variable of interest, can be represented by a linear combination of the state variables  $x_i$  and the system inputs  $u_i$ . Writing the equivalent equations in a compact form yields:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_l \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ c_{l1} & c_{l2} & \cdots & c_{ln} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} d_{11} & \cdots & d_{1q} \\ d_{21} & \cdots & d_{2q} \\ \vdots & \ddots & \vdots \\ d_{n1} & \cdots & d_{nq} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_q \end{bmatrix} \quad (4.5)$$

which can be further summarized in a compact notation:

$$y = \mathbf{Cx} + \mathbf{Du} \quad (4.6)$$

where  $y$  is a column vector of the output variables  $y_i(t)$ ,  $\mathbf{C}$  is a  $l \times n$  matrix of the constant coefficients  $c_{ij}$  that weight the state variables, and  $\mathbf{D}$  is an  $l \times q$  matrix of the constant coefficients  $d_{ij}$  that weight the system inputs. In discrete time, the differential equations become finite differences, and the complete system model for the LTI system is defined by:

$$\begin{aligned} x_{k+1} &= \mathbf{Ax}_k + \mathbf{Bu}_k \\ y_k &= \mathbf{Cx}_k + \mathbf{Du}_k \end{aligned} \quad (4.7)$$

Matrices  $\mathbf{A}$  and  $\mathbf{B}$  are properties of the system and are determined by the system structure and elements. The output equation matrices  $\mathbf{C}$  and  $\mathbf{D}$  are determined by the particular choice of output variables. All four matrices are computed using the algorithms described in Section 3.

#### 4.1.5 Step Five: Criterion of Fit

The fitness of the model can be measured in two terms. Firstly, it can be **measured in its ability to map the inputs to the outputs**, functioning as a good predictor of the generated power in both turbines

as a function of the first turbine yaw angle. For this purpose, the criteria to use is Variance Accounted For (VAF), defined as [70]:

$$\text{VAF}_i(y_i(k), \hat{y}_i(k, \theta)) = \left( 1 - \frac{\frac{1}{N} \sum_{k=1}^N |y_i(k) - \hat{y}_i(k, \theta)|^2}{\frac{1}{N} \sum_{k=1}^N |y_i(k)|^2} \right) \cdot 100 \quad [\%] \quad (4.8)$$

where  $y_i(k)$  is the  $i$ th measured output at time instant  $k$ ,  $\hat{y}_i(k)$  the  $i$ th predicted output at time instant  $k$ ,  $\theta$  the model parameters and  $N$  the number of samples. The VAF criteria gives a quantitative measure of the discrepancy between the model and data. Higher percentages of VAF indicate a stronger strength of association. Secondly, it can be **measured in its ability to reconstruct the high order representation of the states**  $x_k$ , functioning as a good predictor of the wake dynamics as a function of the first turbine yaw angle. For this purpose, the criteria to use is the Normalized Root Mean Squared Error (NRMSE), defined as:

$$\text{NRMSE}_k(\mathbf{x}(k), \hat{\mathbf{x}}(k, \theta)) = \frac{\text{RMSE}(\mathbf{x}(k), \hat{\mathbf{x}}(k, \theta))}{x_{max}(k) - x_{min}(k)} = \frac{\sqrt{\frac{\sum_{i=1}^N (x_i(k) - \hat{x}_i(k, \theta))^2}{N}}}{x_{max}(k) - x_{min}(k)} \cdot 100 \quad [\%] \quad (4.9)$$

where  $\mathbf{x}(k)$  is the state vector at time instant  $k$ ,  $\hat{\mathbf{x}}(k)$  the predicted state vector at time instant  $k$ ,  $\theta$  the model parameters,  $N$  the number of samples and the subscript  $i$  refers to the  $i$ th state in the state vector. The NRMSE is computed using the representation of the high order state vector at each time instant, and its mean value (considering the NRMSE from all time instants)  $\overline{\text{NRMSE}}$  is taken to serve as a fit criteria.

Considering that the final purpose of the model is to be used for control of the total power generated by both turbines, **the most decisive criteria is VAF**, and the  $\overline{\text{NRMSE}}$  is only computed for the best model, as measured by VAF.

#### 4.1.6 Step Six: Model Calculation

The input-output DMD variants are used to compute the wind farm state-space ROM. As discussed in Subsection 3.3.4, **the Koopman operator generalizes DMD to allow any observables to serve as a state**. This decision process of choosing the states is here discussed. Note that input-output information is maintained fixed, corresponding to the data in Figure 4.10. The algorithm to calculate the best model is described in Annex C, namely algorithm C.1, *dynamicmodedecomposition.m*. **In order to establish a baseline, IODMD is firstly used, where the streamwise velocity  $u$  is used for the states**. This corresponds to the traditional approach in the literature [4, 33]. This methodology is dubbed  $\text{IODMD}_u$  for the remaining of the next sections. To evaluate the possible benefits of using alternatives to  $\text{IODMD}_u$  motivated by the Koopman operator to better reproduce the dynamics, three different pathways can be followed to calculate the models:

- 1. Use any variable or combination of variables as states:** instead of using  $u$  as a state, this is replaced by another variable, such as  $v$  or  $w$  or even a combination of all. This attempts to find the ones that are most indicative to recover the dynamics. The results of this first formulation are

referred to as  $\text{IODMD}_{xx}$ , where  $xx$  is the observable used in replacement of  $u$ .

2. **Augment the state vector  $u$  by another flow variable:** this process allows to both maintain an interpretable model and gain insights into the physics of the problem, whilst attempting to improve the accuracy of the model based on the concepts behind the Koopman operator theory. More specifically, reintroducing DMD notation,  $g_1(x)$  is chosen to be  $u$  and combinations with different  $g_2(x)$  are experimented. This formulation is termed  $\text{EIODMD}_{xx,yy}$ , where  $xx$  is the first observable and  $yy$  the second observable used to augment the original snapshot data matrix.
3. **Augment the state vector  $u$  by any variable:** the third hypothesis focuses on augmenting the flow field information with turbine information, following the ideas laid out in [6].

To build the models, the SVD is taken by choosing a certain truncation level  $r$ . Singular values (and the corresponding right and left singular vectors) are continuously added to form state space models of increasing size. It follows that a total of  $r$  models are calculated, each one of increasing size, in order to **find the optimal number of modes that best replicates the dynamics**. A truncation  $r$  of the SVD equal to 100 is chosen. Empirical evidence in the literature, such as [2, 33], shows that even models with no more than 20 states can accurately reproduce the dynamics, hence 100 should be enough to compute satisfactory models and understand the impact of adding increasing states.

#### 4.1.7 Step Seven: Model Validation

To validate models a novel data set is obtained. A new simulation is then performed, where RBS are again used to perturb the system within the same range of values used for identification. Input-output information is then pre-processed in a similar fashion as the data used to identify the models. Post processed signals used to validate the models are depicted in Figure 4.11.

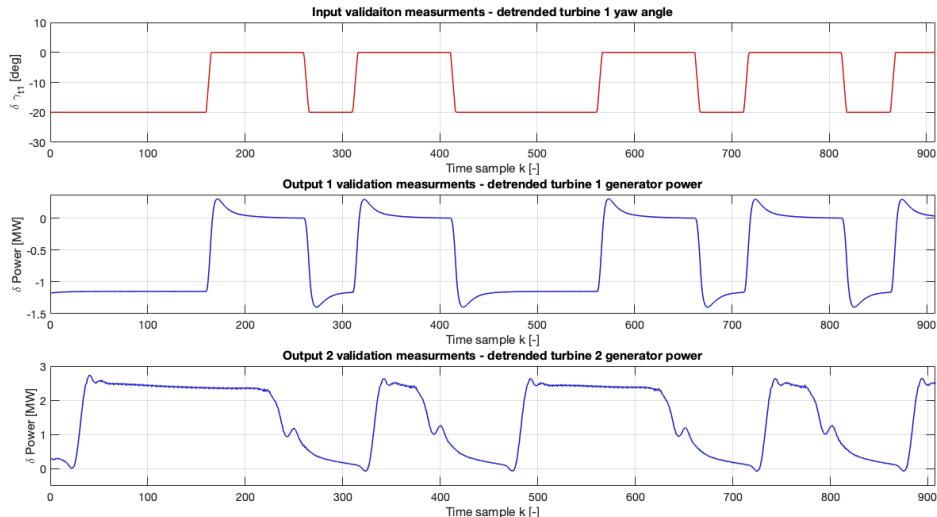


Figure 4.11: Input-output measurements to be used for model validation for wake redirection control: detrended and resampled turbine 1 yaw angle (orange) and generator power of both turbines during validation simulation (blue).

The models are simulated by using the inbuilt MATLAB function `/sim`, which simulates the discrete

linear system to arbitrary inputs, in this case turbine 1 yaw angle validation measurements. The initial condition  $\tilde{x}_o$ , meaning the model state vector at the beginning of the simulation, is computed by projecting the flow field at the initial instant onto the low dimension subspace, by doing  $\tilde{x}_o = \mathbf{U}^* \mathbf{x}_o$ .

The model outputs are then compared to output validation measurements, by using the criteria in (4.8). In addition, the *lsm* function provides the state trajectory<sup>8</sup>, and by using the projection matrix the flow field can be recovered  $\mathbf{x}_k = \mathbf{U}\tilde{\mathbf{x}}_k$ . This means that the ability of the ROM to construct the wake can be evaluated and quantified by using the criteria (4.9).

Using the streamwise velocity  $u$ , the  $\text{IODMD}_u$  algorithm yields the results summarized in Table 4.1:

Table 4.1: Models validation results using  $\text{IODMD}_u$  algorithm

Model properties		Model obtained by IODMD			
Observable	VAF(WT2) <sub>max</sub>	Model size	VAF(WT1)  <sub>r</sub>	NRMSE (id)	NRMSE (val)
$u$	88.40 %	37	99.12 %	2.59%	4.00%

The above tables has 6 columns: the first corresponds to the measurement used to build the data matrices, and therefore the high order state  $\mathbf{x}$ . The second is the highest VAF for turbine 2 generator power (the second output of the system) of all models. **The dynamics of the power produced at turbine 2 are more difficult to model when compared to turbine 1, therefore this measure serves to decide the model to choose.** The third column has the size of the model - the number of states. The fourth is the VAF of turbine 1 power given the chosen model. The fifth and and sixth columns are the average NRMSE throughout the linear simulation. The fifth represents the ability of the model to interpolate data (given the identification data set) and the sixth the ability of the model to predict the wake behaviour (validation data).

The graph in Figure 4.12 depicts the VAF of each one of the models computed, each one having one more state (also referred to as mode). It is clear that the dynamics of turbine 1 are easier to model, as turbine 2 requires more modes.

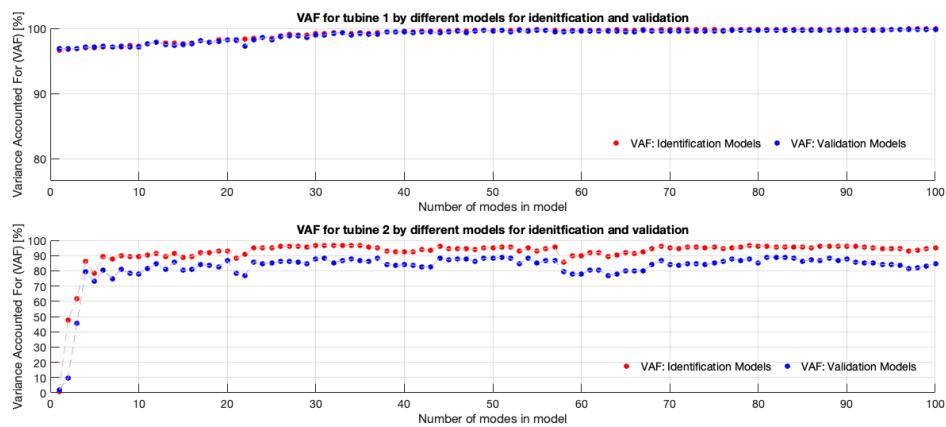


Figure 4.12: Fitness of models. Each model, from left to right, has increasing number of states. In red, output performance using identification data and in blue using validation data.

<sup>8</sup>Interpreting the state an any time instant as a point in a  $n$ -dimensional state space, the dynamic state response  $\mathbf{x}(t)$  can be interpreted as a path or trajectory traced out in state space.

The performance of the best model, whose parameters are described in Table 4.1, can be seen in Figure 4.13, where the green dashed line - the model's outputs - are shown to satisfactorily capture the real dynamics.

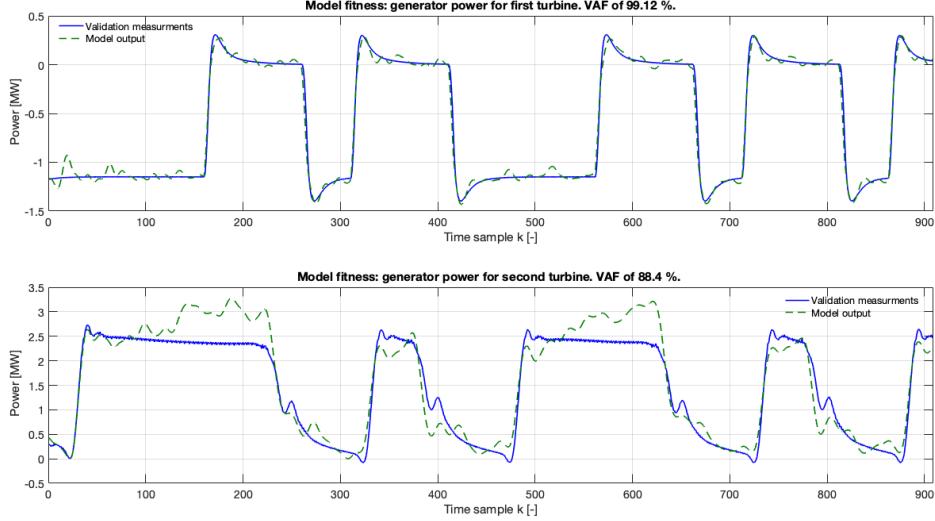


Figure 4.13: Best performing model validation results. In blue, real generated power in turbines 1 and 2 from SOWFA. In green, response from ROM.

In Figure 4.14, the model's capability to reproduce the wake's behaviour is shown, by comparing the state trajectory to the real flow evolution from the simulation at hub height. On the left, the velocity  $u$  is reconstructed based on the low order state trajectory at key time instants when the wake is deflected. On the right, simulation data at the same time instants is shown.

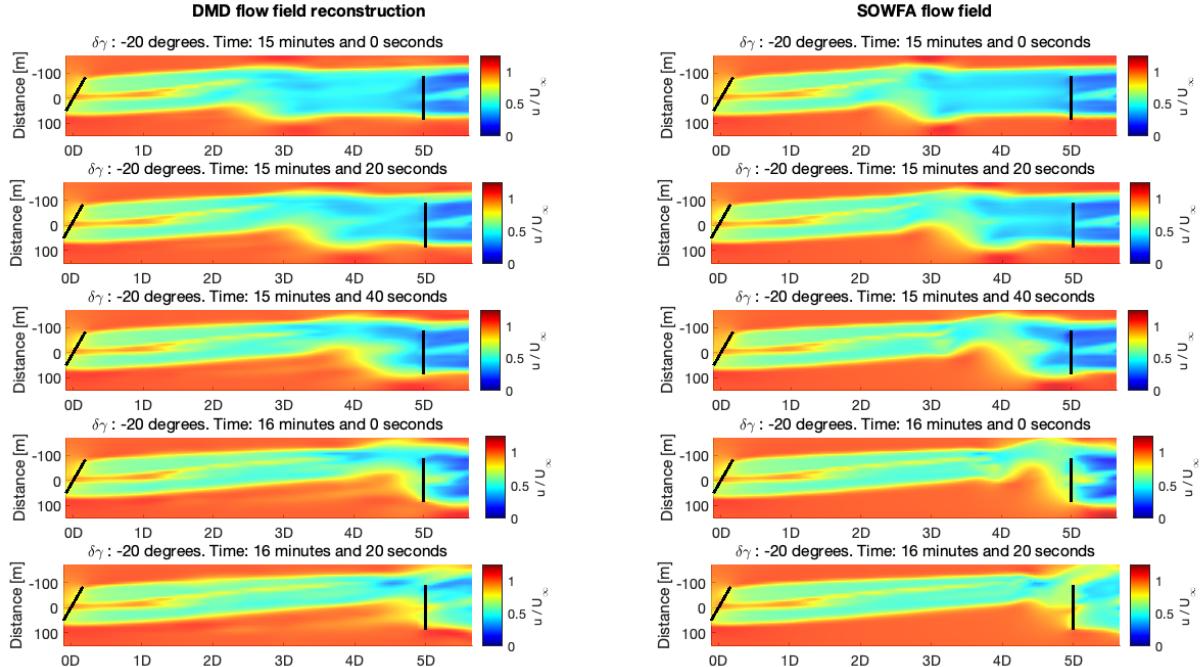


Figure 4.14: Comparison of flow field reconstruction from low order state trajectory (left) and data from simulation (right). State trajectory obtained using validation data. Key snapshots in time are taken to illustrate the ROM's ability in predicting wake dynamics.

Figures 4.15 and 4.16 further illustrate the model's reconstruction capabilities - for validation data - at different time intervals when the altered wake hits the downstream rotor. Deviations are higher for the instants when the wake at the downstream rotor suffers increased alterations, as portrayed in Figure 4.15. As the wake fully develops, deviations decrease, as illustrated in the final snapshots in Figure 4.16.

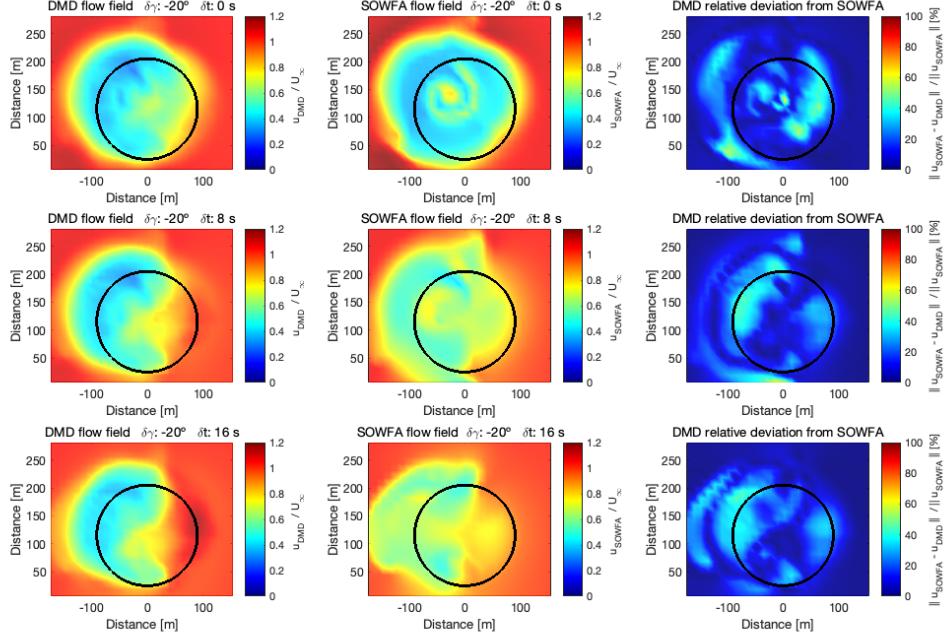


Figure 4.15: Comparison of flow field reconstruction using reduced linear model (left) and data from SOWFA (middle) at downstream turbine rotor plane at different instances (during a total of 16 seconds). IODMD<sub>*u*</sub> flow field deviation with relation to SOWFA data is shown on the right.

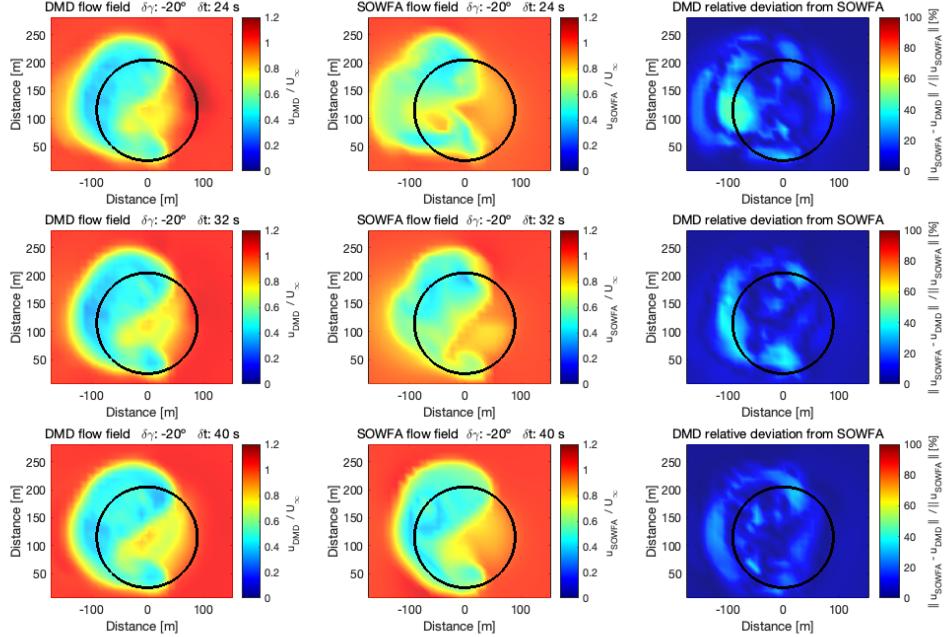


Figure 4.16: Comparison of flow field reconstruction using reduced linear model (left) and data from SOWFA (middle) at downstream turbine rotor plane at different instances (additional 16 seconds). IODMD<sub>*u*</sub> flow field deviation with relation to SOWFA data is shown on the right.

**Figure 4.14 puts in evidence the ability of a linear state space reduced order model with 37 states to reproduce the non linear dynamics of a wind turbine wake with tens of thousands of states.** Minor differences are visible in the each snapshots. Taken together, for every time instant of the simulation, such differences represent an averaged NRMSE of 4%, given that validation data is used.

The dynamical properties of the ROM are summarized in Table 4.2. These properties are dictated by the eigenvalues of the state matrix  $\mathbf{A}$ , which are referred to as poles of the system. The poles lay in the complex plane, presenting both real and imaginary part. The poles are converted from discrete time to continuous time by using the transformation  $s = \ln(z)/T_s$ , where  $s$  is the continuous time pole,  $z$  is the discrete time pole and  $T_s$  is the sampling time, in seconds. The natural frequencies<sup>9</sup>  $\omega_n$  are computed by taking the modulus of each pole in continuous time and are then converted from rad/s to Hz. The damping ratio  $\xi$ , a dimensionless ratio which translates how oscillations of the mode decay after a disturbance, are computed by  $-\cos(\angle \ln(z))$ , or  $-\cos(\text{atan2}(\text{Im}(\ln(z)), \text{Re}(\ln(z))))$ .

Table 4.2: Dynamical properties of IODMD<sub>u</sub> model. First column entries correspond to the numerology chosen for the modes, which are ordered by increasing natural frequency. Second column entries correspond to their order in terms of eigenvectors of state matrix  $\mathbf{A}$ . Third and fourth columns have information of the modes natural frequencies and fifth column the damping ratio

Modes		Dynamical properties		
#	DMD	$\omega_n$ [Hz]	$\omega_n$ St [-]	$\xi$
1	27	0	0	1
2	32	0.0021	0.0415	1
3	37	0.0022	0.0440	1
4	30-31	0.0024	0.0466	0.6250
5	33-34	0.0048	0.0943	0.3646
6	35-36	0.0056	0.1103	0.1336
7	28-29	0.0084	0.1670	0.1771
8	25-26	0.0105	0.2072	0.0103
9	23-24	0.0133	0.2635	0.0940
10	21-22	0.0174	0.3449	0.0428
11	19-20	0.0192	0.3805	0.0205
12	17-18	0.0229	0.4532	0.0319
13	15-16	0.0254	0.5033	0.0239
14	13-14	0.0300	0.5934	0.0040
15	11-12	0.0341	0.6748	0.0032
16	9-10	0.0392	0.7752	0.0062
17	7-8	0.0433	0.8560	0.0127
18	5-6	0.0475	0.9386	0.0218
19	3-4	0.0498	0.9845	0.0082
20	1-2	0.1461	2.8904	0.0054

The location of the poles corresponding to each of the modes can be visualised in Figure 4.17, where they have been labelled according to the numerology in Table 4.2. In discrete time, all poles lay within the

<sup>9</sup>Frequency at which a system tends to oscillate in the absence of any driving or damping force.

unit circle, identified with a dashed line, indicating that the system is stable (no regularization is needed).

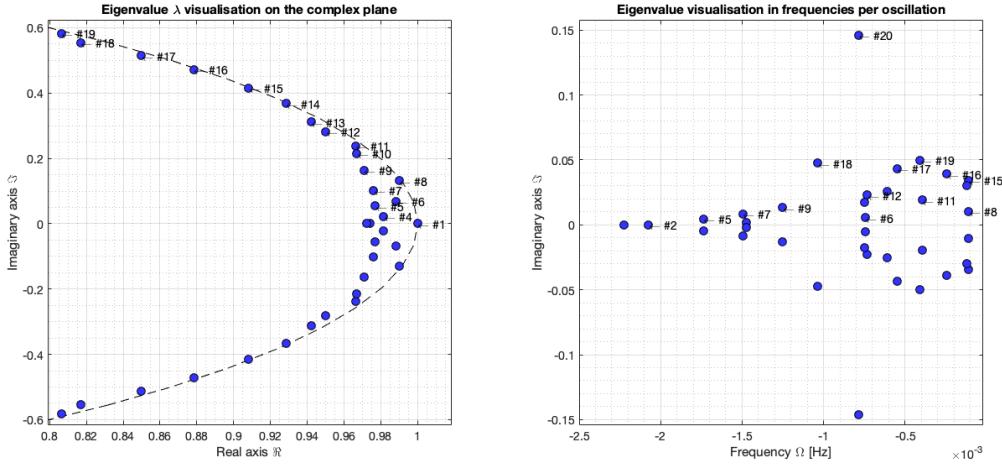


Figure 4.17: DMD eigenvalues of the  $\text{IODMD}_u$  for wake redirection control by yaw misalignment. On the left, the discrete time eigenvalues  $\lambda$  on the complex plane are represented relative to the unit circle (dashed line). The area where the majority of modes are presented has been zoomed in. On the right, the eigenvalues transformed to continuous time are represented.

The model can also be analysed in terms of its frequency response, this is, the steady-state response of a system to a sinusoidal input, fully described in [55]. The Bode diagram provides a graphical framework to present the frequency response characteristics of a system. In Figure 4.18, the standard representation of the logarithmic magnitude of the system, for both outputs, is represented. The logarithmic magnitude is computed by taking the logarithm of base 10 of the amplitude ratio of the output sinusoid to the input sinusoid  $20 \log_{10} \left| \frac{Y(j\omega)}{X(j\omega)} \right|$ , where  $\omega$  is the frequency in rad/s. Near the frequency  $\omega_n$ , a resonance peak occurs, and the damping ratio  $\xi$  determines the magnitude of this resonance peak, as can be observed by cross referencing information in Table 4.2 and Figure 4.18. The Nyquist frequency, defined as half of the sampling rate, is marked as a vertical line.

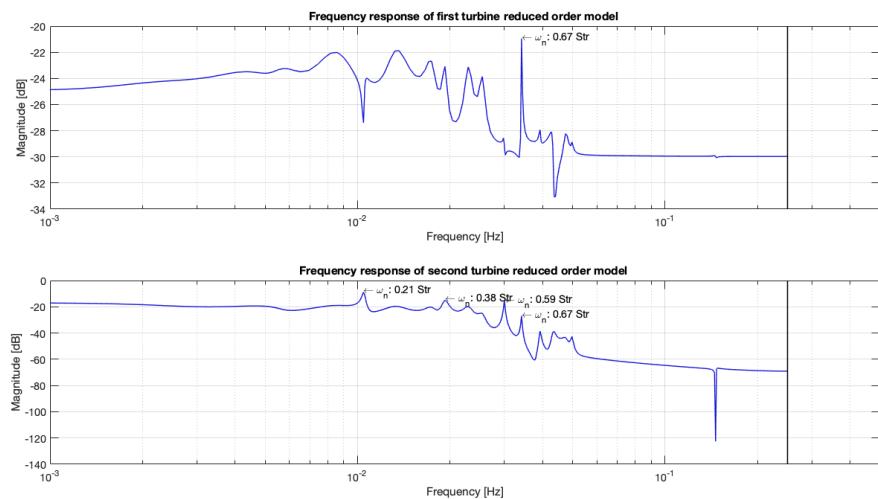


Figure 4.18: Bode magnitude plot of  $\text{IODMD}_u$  model for yaw redirection control. A number of resonance peaks have been identified, where their frequency, in Strouhal number, is depicted.

Each of the eigenvalues in Figure 4.17 corresponds to an eigenvector of the state matrix  $\mathbf{A}$ , which, in turn, **corresponds to a mode or coherent structure in space which inherits the dynamical properties** indicated in Table 4.2. Figures 4.19, 4.20 and 4.21 allow to visualise these structures in different frequency domains. Isosurfaces of different values have been chosen so that the structure can be visualised, which are indicated in the legend. The modes, excited by certain inputs, will contribute to the overall evolution of the flow field and thus model the wake.

It should be noted that these modes are merely representations of the high order eigenvectors, and are not representative of the influence of the control action. Nevertheless, their study **supports a deeper understanding of the wake dynamical properties**, analogous to understanding the behaviour of a beam by decomposing it into different modal contributions.

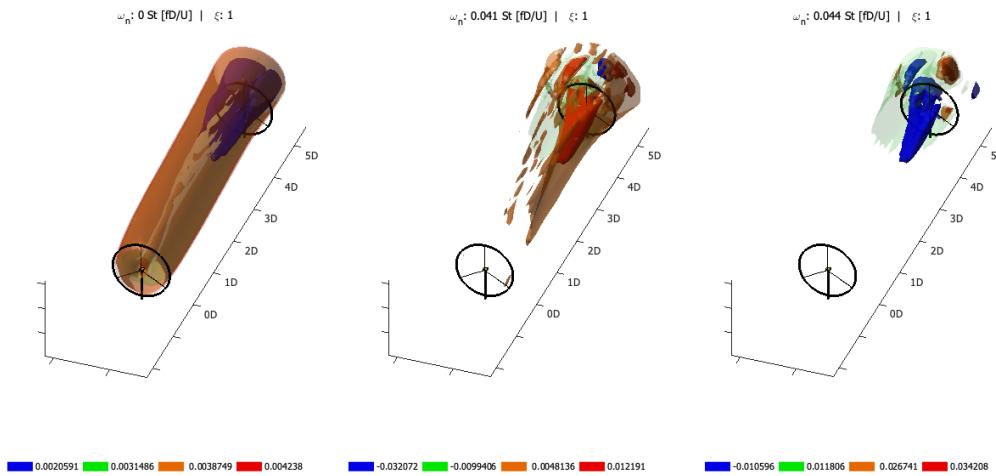


Figure 4.19: DMD modes #1, #2 and #3. Mode #1 corresponds to a background mode that is not changing (i.e, it has zero eigenvalue). Modes #2 and #3 correspond to eigenvalues laying on the real axis.

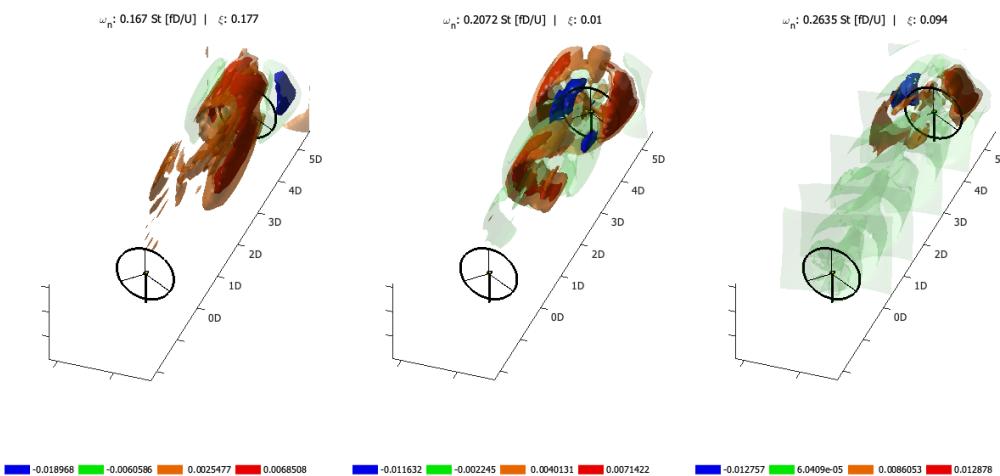


Figure 4.20: DMD modes #7, #8 and #9, with natural frequencies comprised between 0.17 St and 0.26 St. The spatial patterns presented here appear to explain some of the dynamics more related with the downstream turbine. The structures seen here share same similarities with the shapes in Figure 4.14 as the wake is deflected near the downstream rotor.

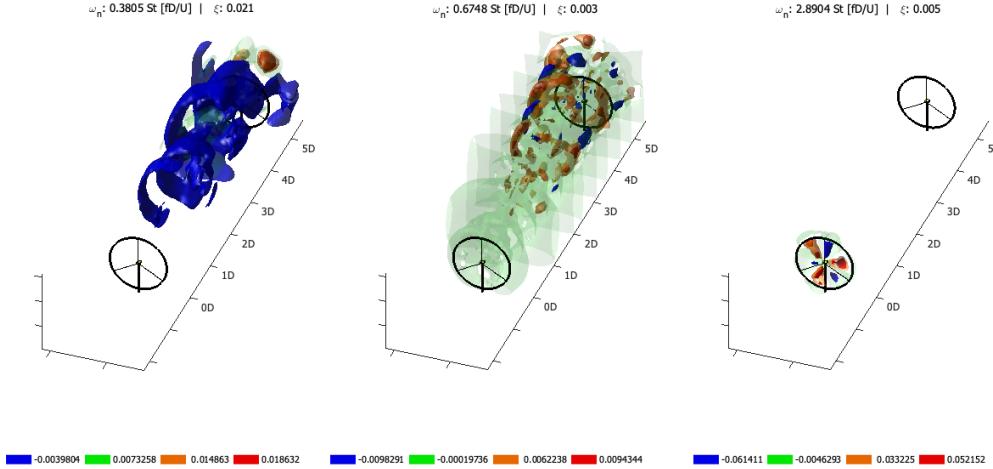


Figure 4.21: DMD modes #11, #15 and #20, with natural frequencies comprised between  $0.38 St$  and  $2.89 St$ . The highest frequency mode appears to be somewhat related with fluid phenomena at the blades/rotor domain.

The procedure of building IODMD models where the velocity component  $u$  is used has been attempted for other control strategies (AIC), as described in [4]. One of the objectives of this work is to **leverage the ideas behind the Koopman operator in order to find coordinates where the non linear dynamics of the wake can better be described. Having the assumption in mind that other variables of interest may exist, several candidates are tested**. The results are summarized in Table 4.3.

Table 4.3: Models validation results using IODMD algorithm with different observables. First column has the data chosen for the states. Second, third and forth columns have the fit criteria VAF and model size, where VAF of the second turbine generator power is the most relevant. The last two columns show the improved performance in terms of VAF and the improvements in terms of the size of model. Reduced models are less computationally expensive for the purposes of model predictive control, hence they are written in green.

Model properties		Model obtained by IODMD			$\Delta$ IODMD <sub><math>u</math></sub>	
Observable	VAF(WT2) <sub>max</sub>	Model size	VAF(WT1)  <sub><math>r</math></sub>	$\Delta$ VAF (WT2) <sub>max</sub>	$\Delta$ size	
$v$	87.12 %	40	99.39 %	-1.28%	+8.11%	
$w$	87.54 %	50	99.35 %	-0.86%	+35.14%	
$u'$	78.14 %	38	98.08 %	-10.26%	+2.70%	
$v'$	61.30 %	31	97.13 %	-27.10%	-16.22%	
$u^2$	89.63 %	26	98.83 %	+1.23%	-29.73%	
$v^2$	89.58 %	12	97.52 %	+1.18%	-67.57%	
$w^2$	87.85 %	47	97.67 %	-0.55%	+27.03%	
$u \cdot v$	69.40 %	15	97.58 %	-19.00%	-59.46%	
$u^2 + v^2$	89.42 %	26	98.84 %	+1.02%	-29.73%	
$u^2 + v^2 + w^2$	89.36 %	49	99.75%	+0.96%	+32.43%	

The attempt to improve the model's ability in recovering the dynamics by changing the information in the snapshot matrices appears to be fruitful. **A VAF of 90% can be obtained with a linear state space model with 26 states by using the non linear observable  $u^2$** , which represents an increase of 1.23% in VAF and decrease in state vector size of 30% when compared to the baseline model derived with  $\text{IODMD}_u$ . Although the increases in VAF do not appear to be substantial, this procedure yielded an increase in VAF of 8% for another wind farm control strategy studied, AIC, where the results are summarized in Annex A, Table A.3.

The frequency responses of  $\text{IODMD}_u$ ,  $\text{IODMD}_{v^2}$  and  $\text{IODMD}_{u^2}$  are depicted in Figure 4.22. It is shown that **independently of the observable used, the three models show similarities in terms of their frequency response**, presenting resonance frequencies near the 0.21 St. This range of frequencies also appears for the models studied in AIC, which is visible in Annex A, Figure A.18.

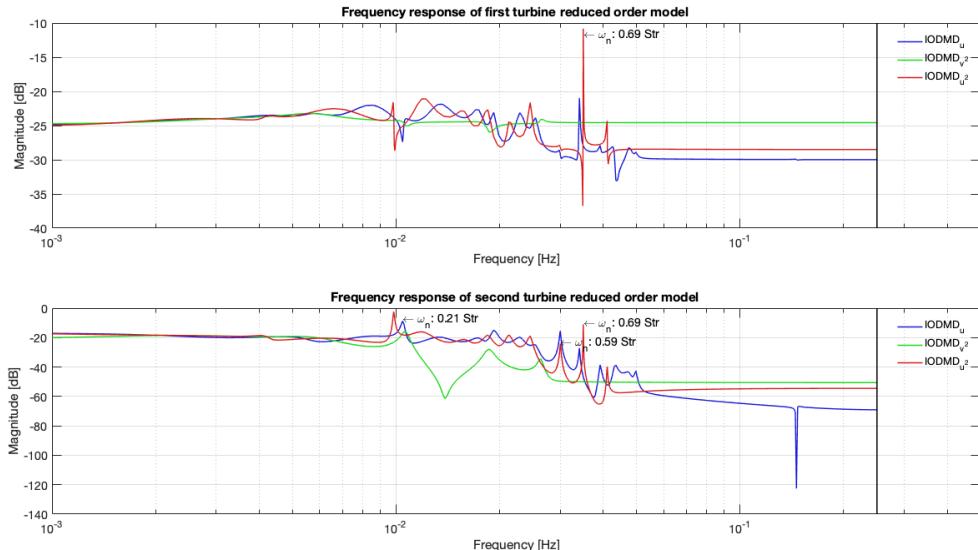


Figure 4.22: Bode magnitude plot of IODMD models using different observables.

Another pattern of interest are the **alike frequencies identified as relevant by IODMD, regardless of observable chosen or wind farm control strategy being studied**. Comparing the frequencies of the model for the case of yaw control in Table 4.2 and the frequencies of the model for the case of collective pitch control in Table A.2, coherent structures in the vicinity of frequencies such as 0.2, 0.59 and 0.67 St are visible, along with the resonance peaks in the respective frequency responses.

#### 4.1.8 Step Eight: Final Model to Use

The model chosen to be used for the design of the predictive controller is the  $\text{IODMD}_u^2$ , which yielded the best VAF for the generator power at the downstream turbine. The results obtained in this section by making use of the first pathway mentioned in Subsection 4.1.6 were satisfactory. The two remaining techniques are further explored to model AIC, in Appendix A.

In the final implementation, the actuator (the yaw motor) is modelled as a low pass filter<sup>10</sup> and is

<sup>10</sup>A low-pass filter is a filter which passes low-frequency signals and blocks, or impedes, high-frequency signals. Linear filters can be considered as linear dynamical systems, the behaviour of which can be described by ordinary differential equations.

connected in series with the IODMD $_u^2$  model. By doing this, the slow dynamics of the actuator are taken into account and the state space feedthrough matrix entries  $\mathbf{D}$  are reduced to zero, which facilitates the design of the predictive controller, as the system becomes strictly causal<sup>11</sup>.

In this chapter, the data driven techniques appropriate for the modelling of high dimensional systems for control are used. It is shown that by using IODMD $_u$  a reduced order state space model with 37 states can reproduce the downstream turbine generator power dynamics with a VAF of 88% and simultaneously reconstruct the upstream turbine wake with an average NRMSE of 4%. Other coordinates to describe the system's dynamics are further explored, and it is found that using  $u^2, v^2$  or  $u^2 + v^2$  leads to increased benefits in terms of VAF of the downstream turbine generator power in the order of 1.02% to 1.23%.

In the next chapter, the IODMD $_u^2$  is used to design a model predictive controller which is implemented in SOWFA so that a pre specified collective amount of power can be produced.

---

<sup>11</sup>A system is said to be causal if it is non-anticipatory, i.e., it cannot respond to inputs that will be applied in the future, but only on past and present inputs. A strictly causal system only depends on past inputs.

# CHAPTER 5

## MODEL PREDICTIVE CONTROL

---

### 5.1 Linear Model Predictive Control Overview

Model Predictive Control (MPC), also referred to as Model Based Predictive Control (MBPC), is an advanced control strategy that computes the optimal control action, at each time instant, to be applied to a system, given a certain objective. The computation of the optimal control action relies on dynamical model of the process, most often linear empirical models obtained by systems identification.

MPC has been in use in the process industries such as chemical plants and oil refineries since the 1980s, and according to [13] its inherent attributes have made this control strategy a very successful one in the context of industrial control design: **(1) simplicity** - the underlying idea is intuitive and does not require complex mathematics, **(2) richness** - the primary MPC components can be tailored to the details of the problem in hand, **(3) practicality** - allows to solve problems bounded by input or output constraints and is easily adaptable to multivariable processes **(4) demonstrability** - it has been proven to work in industrial contexts and is routinely and profitably employed. Its several advantages and wide applicability provide strong motivations for its usage in the context of wind farm power optimisation.

Many real processes are not linear. As it has been indicated, the behaviour of a fluid evolving behind a wind turbine is strongly non linear. Nevertheless, it can often be considered to be approximately linear over a small operating range and the Koopman operator theory provides mathematical evidence proving that DMD can be used as a tool to model the existing non-linearities. **As the prospect of capturing the non linear dynamics with a simple linear model is high, the linear MPC strategy is chosen, permitting to keep the computational cost low.**

MPC is based on iterative, finite horizon optimisation of a plant model. At time  $t$ , the current plant state is sampled and a cost-function is minimised for a time horizon in the future  $t + H_p$ , where  $H_p$  is usually referred to as prediction horizon. The minimisation task yields the optimal control actions over a predefined control horizon  $H_c$ , and only the first step is implemented. For the purposes of this work, the prediction and control horizons are kept the same.

MPC is rather a methodology than a single technique, as argued in [67]. The difference in the various methods is mainly the way the problem is translated into a mathematical formulation, so that it becomes solvable in the limited time interval available. Despite the method chosen, five important items

are part of the design procedure: **(1) process model** - the model has been previously chosen to be the state space model, which will then be used to make an estimation of the future behaviour of the system, namely state and output predictions, **(2) performance index** - a cost function which quantifies how close a certain desired reference is being tracked, **(3) constraints** - imposed boundaries which the control action must not violate, due to operational or physical limitations, **(4) optimisation** - an optimisation algorithm is applied to compute the sequence of future control signals that minimises the performance index subject to the given constraints, **(5) receding horizon principle** - implementation of only the first control sample. The choosing and explanation of these five items within the context of this thesis is given below.

For simplicity purposes, the notation distinguishing the low order and high order representations of the state space models is dropped in the following subsections, and all state space matrices refer to the low order representation, this is, the ROM.

## 5.2 Model Predictive Control Tracking Problem Formulation

### 5.2.1 Process Model

An important difference between MPC and other control methods is the explicit use of a model. The great advantage is that the behaviour of the controller can be studied in great detail and possible failures can be well detected. On the other hand, a detailed study of the plant behaviour has to be performed, which leads to about 80% of the work being focused on modelling and identification of the plant, as stated in [67]. It is therefore crucial that a reliable model which accurately relates inputs and outputs is firstly investigated. Two types of MPC models, based on the state space description, are used:

- **Direct Input Output (DIO) Models:** the input signal is directly applied to the models
- **Increment Input Output (IIO) Models:** the increments of the input signal are applied to the models instead of the input directly.

#### 5.2.1.1 Direct Inputs Outputs Model

The linear prediction model can be written based on the state space equations, knowing that the future state is a function of the present state  $x_0$ , the current and future control actions and the dynamics embedded in the **A** and **B** matrices:

$$\begin{aligned} x_1 &= \mathbf{A}x_0 + \mathbf{B}u_0 \\ y_0 &= \mathbf{C}x_0 + \mathbf{D}u_0 \end{aligned} \tag{5.1}$$

$$\begin{aligned} x_2 &= \mathbf{A}x_1 + \mathbf{B}u_1 = \mathbf{A}(\mathbf{A}x_0 + \mathbf{B}u_0) + \mathbf{B}u_1 = \mathbf{A}^2x_0 + \mathbf{AB}u_0 + \mathbf{Bu}_1 \\ y_1 &= \mathbf{Cx}_1 + \mathbf{Du}_1 = \mathbf{C}(\mathbf{Ax}_0 + \mathbf{Bu}_0) + \mathbf{Du}_1 = \mathbf{CA}x_0 + \mathbf{CB}u_0 + \mathbf{Du}_1 \end{aligned} \tag{5.2}$$

$$\begin{aligned} \mathbf{x}_3 &= \mathbf{Ax}_2 + \mathbf{Bu}_2 = \mathbf{A}(\mathbf{A}^2\mathbf{x}_0 + \mathbf{AB}\mathbf{u}_0 + \mathbf{B}\mathbf{u}_1) + \mathbf{Bu}_2 = \mathbf{A}^3\mathbf{x}_0 + \mathbf{A}^2\mathbf{Bu}_0 + \mathbf{AB}\mathbf{u}_1 + \mathbf{Bu}_2 \\ \mathbf{y}_2 &= \mathbf{Cx}_2 + \mathbf{Du}_2 = \mathbf{C}(\mathbf{A}^2\mathbf{x}_0 + \mathbf{AB}\mathbf{u}_0 + \mathbf{B}\mathbf{u}_1) + \mathbf{Du}_2 = \mathbf{CA}^2\mathbf{x}_0 + \mathbf{CAB}\mathbf{u}_0 + \mathbf{CB}\mathbf{u}_1 + \mathbf{Du}_2 \end{aligned} \quad (5.3)$$

The predicted measurements may be written in a compact form and generalised for a prediction horizon  $H_p$ . Designating the output vector  $\mathbf{y}$  by  $\mathbf{P}$ , corresponding to the generator power variations on the upstream and downstream turbines:

$$\begin{bmatrix} \mathbf{P}_k \\ \mathbf{P}_{k+1} \\ \mathbf{P}_{k+2} \\ \vdots \\ \mathbf{P}_{k+H_p} \end{bmatrix} = \begin{bmatrix} \mathbf{D} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CB} & \mathbf{D} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CAB} & \mathbf{CB} & \mathbf{D} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{H_p-1}\mathbf{B} & \mathbf{CA}^{H_p-2}\mathbf{B} & \mathbf{CA}^{H_p-3}\mathbf{B} & \cdots & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{u}_k \\ \mathbf{u}_{k+1} \\ \mathbf{u}_{k+2} \\ \vdots \\ \mathbf{u}_{k+H_p} \end{bmatrix} + \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{H_p} \end{bmatrix} \mathbf{x}_k \quad (5.4)$$

If the direct feed through matrix is assumed to be zero, the prediction problem simplifies to:

$$\begin{bmatrix} \mathbf{P}_{k+1} \\ \mathbf{P}_{k+2} \\ \mathbf{P}_{k+3} \\ \vdots \\ \mathbf{P}_{k+H_p} \end{bmatrix} = \begin{bmatrix} \mathbf{CB} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CAB} & \mathbf{CB} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CA}^2\mathbf{B} & \mathbf{CAB} & \mathbf{CB} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{H_p-1}\mathbf{B} & \mathbf{CA}^{H_p-2}\mathbf{B} & \mathbf{CA}^{H_p-3}\mathbf{B} & \cdots & \mathbf{CB} \end{bmatrix} \begin{bmatrix} \mathbf{u}_k \\ \mathbf{u}_{k+1} \\ \mathbf{u}_{k+2} \\ \vdots \\ \mathbf{u}_{k+H_p-1} \end{bmatrix} + \begin{bmatrix} \mathbf{CA} \\ \mathbf{CA}^2 \\ \mathbf{CA}^3 \\ \vdots \\ \mathbf{CA}^{H_p} \end{bmatrix} \mathbf{x}_k \quad (5.5)$$

which can be rewritten in a more compact form:

$$\bar{\mathbf{P}} = \bar{\mathbf{H}}\bar{\mathbf{U}} + \Gamma\mathbf{x}_k \quad (5.6)$$

where  $\bar{\mathbf{P}} \in \mathbb{R}^{H_p \cdot l}$ ,  $\bar{\mathbf{H}} \in \mathbb{R}^{H_p \cdot l \times H_c \cdot q}$ ,  $\bar{\mathbf{U}} \in \mathbb{R}^{H_c \cdot q}$ ,  $\Gamma \in \mathbb{R}^{H_p \cdot l \times r}$ . The number of inputs and outputs is, respectively,  $q$  and  $l$ .

The matrices are defined as follows:

$$\bar{\mathbf{P}} = \begin{bmatrix} \mathbf{P}_{k+1} \\ \mathbf{P}_{k+2} \\ \vdots \\ \mathbf{P}_{k+H_p} \end{bmatrix} \quad \bar{\mathbf{H}} = \begin{bmatrix} \mathbf{CB} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CAB} & \mathbf{CB} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{H_p-1}\mathbf{B} & \mathbf{CA}^{H_p-2}\mathbf{B} & \cdots & \mathbf{CB} \end{bmatrix} \quad \bar{\mathbf{U}} = \begin{bmatrix} \mathbf{u}_k \\ \mathbf{u}_{k+1} \\ \vdots \\ \mathbf{u}_{k+H_p-1} \end{bmatrix} \quad \Gamma = \begin{bmatrix} \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{H_p} \end{bmatrix} \quad (5.7)$$

### 5.2.1.2 Increment Inputs Outputs Model

In the IIO formulation, input increments  $\Delta u_k$  defined as  $u_k - u_{k-1}$  are considered. The reasoning behind this approach is based on the **improved steady state behaviour of the controller**, as proved in [67]. The process outputs are then simulated based on input increments, and the standard state space model

can be rewritten in terms of increment variables:

$$\begin{aligned}\Delta \mathbf{x}_{k+1} &= \mathbf{A} \Delta \mathbf{x}_k + \mathbf{B} \Delta \mathbf{u}_k \\ \Delta \mathbf{y}_k &= \mathbf{C} \Delta \mathbf{x}_k\end{aligned}\quad (5.8)$$

Taking the model output  $\mathbf{y}_k$  as  $\mathbf{y}_{k-1} + \mathbf{C} \Delta \mathbf{x}$ , the new state space can be formulated:

$$\begin{aligned}\begin{bmatrix} \mathbf{y}_k \\ \Delta \mathbf{x}_{k+1} \end{bmatrix} &= \begin{bmatrix} \mathbf{I} & \mathbf{C} \\ \mathbf{0} & \mathbf{A} \end{bmatrix} \begin{bmatrix} \mathbf{y}_{k-1} \\ \Delta \mathbf{x}_k \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{B} \end{bmatrix} \Delta \mathbf{u}_k \\ \mathbf{y}_k &= \begin{bmatrix} \mathbf{I} & \mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{y}_{k-1} \\ \Delta \mathbf{x}_k \end{bmatrix}\end{aligned}\quad (5.9)$$

which corresponds to the extended state space:

$$\begin{aligned}\mathbf{x}_{k+1}^e &= \mathbf{A}_e \mathbf{x}^e + \mathbf{B}_e \Delta \mathbf{u}_k \\ \mathbf{y}_k &= \mathbf{C}_e \mathbf{x}^e\end{aligned}\quad (5.10)$$

where the new extended state is now:

$$\mathbf{x}_k^e = \begin{bmatrix} \mathbf{y}_{k-1} \\ \Delta \mathbf{x}_k \end{bmatrix}\quad (5.11)$$

Analogous to the direct IO MPC, the future predictions for the IIO MPC can be written in a compact form, leading to:

$$\begin{bmatrix} \mathbf{P}_{k+1} \\ \mathbf{P}_{k+2} \\ \vdots \\ \mathbf{P}_{k+H_p} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_e \mathbf{B}_e & 0 & \cdots & 0 \\ \mathbf{C}_e \mathbf{A}_e \mathbf{B}_e & \mathbf{C}_e \mathbf{B}_e & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_e \mathbf{A}_e^{H_p-1} \mathbf{B}_e & \mathbf{C}_e \mathbf{A}_e^{H_p-2} \mathbf{B}_e & \cdots & \mathbf{C}_e \mathbf{B}_e \end{bmatrix} \begin{bmatrix} \Delta \mathbf{u}_k \\ \Delta \mathbf{u}_{k+1} \\ \vdots \\ \Delta \mathbf{u}_{k+H_p-1} \end{bmatrix} + \begin{bmatrix} \mathbf{C}_e \mathbf{A}_e \\ \mathbf{C}_e \mathbf{A}_e^2 \\ \vdots \\ \mathbf{C}_e \mathbf{A}_e^{H_p} \end{bmatrix} \mathbf{x}_k^e\quad (5.12)$$

which can be rewritten in a more compact form:

$$\bar{\mathbf{P}} = \bar{\mathbf{H}}_e \Delta \bar{\mathbf{U}} + \Gamma_e \mathbf{x}_k^e\quad (5.13)$$

## 5.2.2 Performance Index

The MPC is used to solve a tracking problem, meaning that an optimal control law forces the plant to track a desired reference trajectory  $\mathbf{P}_{ref}(t)$  over a specified time interval  $[t_o, T]$ . **The reference corresponds to the total wind farm power generated by both turbines.** The total wind farm power prediction at each time step in the future, designated by vector  $\hat{\mathbf{P}}$ , is computed by summing the power prediction of the individual turbines, resorting to matrix  $\mathbf{F}$ :

$$\hat{\mathbf{P}} = \mathbf{F} \bar{\mathbf{P}} : \quad (5.14)$$

$$\hat{\mathbf{P}} = \begin{bmatrix} 1 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{P}_{k+1} \\ \mathbf{P}_{k+2} \\ \vdots \\ \mathbf{P}_{k+H_p} \end{bmatrix} \quad (5.15)$$

A cost function  $J$  can be written so as to formulate the tracking problem to achieve a certain value of collective power  $\mathbf{P}_{ref}$ . Making use of the IIO model due to its advantages with relation to the DIO, the quadratic cost function to be minimised is then:

$$J = (\hat{\mathbf{P}} - \mathbf{P}_{ref})^T \mathbf{Q} (\hat{\mathbf{P}} - \mathbf{P}_{ref}) + \Delta \bar{\mathbf{U}}^T \mathbf{R} \Delta \bar{\mathbf{U}} \quad (5.16)$$

where  $\mathbf{Q} \in \mathbb{R}^{(H_p \cdot l) \times (H_p \cdot l)}$  and  $\mathbf{R} \in \mathbb{R}^{(H_c \cdot q) \times (H_c \cdot q)}$  are relative weights of reference deviation and input usage, respectively.

Inserting the definition (5.14) in the cost function  $J$ :

$$J = (\mathbf{F}\bar{\mathbf{P}} - \mathbf{P}_{ref})^T \mathbf{Q} (\mathbf{F}\bar{\mathbf{P}} - \mathbf{P}_{ref}) + \Delta \bar{\mathbf{U}}^T \mathbf{R} \Delta \bar{\mathbf{U}} \quad (5.17)$$

Substituting (5.13) in (5.17), the following expression for the cost function becomes (along with the optimisation problem in hand):

$$\min_{\begin{bmatrix} u_{k+1} & \cdots & u_{k+H_p} \end{bmatrix}} J = (\mathbf{F}\bar{\mathbf{H}}_e \Delta \bar{\mathbf{U}} + \mathbf{F}\Gamma_e \mathbf{x}_k^e - \mathbf{P}_{ref})^T \mathbf{Q} (\mathbf{F}\bar{\mathbf{H}}_e \Delta \bar{\mathbf{U}} + \mathbf{F}\Gamma_e \mathbf{x}_k^e - \mathbf{P}_{ref}) + \Delta \bar{\mathbf{U}}^T \mathbf{R} \Delta \bar{\mathbf{U}} \quad (5.18)$$

The cost function can be further simplified, by observing that a constant term appears  $\mathcal{C} = \mathbf{F}\Gamma_e \mathbf{x}_k^e - \mathbf{P}_{ref}$ :

$$J = (\mathbf{F}\bar{\mathbf{H}}_e \Delta \bar{\mathbf{U}} + \mathcal{C})^T \mathbf{Q} (\mathbf{F}\bar{\mathbf{H}}_e \Delta \bar{\mathbf{U}} + \mathcal{C}) + \Delta \bar{\mathbf{U}}^T \mathbf{R} \Delta \bar{\mathbf{U}} \quad (5.19)$$

Making use of the mathematical properties of matrices, the expression in (5.19) is further simplified:

$$J = \Delta \bar{\mathbf{U}}^T \bar{\mathbf{H}}_e^T \mathbf{F}^T \mathbf{Q} \mathbf{F} \bar{\mathbf{H}}_e \Delta \bar{\mathbf{U}} + \Delta \bar{\mathbf{U}}^T \bar{\mathbf{H}}_e^T \mathbf{F}^T \mathbf{Q} \mathcal{C} + \mathcal{C}^T \mathbf{Q} \mathbf{F} \bar{\mathbf{H}}_e \Delta \bar{\mathbf{U}} + \mathcal{C}^T \mathbf{Q} \mathcal{C} + \Delta \bar{\mathbf{U}}^T \mathbf{R} \Delta \bar{\mathbf{U}} \quad (5.20)$$

For the purpose of MPC, the cost function is rewritten in the following fashion, facilitating the formulation of the optimisation step:

$$J = \Delta \bar{\mathbf{U}}^T (\bar{\mathbf{H}}_e^T \mathbf{F}^T \mathbf{Q} \mathbf{F} \bar{\mathbf{H}}_e + \mathbf{R}) \Delta \bar{\mathbf{U}} + 2 \Delta \bar{\mathbf{U}}^T \bar{\mathbf{H}}_e^T \mathbf{F}^T \mathbf{Q} \mathcal{C} + \mathcal{C}^T \mathbf{Q} \mathcal{C} \quad (5.21)$$

### 5.2.3 Constraints

Inequality constraints exist due to physical limitations. In the context of wind farms, the turbine's control variables should be maintained within a certain interval. **The yaw angle is restricted to values between 0 and -35 degrees.** This restriction is set so that the operational DOF matches the ones used to identify the model. The decision of setting the lower and upper limits to the chosen values is to evaluate the

performance of the model outside the operating regime chosen to perform systems identification.

Constraints on the input level are used, and the minimisation of the cost functions is then subject to:

$$u_{min} \leq u_k \leq u_{max}, \quad k = 1, \dots, H_p - 1 \quad (5.22)$$

The formulation in (5.22) states that the optimal control action at each time instant within the control horizon  $H_c$  is bounded between  $u_{min}$  and  $u_{max}$ . The MPC formulation, however, computes the optimal incremental control action  $\Delta u$ , therefore restriction have to be written in terms of incremental actions. This is addressed in the next subsection 5.2.4.

## 5.2.4 Optimisation

The minimisation of the cost function  $J$  in (5.21) subject to the restrictions on the yaw angle in (5.22) can be written as a Quadratic Programming (QP)<sup>1</sup> problem, of the type:

$$\Delta \bar{\mathbf{U}}^* = \min_{\Delta \bar{\mathbf{U}}} \left\{ \frac{1}{2} \Delta \bar{\mathbf{U}}^T \mathcal{H} \Delta \bar{\mathbf{U}} + f^T \Delta \bar{\mathbf{U}} \right\}, \text{ subject to : } \mathbf{M} \Delta \bar{\mathbf{U}} \leq \Lambda \quad (5.23)$$

where  $\mathcal{H}$  and  $f$  are taken from the cost function formulation (5.21):

$$\mathcal{H} = 2(\bar{\mathbf{H}}_e^T \mathbf{F}^T \mathbf{Q} \bar{\mathbf{F}} \bar{\mathbf{H}}_e + \mathbf{R}) \quad f = 2(\bar{\mathbf{H}}_e^T \mathbf{F}^T \mathbf{Q} * (\mathbf{F} \Gamma_e \mathbf{x}_k^e - \mathbf{P}_{ref})) \quad (5.24)$$

where the constant term  $\mathcal{C}^T \mathbf{Q} \mathcal{C}$  is absent due to its irrelevance in the optimisation problem.

For each instant the optimisation problem is solved, the future increments of the yaw angle for a given horizon, in addition to the yaw angle at that instant  $u_o$  have to be kept within the pre defined range, as described in (5.22). Mathematically, this translates into:

$$\begin{aligned} \Delta u_k + \Delta u_{k+1} + \Delta u_{k+2} + \dots + \Delta u_{k+H_p-1} &\leq u_{max} - u_o \\ \Delta u_k + \Delta u_{k+1} + \Delta u_{k+2} + \dots + \Delta u_{k+H_p-1} &\geq u_{min} - u_o \end{aligned} \quad (5.25)$$

which, rearranging so as to correspond to the QP formulation in (5.23):

$$\begin{aligned} \sum_{k=1}^{k+H_p-1} \Delta u_k &\leq u_{max} - u_o \\ - \sum_{k=1}^{k+H_p-1} \Delta u_k &\leq u_o - u_{min} \end{aligned} \quad (5.26)$$

Algebraically, the restrictions in (5.26) are accomplished resorting to matrices  $\mathbf{M}$  and  $\Lambda$ , defined as follows:

---

<sup>1</sup>Quadratic Programming (QP) is concerned with the minimization of a quadratic objective function subject to a system of linear constraints.

$$\mathbf{M} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 0 & \cdots & 0 \\ 1 & 1 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \cdots & 1 \\ -1 & 0 & 0 & \cdots & 0 \\ -1 & -1 & 0 & \cdots & 0 \\ -1 & -1 & -1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -1 & -1 & -1 & \cdots & -1 \end{bmatrix} \quad \mathbf{\Lambda} = \begin{bmatrix} u_{max} - u_o \\ u_{max} - u_o \\ u_{max} - u_o \\ \vdots \\ u_{max} - u_o \\ u_o - u_{min} \\ u_o - u_{min} \\ u_o - u_{min} \\ \vdots \\ u_o - u_{min} \end{bmatrix} \quad (5.27)$$

The optimisation problem in (5.23) is solved at each time instant, resorting to the inbuilt matlab function *quadprog*. The later solves the minimisation problem indicated subject to the inequality constraints in (5.26) and (5.27). As a result, the future optimal incremental control actions - in light of the problem in hands, the increments of  $\gamma_{t1}$  - are computed.

## 5.2.5 Receding Horizon Principle

Predictive control uses the receding horizon principle, meaning that **after computing the optimal control sequence over the defined control horizon  $H_c$ , only the first control sample is implemented**. This results in a new control setting for the first turbine. The horizon is then shifted one time sample and the optimisation is restarted with new information on the measurements.

This principle is exemplified in Figure 5.1, where it is visible that at time  $k$  the future control sequence is optimised so that the performance index  $J(u, k)$  is minimised subject to constraints. This translates into the predicted output  $y(k)$  being in close proximity to the reference  $r(k)$ . Subsequently, at time  $k$  the first element of the optimal sequence is applied to the real process.

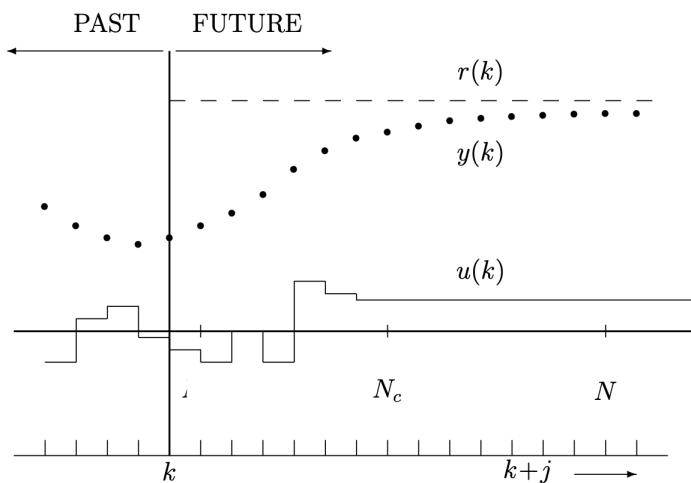


Figure 5.1: The 'Moving horizon' in predictive control, taken from [67].  $N$  corresponds to the prediction horizon often referred to as  $H_p$  and  $N_c$  to the control horizon, often referred to as  $H_c$ .

## 5.3 Model Predictive Control Implementation in SOWFA

The final implementation of the MPC controller in SOWFA is divided into three steps:

1. **Optimisation of variables in simulation environment:** MPC settings are firstly tuned based on simulations performed in MATLAB.
2. **Open-loop testing of MPC controller in SOWFA:** the optimal control action computed by the MPC at each time instant is administered to the first turbine in SOWFA. The divergences between true measurements from SOWFA and model predictions are not corrected.
3. **Closed-loop testing of MPC controller in SOWFA with observer:** MPC optimal control action is administered to the first turbine in SOWFA and at each time instant the states are estimated by using deviations of the model prediction with relation to measurements from SOWFA.

### 5.3.1 Model Predictive Control Simulation

MPC parameters, such as the prediction horizon  $H_p$ , control horizon  $H_c$ , tracking penalty matrix  $\mathbf{Q}$  and control action penalty  $\mathbf{R}$  must be tuned, so as to achieve the desired balance between tracking and control effort. The **prediction horizon  $H_p$  should be high enough so that the slow dynamics of the system are taken into account**. In other words, the impact of the first turbine control settings are only felt at the downstream turbine - in terms of generator power output - moments later. Several values are experimented with for the same collective power reference. The prediction and control horizons are chosen to be equal for simplicity purposes. Simulation results are summarised in Figure 5.2, where the influence of the prediction horizon on the tracking performance is put in evidence. The penalty matrices are kept the same for each simulation, so that the impact of the prediction horizon can be studied.

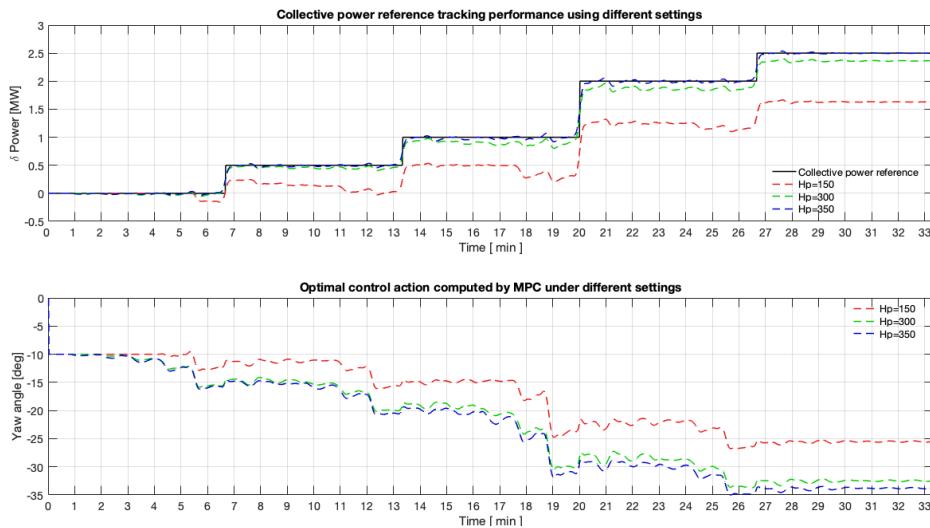


Figure 5.2: Predictive controller performance for collective power reference tracking using different values for the prediction horizon  $H_p$ . The control horizon  $H_c$  is kept the same as  $H_p$ . Penalty matrix  $\mathbf{Q}$  entries are equal to 10 and penalty matrix  $\mathbf{R}$  is equal to the identity matrix.

It is worth noticing the benefits of a predictive controller for the wind farm system, which are especially depicted in Figure 5.2, at minute 20 of the simulation. At this time instant, the collective power reference - in terms of gains with relation to the baseline position of -10 degrees - jumps from 1 to 2 [MW]. For the tracking to be possible, the **predictive controller takes into account the wind farm system inherently late dynamics and acts anticipatedly, varying the control action 2 to 3 minutes before the new reference sets in.**

Different parameters of the controller are tested. The penalty matrix  $\mathbf{Q}$  entries are varied between 10 and 100, so as to evaluate the impact in following a certain reference. The results of the experiments with these parameters are illustrated in Figure 5.3.

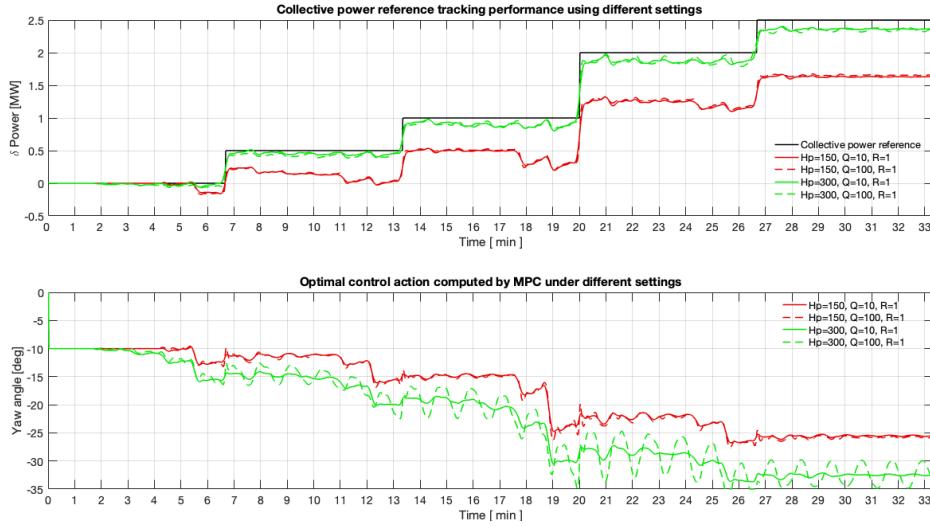


Figure 5.3: Predictive controller performance for collective power reference tracking using different values for the prediction horizon  $H_p$  and tracking penalty  $\mathbf{Q}$ . The control horizon  $H_c$  is kept the same as  $H_p$ . Red lines indicate a prediction horizon of 150 time instants and green lines 300. Solid lines indicate a penalty matrix with entries of 10 and dashed lines of 100.

The upper graph indicates that increasing the tracking penalty matrix  $\mathbf{Q}$  **does not necessarily lead to higher tracking performance and that a better performance might be related with a higher prediction horizon.** The lower graph further suggests that a higher tracking penalty might lead to more abrupt control actions. This is expected, as the tracking penalty is much higher than the control action penalty.

The several simulations made with different combinations of the controller parameters provide a handful of indications in terms of the MPC setting to use:

- **Prediction and control horizons  $H_p$  and  $H_c$ :** should be high enough so that the late dynamics are embedded into the controller. Values above 300 should provide satisfactory performance in terms of collective power tracking, based on evidence gathered in simulations. Nevertheless, caution should be taken as the optimisation step requires more computational power as the horizon increases.
- **Tracking penalty  $\mathbf{Q}$  and control penalty  $\mathbf{R}$ :** A tracking penalty of 10 and a control action penalty of 1 appear to be suitable for the tracking problem based on the simulations, presenting an adequate balance between tracking performance and control action effort.

### 5.3.2 Open-loop Testing in SOWFA

The open-loop simulation - as described in Appendix C, algorithm 2 is firstly attempted. This procedure yields additional insights into the suitability of the model for predictive control. The following controller settings are used:  $H_p = H_c = 300$ ,  $\mathbf{Q}=10$  and  $\mathbf{R}=1$ .

The optimal control action from the MPC is depicted in the upper graph of Figure 5.4. It should be noted that the first 200 seconds are used to set the yaw angle of the first turbine at -10 degrees, the baseline position, and for the wake to fully develop. The controller is implemented after this time window. The resulting collective power produced is illustrated in blue in the lower graph of the same figure, where the reference is represented in black.

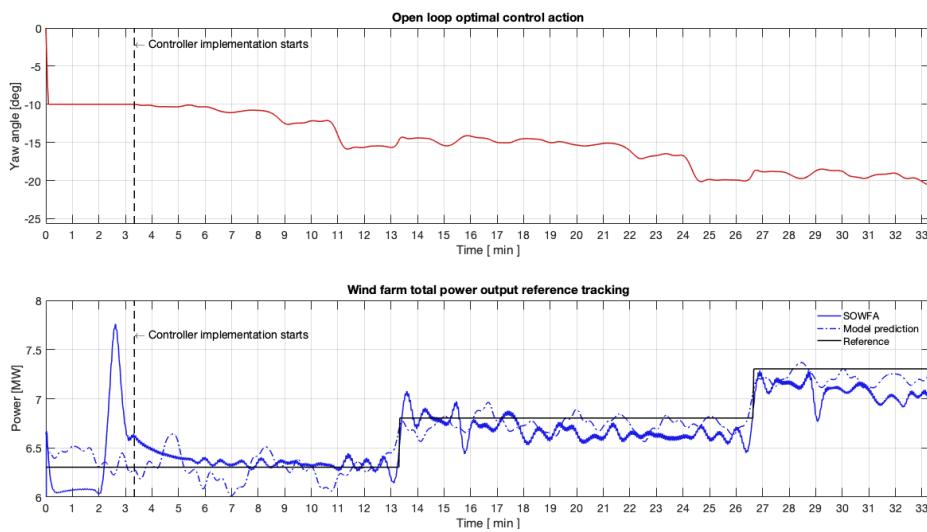


Figure 5.4: Open-loop predictive controller performance for collective power reference tracking implemented in SOWFA. Upper graph shows optimal control action and lower plot collective generator power data retrieved from SOWFA.

From the lower graph in Figure 5.4 it is shown that **the true generator power in SOWFA does not perfectly track the collective power reference, which may in part be explained by the non linearities in the system**. This hypothesis is substantiated by ascertaining the differences from SOWFA measurements and model predictions, depicted in Figure 5.5.

As the yaw angle dictated by the controller increases, so do the disparities from the model predictions and SOWFA measurements. Model estimations, represented with a dashed line in the three graphs in Figure 5.5, show that the model underestimates true power output from the upstream turbine and overestimates power production from the downstream turbine. **Collectively, this leads to an overestimation of the total power production with relation to SOWFA for higher yaw angles**, as the third graph depicts.

These discrepancies, which are somewhat expected as a consequence of having a reduced sized linear model predicting the behaviour of a non linear system, can be reduced. This is achieved by using an *observer*, as explained in the following subsection.

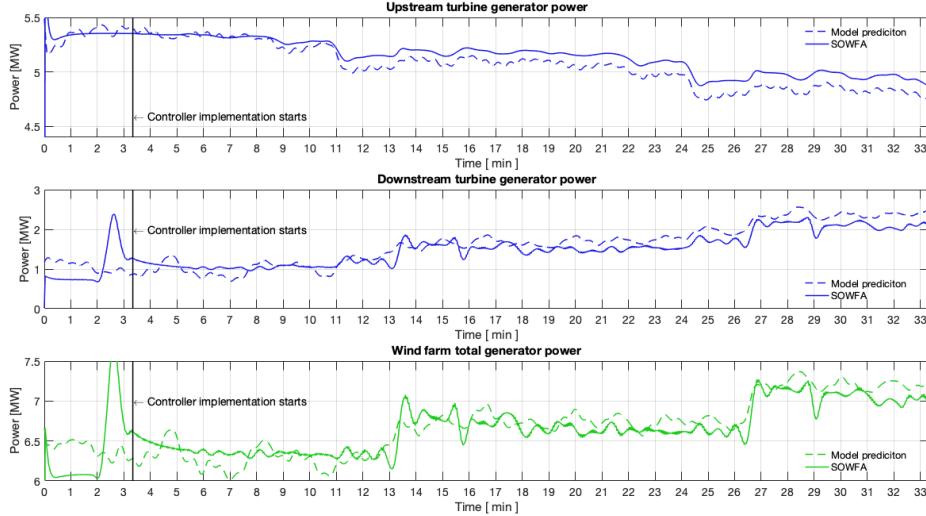


Figure 5.5: Open-loop implementation of MPC in SOWFA for tracking problem of a collective power reference. The discrepancies between predictions given by model and SOWFA measurements of turbine generated power are observable.

### 5.3.3 Closed-loop Testing in SOWFA

The MPC requires an accurate knowledge of the states to provide a reliable prediction of the power produced in each turbine. Nevertheless, it is not possible to know the correct value of the states at each time step, as the fluid flow information is not directly available from SOWFA. To overcome this, the **states are estimated by using measurement data that is accessible**, since turbine data, and specifically, power generated in each turbine, is directly accessible from SOWFA.

The estimation of the states is performed with an *observer*, which was introduced by D.Luenberger in 1963 and can be defined as a **dynamical system whose state variables are the estimates of the state variables of another system**. The decision of implementing a Luenberger observer instead of a Kalman filter - which is known to be the best possible linear estimator under certain assumptions - is due to the fact that for the wind farm control problem, information about process and measurement noise is not known, as mentioned in [1]. Therefore, **a simpler observer proves to be the most practical route for the problem in hand**. In Figure 5.6 the observer is represented in yellow, and the state estimates are fed back into the MPC.

Luenberger showed that, for any *observable* linear system, **an observer can be designed having the property that the estimation error (i.e, the difference between the state of the actual system and the state of the observer) can be made to go to zero as fast as one may desire** [27].

Defining the estimation error as the difference between the true states  $x_k$  and estimated states  $\hat{x}_k$ :

$$e_k = x_k - \hat{x}_k \quad (5.28)$$

Luenberger's method is then to make the error acceptably small. The dynamics of the estimate error,

based on the block diagram in Figure 5.6, can be written as:

$$\begin{aligned}
e_{k+1} &= \mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1} \\
&= \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k - \mathbf{L}(\mathbf{C}\mathbf{x}_k - \mathbf{C}\hat{\mathbf{x}}_k) - \mathbf{B}\mathbf{u}_k - \mathbf{A}\hat{\mathbf{x}}_k \\
&= \mathbf{A}(\mathbf{x}_k - \hat{\mathbf{x}}_k) - \mathbf{LC}(\mathbf{x}_k - \hat{\mathbf{x}}_k) \\
&= (\mathbf{A} - \mathbf{LC})\mathbf{e}_k
\end{aligned} \tag{5.29}$$

The dynamics of the estimator are then determined by the eigenvalues of  $\mathbf{A} - \mathbf{LC}$ , as equation (5.29) demonstrates. Determination of the matrix  $\mathbf{L}$  is accomplished by a pole-placement task, where the position of the poles of  $\mathbf{A} - \mathbf{LC}$  is specified. These eigenvalues can be placed arbitrarily if the pair  $(\mathbf{A}, \mathbf{C})$  is *observable*<sup>2</sup>.

The algebraic observability theorem [27] states that the (unforced) time invariant system is observable if and only if the rank<sup>3</sup> of the *observability test matrix* in (5.30)

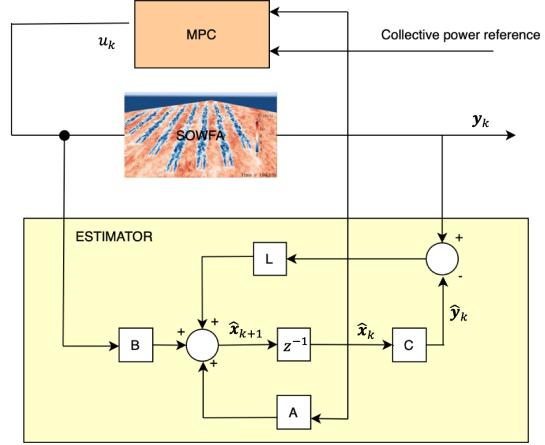


Figure 5.6: MPC closed loop schematics: simulation given a total wind farm power production reference and state estimator implementation. Adapted from [12].

$$\mathcal{O} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix} \tag{5.30}$$

is equal to the order  $n$  of the system. This condition is inspected and all states of the system are observable.

The placement of the estimator poles is achieved by making use of the inbuilt matlab command *place* and by specifying the eigenvalues to be 0.995 of the eigenvalues of  $\mathbf{A}$  (each pole of the estimator has 0.995 of the real and imaginary parts of the original discrete pole), thus creating a dynamical system faster than the original one. This value was found to be sufficient for the state estimation task, as a faster observer would lead to abrupt control actions.

The results concerning the implementation of the MPC in closed-loop with an estimator in SOWFA are depicted in Figure 5.7. The final controller settings are used:  $H_p = H_c = 350$ ,  $\mathbf{Q}=10$  and  $\mathbf{R}=1$ . The yaw angle is restricted to 0 and -35 degrees and the yaw angle rate is constrained to values between -0.25 and 0.25. The later restriction is set to avoid aggressive control actions as a consequence of state

<sup>2</sup>An unforced system is said to be observable if and only if it is possible to determine *any* (arbitrary initial) state  $x(t) = x_t$  by using only a finite record,  $y(\tau)$  for  $t \leq \tau \leq T$ , of the output.

<sup>3</sup>The rank of a matrix is the dimension of the vector space generated (or spanned) by its columns, which corresponds to the maximum number of linearly independent columns of the matrix.

correction.

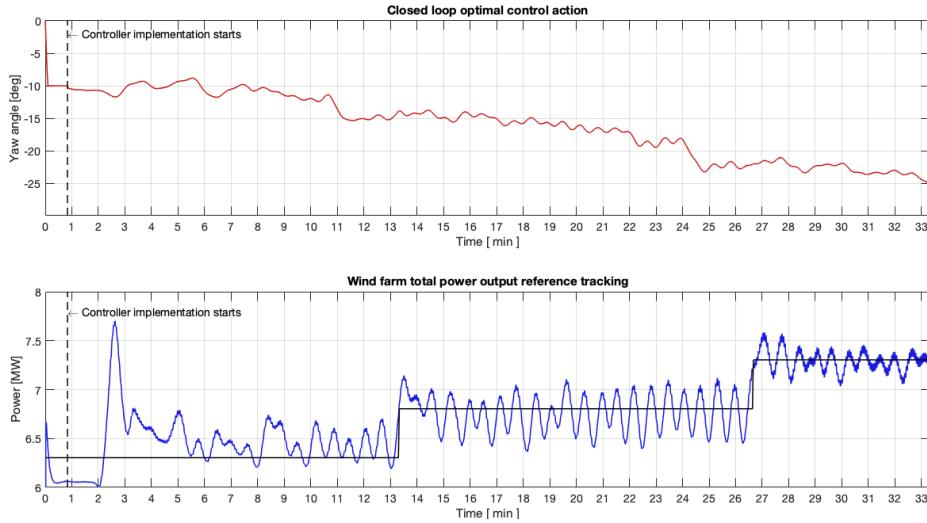


Figure 5.7: Closed-loop predictive controller performance for collective power reference tracking implemented in SOWFA. Upper graph shows optimal control action and lower plot collective generator power data retrieved from SOWFA.

The controller is implemented after the first 50 seconds. The first peak in collective power is explained by the peak in power production at the downstream turbine due to the modification of the wake's properties as the first turbine yaw angle is initially set to -10 degrees. Figure 5.8 shows how **the control action leads to oscillations of power production at the downstream turbine**. To overcome the oscillations and improve the smoothness of the tracking solution, further restrictions are imposed on the yaw angle rate. Two additional and final simulations are carried out, where the rate is restricted to values between -0.01 and 0.01 and -0.005 and 0.005.

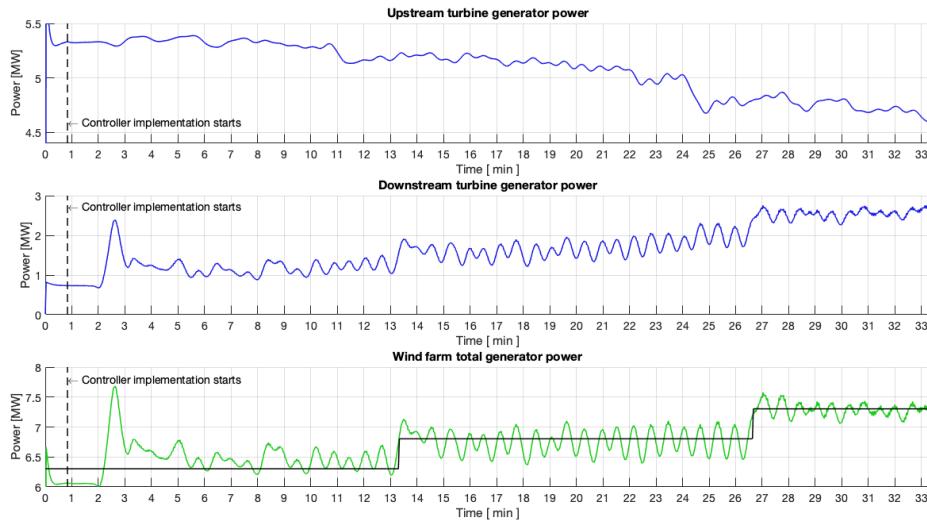


Figure 5.8: Closed-loop predictive controller performance for collective power reference tracking implemented in SOWFA: analysis at turbine level. Upper graph shows upstream turbine generator power, middle graph downstream turbine generator power and lower plot collective generator power. All data is retrieved from SOWFA.

Figure 5.9 depicts the upstream turbine yaw angle yielded by the MPC where the optimisation problem

has been further restricted. The restrictions can be seen in the lower plot of Figure 5.9 for both simulations.

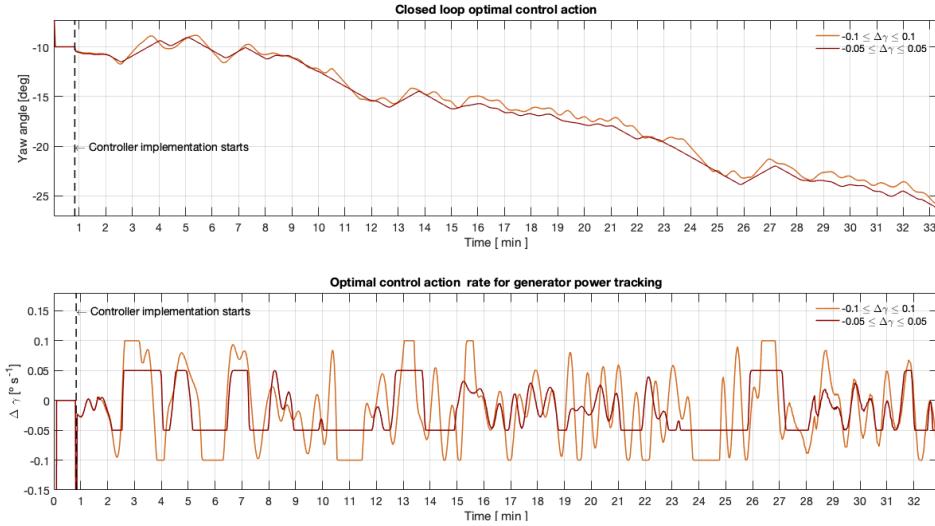


Figure 5.9: Closed-loop predictive controller performance regarding control action effort. Upper graph shows optimal control action and lower plot corresponding rate.

As a consequence of a smoother control action, **less oscillations in terms of produced power at the downstream turbine are measured**, as Figure 5.10 illustrates. Therefore, **the collective power reference can be tracked satisfactorily while simultaneously avoiding large fluctuations in total produced power**.

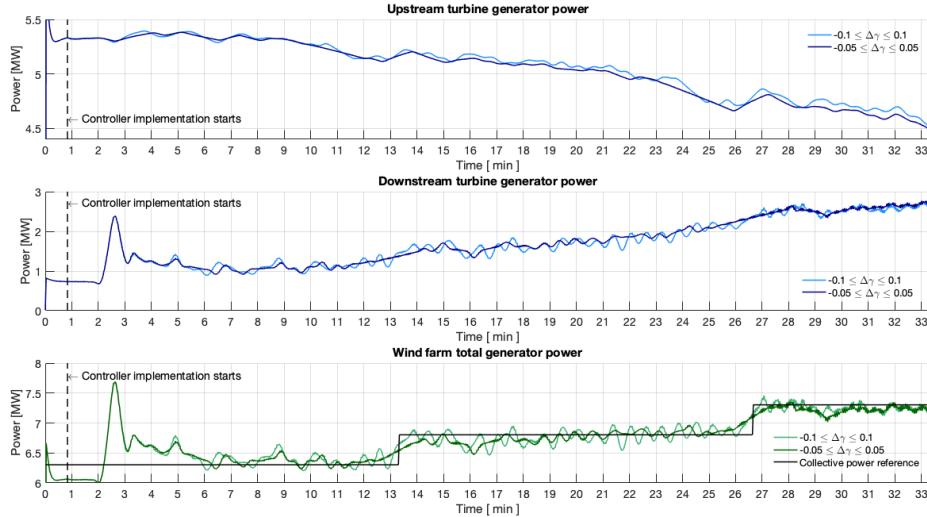


Figure 5.10: Closed-loop predictive controller performance for collective power reference tracking implemented in SOWFA with further restriction on yaw angle rate: analysis at turbine level. Upper graph shows upstream turbine generator power, middle graph downstream turbine generator power and lower plot collective generator power. All data is retrieved from SOWFA.

To the best knowledge of the author of this dissertation, the design of a MPC based on a data driven reduced order model and implementation in a CFD simulator of high fidelity (so that two wind turbines may work in a cooperative fashion to produce a pre-established amount of collective power) using wake redirection control by yaw misalignment is the first of its kind in the literature.

# CHAPTER 6

## CONCLUSIONS

---

Wind farms are highly complex systems, in the sense that they are high order systems with millions of states and the equations governing their dynamics are non linear with no known analytical solution. Controlling wind farms, therefore, becomes a challenging task, as models using the full flow field information and describing their behaviour accurately may prove to be intractable for implementation in real time.

The work in this thesis aims to **reduce the knowledge gap between data-driven methodologies which are equation free and circumvent the need to deal with the governing equations and their suitability for real time control**. A summary of the work developed is presented in this chapter by addressing three main topics: (1) final conclusions, which corresponds to the most meaningful results (2) contributions to the scientific and wind community - in terms of the most innovative aspects and (3) recommendations and future directions for further research.

### 6.1 Final Conclusions

Five main results should be emphasized and these represent the thesis conclusions:

- 1. Reduced order models are able to accurately predict wake behaviour and map it to power output by using only the streamwise velocity component.** Two wind farm control strategies are studied in this work. WRC by yaw misalignment is the core of this thesis, and AIC is described in Annex A. A ROM with 37 states is able to both predict the wake dynamics and provide an accurate mapping to power output of each wind turbine. The wake is predicted with a mean NRMSE of 4%, demonstrating the ability of a linear state space ROM in predicting the behaviour of a non linear high dimensional system, within a certain operational domain. Furthermore, the model reproduces power generated in the downstream turbine with a VAF of 88% for validation data. The case is similar for AIC, where a ROM with 38 states reconstructs the wake with a mean NRMSE of 3.75% for validation data. The power generated at each turbine presents, nevertheless, more complex dynamics when compared to WRC, and 74% of VAF for the power generated at the second turbine is registered.

**2. Ideas sustained by the Koopman operator theory are able to improve models accuracy.** For both WRC and AIC, the VAF of the power generated at the downstream turbine is increased with relation to a baseline case where the streamwise velocity component is used. This indicates that for a given control strategy there might exist some coordinate which is better suited to describe the dynamics. In the case of WRC, using  $u^2$  leads to a ROM with 26 states and a VAF of the power captured at the second turbine of 89.63%, a small increase of 1.23%. For AIC the benefits of capitalising on the ideas of the Koopman operator are higher, and it is shown that using  $w$  leads to a ROM able to predict the power at the second turbine with a VAF of 81.71%, 8.12% higher when compared to the baseline case where  $u$  is used. An interesting observable to choose which yields positive improvements in VAF for both control strategies is the sum of squares of the velocities components, such as  $u^2 + v^2$  or  $u^2 + v^2 + w^2$ . It can only be hypothesized that these measurements may provide favourable coordinates for the non linear dynamics, somewhat accounting for the total energy in the wake.

**3. The methodology used sheds light into the dynamical properties of the models.** Using IODMD allows to derive grey box models, where the states maintain physical meaning and its properties can be evaluated to gain insights into the dynamics of the system. In the case of WRC, the frequency response of different models using distinct observables is evaluated in order to gauge consistency and resonance peaks are observed in the vicinity of 0.21 St. Other peaks appear at higher frequencies, such as 0.59 and 0.69 St. The same principle applies to AIC, where consistency in the frequency response is found in two different models. The best performing model, however, shows a clear resonance peak close to 0.20 St. For both cases, similar dynamical characteristics are found. More specifically, modes with similar frequencies and damping ratios of the same order of magnitude are found in the models for both control strategies, although no explanation can be provided at this point for such findings.

**4. Model based predictive controllers are able to embed the slow wind farm system dynamics and provide an optimal control action for a given problem.** The advanced control strategy used takes into account the time the wake takes to travel from the upstream to the downstream turbine. This is evident by evaluating the control action computed by the controller scheme such that a collective power reference is followed. The two turbines are able to cooperate between themselves so that they can achieve the desired goal. An appropriate value for the prediction horizon is then pivotal so that the controller follows the reference satisfactorily. Taking into account the existing trade-off between reference tracking performance and computational power required, the value of 350 time samples for the prediction horizon is chosen.

**5. The model based controllers are suitable for real time implementation in high fidelity computational fluid dynamics solvers.** The controller is successfully implemented in SOWFA for the collective power reference tracking problem in a two step approach. The open-loop implementation of the controller provides initial promising results: the reference is tracked, showing that the optimised yaw angle takes into account the aerodynamic interaction between turbines. The subsequent closed loop implementation by designing a simple estimator using pole placement demonstrates that the collective power reference

can be followed. Furthermore, it is shown that additional constraints on the yaw angle rate lead to a smoother solution for the tracking problem.

## 6.2 Contributions

Three main contributions to the scientific community are achieved with the research conducted, which are the innovative core of this work:

- 1. Wake redirection control by yaw misalignment is modelled and controlled from a equation free perspective.** Applying pure data driven techniques and understanding the ability of IODMD to study wake steering in detail in terms of wake reconstruction and mapping to turbine's power is in itself an innovative experiment. Other existing examples in the literature place higher emphasis on AIC.
- 2. The ideas behind the Koopman operator are explored in the context of wind farm reduced order data driven models.** Leveraging the ideas of the Koopman operator to increase IODMD ability to model the system dynamics within the context of wind farm modelling and control is an unprecedented endeavour. This study, performed both to WRC and AIC, elucidates others who are using data driven techniques on the observables that provide benefits and establish baseline results that can be used as benchmarks.
- 3. A model based controller derived with a data driven technique is tested for a collective power reference tracking problem.** The design and testing of a predictive controller in a high fidelity simulator based on a ROM derived using IODMD for the control problem in this thesis - tracking a reference for collective power produced - is a first experiment of its kind in the literature.

## 6.3 Recommendations and Future Work

Several directions of research for further developments are outlined, regarding three particular areas:

- A. Application of data driven modelling techniques to other wind farm control strategies.** The same techniques applied in this dissertation can be used for other wind farm control strategies, such as IPC. The later might present more complex dynamics, nevertheless the evidence found by making use of the Koopman operator theory points towards the hypothesis that the non linear dynamics can be captured with a linear model. Controllers can then be designed and the most beneficial control settings assessed.
- B. Improvement of systems identification procedure to compute models with increased accuracy.**

- **Further explore ideas behind the Koopman operator theory:** a limited number of trials was performed where different observables were tested so as to provide a proof of concept. Nonetheless,

many others can be tested, such as the vorticity field. Expert knowledge based on fluid dynamics should lead the choosing of these observables.

- **Use more advanced data driven techniques:** Sparsity-promoting dynamic mode decomposition is an example of a more advanced technique which strives for a balance between the quality of approximation and the number of modes that are used to approximate a certain flow field. It was first introduced in [36] to better capture the dominant dynamics of the system by attempting to exclude the structures identified by standard DMD that weekly contribute to the flow. This technique is then extended to include inputs, as described in [5].
- **Use other type of models to better capture the non linear dynamics:** Linear Parameter Varying (LPV) ROMs provide a possible pathway to better capture the non linear dynamics, as described in [1]. This can be achieved by constructing different ROMs at fixed operating conditions and then stitching them together using a parameter varying linearization, whilst guaranteeing that the states of the final parameter varying ROM have consistent meaning across all operating conditions.

**C. Testing of different control problems and optimisation techniques.** Other control problems can be formulated, having always in mind the goals of wind farm control strategies (increase energy production, reduce or rebalance loads and improve grid services quality). Therefore, future work may attempt to include other outputs such as wind turbine loads so that controllers aiming to minimize loads can be designed. Other control problems include estimating the wake centre so that the velocity deficit can be redirected.

## BIBLIOGRAPHY

---

- [1] J. Annoni. *Modeling for Wind Farm Control*. PhD thesis, University of Minnesota, May 2016.
- [2] J. Annoni and P. Seiler. A low-order model for wind farm control.
- [3] J. Annoni, P. Gebraad, A. Scholbrock, P. Fleming, and J. W. Wingerden. Analysis of axial-induction-based wind plant control using an engineering and a high-order wind plant model. *Wind Energy*, 19, 08 2015.
- [4] J. Annoni, P. M. O. Gebraad, and P. Seiler. Wind farm flow modeling using an input-output reduced-order model. *2016 American Control Conference (ACC)*, pages 506–512, 2016.
- [5] J. Annoni, P. Seiler, and M. R. Jovanović. Sparsity-promoting dynamic mode decomposition for systems with inputs. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 6506–6511, 2016.
- [6] H. Arbabi, M. Korda, and I. Mezic. A data-driven koopman model predictive control framework for nonlinear partial differential equations. *2018 IEEE Conference on Decision and Control (CDC)*, pages 6409–6414, 2018.
- [7] M. Bastankhah and F. Porté-Agel. Experimental and theoretical study of wind turbine wakes in yawed conditions. *Journal of Fluid Mechanics*, 806:506–541, 2016.
- [8] P. Benner, C. Himpe, and T. Mitchell. On input-output dynamic mode decomposition. *Advances in Computational Mathematics*, 44:1751–, 12 2018.
- [9] F. D. Bianchi, D. B. Hernán, and R. J. Mantz. *Wind Turbine Control Systems : Principles, Modelling and Gain Scheduling Design*. Springer, first edition, 2007.
- [10] S. Boersma, B. M. Doekemeijer, P. M. O. Gebraad, P. A. Fleming, J. Annoni, A. K. Scholbrock, J. A. Frederik, and J.-W. van Wingerden. A tutorial on control-oriented modeling and control of wind farms. *2017 American Control Conference (ACC)*, pages 1–18, 2017.
- [11] C. Botasso. Introduction to wind turbine aerodynamics. Slides from Wind Energy Systems course from Technical University of Munich, October 2013.
- [12] M. A. Botto. State observers. Slides from Optimal Control, Department of Mechanical Engineering, Instituto Superior Técnico, 2019.

- [13] Y. B.R.Mehta. *Indutrial Process Automation Systems: Deisgn and Implementation*. 2015.
- [14] B. Brunton, L. Johnson, and N. K. Jeffrey Ojemann. Extracting spatial-temporal coherent patterns in large-scale neurral recordings using dynamic mode decomposition. *Journal of Neuroscience Methods*, 258:1–15, January 2016. ISSN 0165-0270.
- [15] S. L. Brunton, B. W. Brunton, J. L. Proctor, and J. N. Kutz. Koopman invariant subspaces and finite linear representations of nonlinear dynamical systems for control. *PLOS*, 11(2), Feb 2016.
- [16] F. Campagnolo, V. Petrović, C. Bottasso, and A. Croce. Wind tunnel testing of wake control strategies. pages 513–518, 07 2016.
- [17] N. Cassamo and J.-W. van Wingerden. Model predictive control for wake redirection: a koopman dynamic mode decomposition approach. *2021 American Control Conference (ACC)*, 2020. To be submitted.
- [18] N. Cassamo and J.-W. van Wingerden. On the potential of reduced order models for wind farm control: a koopman dynamic mode decomposition approach. *Energies*, 2020. To be submitted.
- [19] K. Chen, J. Tu, and C. Rowley. Variants of dynamic mode decomposition: Boundary condition, koopman, and fourier analyses. *Journal of Nonlinear Science*, 22, 12 2012.
- [20] M. J. Churchfield, S. Lee, J. Michalakes, and P. J. Moriarty. A numerical study of the effects of atmospheric and wake turbulence on wind turbine dynamics. *Journal of Turbulence*, 13:N14, 2012.
- [21] B. M. Doekemeijer, S. Kern, S. Maturu, S. Kanev, B. Salbert, J. Schreiber, F. Campagnolo, C. L. Bottasso, S. Schuler, F. Wilts, T. Neumann, G. Potenza, F. Calabretta, F. Fioretti, and J.-W. van Wingerden.
- [22] P. Fleming, J. King, K. Dykes, E. Simley, J. Roadman, A. Scholbrock, P. Murphy, J. K. Lundquist, P. Moriarty, K. Fleming, J. van Dam, C. Bay, R. Mudafort, H. Lopez, J. Skopek, M. Scott, B. Ryan, C. Guernsey, and D. Brake. Initial results from a field campaign of wake steering applied at a commercial wind farm - part 1.
- [23] P. Fleming, P. Gebraad, S. Lee, J. W. Wingerden, K. Johnson, M. Churchfield, J. Michalakes, P. Spalart, and P. Moriarty. Evaluating techniques for redirecting turbine wake using sowfa. *Renewable Energy*, 70:211–218, 10 2014.
- [24] P. Fleming, J. Annoni, J. Shah, L. Wang, S. Ananthan, Z. Zhang, K. Hutchings, P. Wang, W. Chen, and L. Chen. Field test of wake steering at an offshore wind farm. *Wind Energy Science Discussions*, pages 1–17, 02 2017.
- [25] J. Frederik, B. Doekemeijer, S. Mulders, and J. W. Wingerden. The helix approach: using dynamic individual pitch control to enhance wake mixing in wind farms, 12 2019.

- [26] J. Frederik, R. Weber, S. Cacciola, F. Campagnolo, A. Croce, C. Bottasso, and J. W. Wingerden. Periodic dynamic induction control of wind farms: proving the potential in simulations and wind tunnel experiments. *Wind Energy Science Discussions*, pages 1–18, 08 2019.
- [27] B. Friedland. *Control Systems Design. An Introduction to State-Space Methods*. Dover Publications, Inc, 2018.
- [28] P. M. O. Gebraad. *Data-Driven Wind Plant Control*. PhD thesis, Technische Universiteit Delft, 2014.
- [29] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, third edition, 1996.
- [30] J. H. Tu, C. W. Rowley, D. M. Luchtenburg, S. L. Brunton, and J. Nathan Kutz. On dynamic mode decomposition: Theory and applications. *Journal of Computational Dynamics*, 1(2):391–421, 2014. ISSN 2158-2505.
- [31] P. Holmes, J. Lumley, G. Berkooz, and C. Rowley. *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge University Press, New York, United States, second edition, 2002.
- [32] M. F. Howland, S. K. Lele, and J. O. Dabiri. Wind farm power optimization through wake steering. *Proceedings of the National Academy of Sciences*, 116(29):14495–14500, 2019. ISSN 0027-8424.
- [33] G. V. Iungo, C. Santoni-Ortiz, M. Abkar, F. Porté-Agel, M. A. Rotea, and S. Leonardi. Data-driven reduced order model for prediction of wind turbine wakes. *Journal of Physics: Conference Series*, 625:012009, jun 2015. doi: 10.1088/1742-6596/625/1/012009.
- [34] J. J and G. A.L.Rogers. *Wind Energy Explained*. John Wiley and Sons Ltd., second edition, 2008.
- [35] N. Jenkins, A. Burton, D. Sharpe, and E. Bossanyi. *Wind Energy Handbook*. John Wiley and Sons Ltd, United Kingdom, third edition, 2001.
- [36] M. Jovanovic, P. Schmid, and J. Nichols. Sparsity-promoting dynamic mode decomposition. *Physics of Fluids*, 26, 09 2013.
- [37] T. Katayama. *Subspace Methods for System Identification*. Springer, 2005.
- [38] I. Kevrekidis, C. Rowley, and M. Williams. A kernel-based method for data-driven koopman spectral analysis. *Journal of Computational Dynamics*, 2:247–265, 05 2016.
- [39] I. Komusanac, G. Brindley, and D. Fraile. Wind energy in europe: Trends and statistics. Technical report, Global Wind Energy Council, February 2020.
- [40] B. O. Koopman. Hamiltonian systems and transformation in hilbert space. *Proceedings of the National Academy of Sciences*, 17(5):315–318, 1931.
- [41] M. Korda and I. Mezić. Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control. *Automatica*, 93:149–160, Jul 2018. ISSN 0005-1098.

- [42] J. Kutz, J. Proctor, and S. Brunton. Applied koopman theory for partial differential equations and data-driven modeling of spatio-temporal systems. *Complexity*, 2018:1–16, 12 2018.
- [43] J. N. Kutz, S. L. Brunton, B. W. Brunton, and J. L. Proctor. *Dynamic Mode Decomposition: Data-Driven Modelling of Complex Systems*. siamr, 2016.
- [44] L. Ljung. *System Identification: Theory for the User*. Prentice Hall PTR, 1999.
- [45] H. Lu and D. M. Tartakovsky. Predictive accuracy of dynamic mode decomposition. *arXiv: Numerical Analysis*, 2019.
- [46] P. Z. H.-O. P. D. R. J. S. P. S. A. P. W. M.-O. C. P. R. P. S. C. J. M. Y. C. X. Z. M. G. E. L. T. M. M. T. Masson-Delmotte, V. and T. Waterfield. Summary for policymakers. in: Global warming of 1.5c. an ipcc special report on the impacts of global warming of 1.5c above pre-industrial levels and related global greenhouse gas emission pathways, in the context of strengthening the global response to the threat of climate change, sustainable development, and efforts to eradicate poverty. Technical report, Intergovernmental Panel on Climate Change.
- [47] I. Mezic. Analysis of fluid flows via spectral properties of the koopman operator. *Annual Review of Fluid Mechanics*, 45:357–378, 01 2013.
- [48] S. J. Miklavcic. *An Illustrative Guide to Multivariate and Vector Calculus*. Springer, 2020.
- [49] R. Morel and I. Shishlov. Ex-post evaluation of the kyoto protocol: four key lessons for the 2015 paris agreement. Technical report, A Caisse des Dépôts et Consignations, May 2014.
- [50] W. Munters and J. Meyers. An optimal control framework for dynamic induction control of wind farms and their interaction with the atmospheric boundary layer. *Philosophical Transactions of the Royal Society of London, Series A: Mathematical, Physical and Engineering Sciences*, 375(2091), April 2017. ISSN 1364-503X.
- [51] W. Munters and J. Meyers. Towards practical dynamic induction control of wind farms: analysis of optimally controlled wind-farm boundary layers and sinusoidal induction control of first-row turbines. *Wind Energy Science*, 3(1):409–425, June 2018. ISSN 2366-7443. doi: 10.5194/wes-3-409-2018.
- [52] W. Munters and J. Meyers. Dynamic strategies for yaw and induction control of wind farms based on large-eddy simulation and optimization. *Energies*, 11, Jan. 2018. ISSN 1996-1073.
- [53] F. M. White. *Fluid Mechanics*. McGraw-Hill, 6th edition, 2008.
- [54] U. Nations. Kyoto protocol to the united nations framework convention on climate change, 1998.
- [55] K. Ogata. *Modern Control Engineering*. Prentice Hall, fifth edition, 2010.
- [56] J. Proctor and P. Eckhoff. Discovering dynamic patterns from infectious disease data using dynamic mode decomposition. *International Health*, January 2015.

- [57] J. Proctor, S. Brunton, and J. Kutz. Generalizing koopman theory to allow for inputs and control. *SIAM Journal on Applied Dynamical Systems*, 17, 02 2016.
- [58] J. L. Proctor, S. L. Brunton, and J. N. Kutz. Dynamic mode decomposition with control. *SIAM Journal on Applied Dynamical Systems*, 15(1):142–161, Jan 2016.
- [59] H. Ritchie. Renewable energy. *Our World in Data*, 2017.
- [60] C. Rowley, I. Mezic, S. Bagheri, P. Schlatter, and D. Henningson. Spectral analysis of nonlinear flows. *Journal of Fluid Mechanics*, 641:115 – 127, 12 2009.
- [61] R. S. Samiddha Mukherjee. Big data - concepts, applications, challenges and future scope. *International Journal of Advanced Research in Computer and COmmunications Engineering*, 5, February 2016.
- [62] P. Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656:5–28, August 2010.
- [63] P. Schmid and J. Sesterhenn. Dynamic mode decomposition of numerical and experimental data. volume 61st Annual Meeting of the APS Division of Fluid Dynamics. American Physical Society, November 2008.
- [64] P. Schmid, K. E. Meyer, and O. Pust. Dynamic mode decomposition and proper orthogonal decomposition of flow in a lid-driven cylindrical cavity. *8th International Symposium on Particle image Velocimetry - PIV09*, August 2009.
- [65] M. Steinbuch, W. Boer, de, O. Bosgra, S. Peeters, and J. Ploeg. Optimal control of wind power plants. *Journal of Wind Engineering and Industrial Aerodynamics*, 27(1-3):237–246, 1988.
- [66] L. Trefethen and D. Bau. *Numerical Linear Algebra*. Society for Industrial and Applied Mathematics, 1997.
- [67] T. J. van den Boom and T. C. Backx. *Model Predictive Control*. Delft University of Technology, Delft Center for Systems and Control, September 2005. Lecture Notes for the course SC4060.
- [68] T. Van Gestel, J. Suykens, P. Van Dooren, and B. De Moor. Identification of stable models in subspace identification by using regularization. *Automatic Control, IEEE Transactions on*, 46:1416 – 1420, 10 2001.
- [69] J. W. van Wingerden, P. A. Fleming, T. Göçmen, I. Eguinoa, B. M. Doekemeijer, K. Dykes, M. Lawson, E. Simley, J. King, D. Astrain, M. Iribas, C. L. Bottasso, J. Meyers, S. Raach, K. Kölle, and G. Giebel. Expert elicitation on wind farm control, 2020.
- [70] M. Verhaegen, V. Verdult, , and N. Bergboer. *Filtering and System Identification: An Introduction to using Matlab Software*. Delft University of Technology, Delft Center for Systems and Control, August 2007. SOFTWARE MANUAL FOR THE LTI SYSTEM IDENTIFICATION TOOLBOX.

- [71] M. Viberg. Subspace-based methods for the identificaiton of linear time-invariant systems. *Automatica*, 31(12):1835–1851, 1995.
- [72] M. Williams, I. Kevrekidis, and C. Rowley. A data-driven approximation of the koopman operator: Extending dynamic mode decomposition. *Journal of Nonlinear Science*, 25, 08 2014.
- [73] F. Zahle, R. Bitsche, T. Ki, A. Yde, L. C. Henriksen, M. H. Hansen, J. Blasques, M. Gaunaa, and A. Natarajan. *The DTU 10-MW Reference Wind Turbine*. Section for Aeroelastic Design and Section for Structures Technical University of Denmark DTU Wind Energy – Risø Campus, 2013.

## APPENDIX A

### AXIAL INDUCTION CONTROL

---

The algorithms and program used to obtain all results showed throughout this thesis **can also be applied for other wind farm control strategies**. This means that after performing a simulation in SOWFA where a certain perturbation is given to excite the system - with the final goal of observing increases in generated power at the wind farm level when compared to a greedy control strategy - turbine data and flow field data can be quickly loaded, analysed and used to build reduced order models.

More specifically, the algorithms, given data from a simulation in SOWFA, by running the program created for this work, allow to:

1. **Visualise flow field information:** specific functions allow to qualitatively represent the wake moving downstream, such as *makeframes.m* and *cuthubheightvec.m* - to create animations - or *wake\_vorticity\_deflection\_yaw.m* and *wake\_vorticity\_pitch.m* - to create snapshots. From the complete flow field, the isosurface of the vorticity field is computed and represented.
2. **Visualise turbine information:** function *visualisefirstresults.m* automatically reads data from a specified directory and plots information about tower nacelle yaw angle, torque, rotor speed, blade pitch angles, turbine loads and control law variables of both turbines.
3. **Calculate models:** *dynamicmodedecomposition.m* is used to compute the models given input/output and states information. The type of DMD algorithm has to be specified along with the maximum number of models to obtain.
4. **Validate models:** models are validated by predicting outputs for a different data set, using the function *validatemodels.m*. The VAF by model is represented for identification and validation data by making use of function *idvaloverview*.
5. **Analyse dynamical properties of models:** model's frequency response is then evaluated, along with representation of the eigenvectors of the high dimensional state matrix **A**. *dynamicalanalysis.m* and *visualisepodmodes* are two examples of functions developed for this purpose.
6. **Reconstruct flow field:** The flow field is then reconstructed by evaluating the state trajectory of the best performing model, measured firstly by the VAF criteria, and projecting to the high

dimensional state representation. To qualitatively evaluate the wake reconstruction, the functions *compareconstruction.m* and *compareconstruction\_pitch.m* are used.

All information is directly saved to a specified directory in the computer (or hard drive, as a single run of the program yields gigabytes of results) of the user. The main functions of the program, the workflow of variables and the parameters the user should define are schematically depicted in Annex C.

To demonstrate the usefulness of the program, another wind farm control strategy is analysed, and the results are here presented. **The control strategy in study is Axial Induction Control (AIC) by collectively pitching the blades.** Following the system identification loop, the simulation and data record are the following (simulation design and procedure is the same as the one described in Chapter 4)

**A. Inputs are RBS of blade pitch angles, which are equal for all blades.** Pitch angle varies from 0 to 4 degrees at upstream turbine and remain unchanged (no pitching) at downstream turbine.

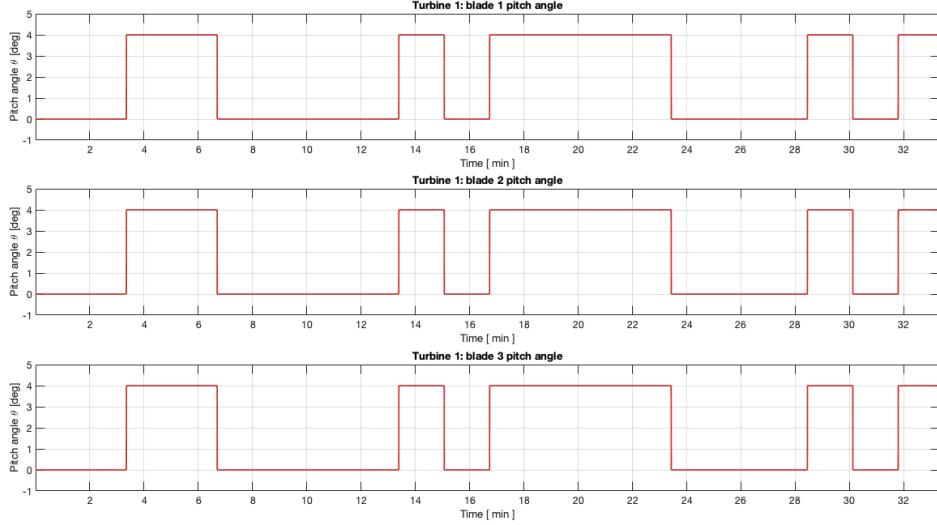


Figure A.1: Collective pitch angle variation of upstream turbine  $\theta_{t1}$  during simulation.

**B. Generator power in each turbine presents more complex dynamics when compared to the yaw control strategy evaluated in chapter 4.**

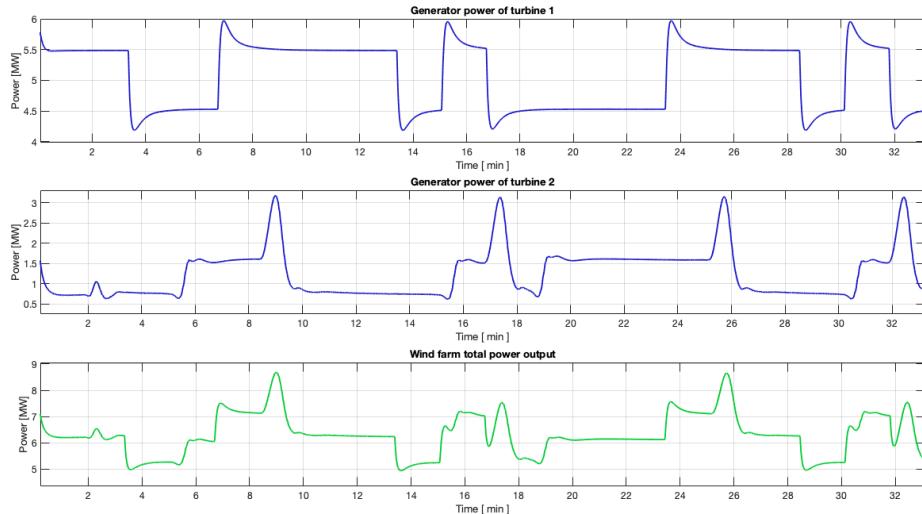


Figure A.2: Generator power variations of both turbines during simulation and total generated power at wind farm level.

The effectiveness of the wind farm control strategy can be observed in figure A.2, which is lower when compared to yaw deflection and in accordance with evidence present in the literature. The power gained at the downstream turbine - 0.85 [MW] does not compensate for the power lost up the upstream turbine when the turbine blades are pitched - 0.9 [MW]. This can be seen in minute 20 to 24 of the simulation. The benefits are only visible momentarily and at some instances. For example, from minutes 7 to 9.

**C. Wake dynamics are mostly visible at the downstream turbine rotor, where more energy is available.** Power variations are explained by variations in the wake's characteristics, which can qualitatively be visualised in Figures A.3 and A.4, and quantitatively in Figure A.5:

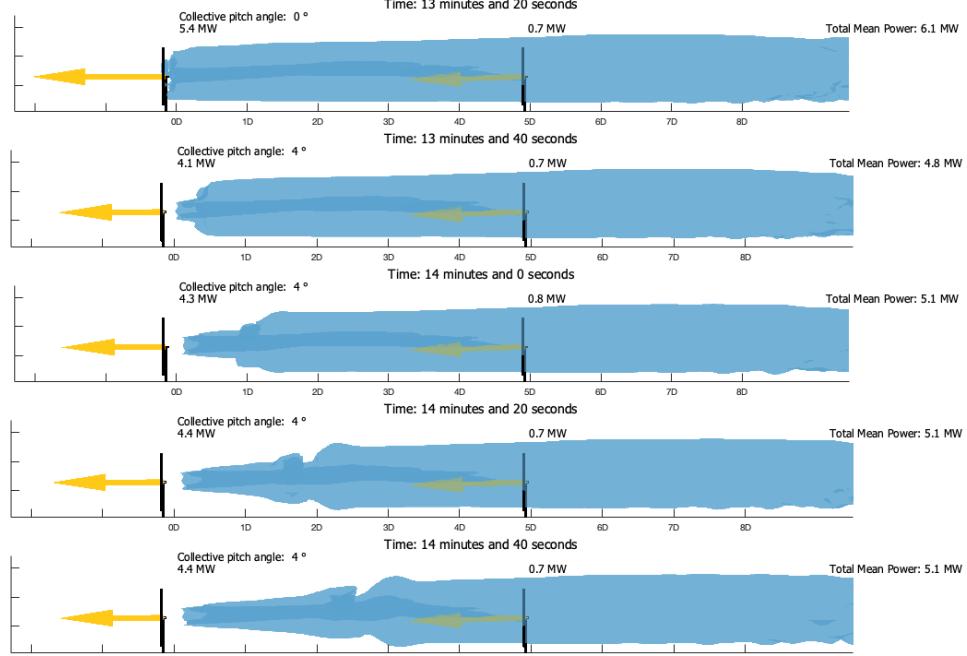


Figure A.3: Wake dynamics: (1) greedy control operating point (2) blades pitch 4 degrees and power captured by turbine 1 decreases (3), (4) and (5) wakes propagates downstream.

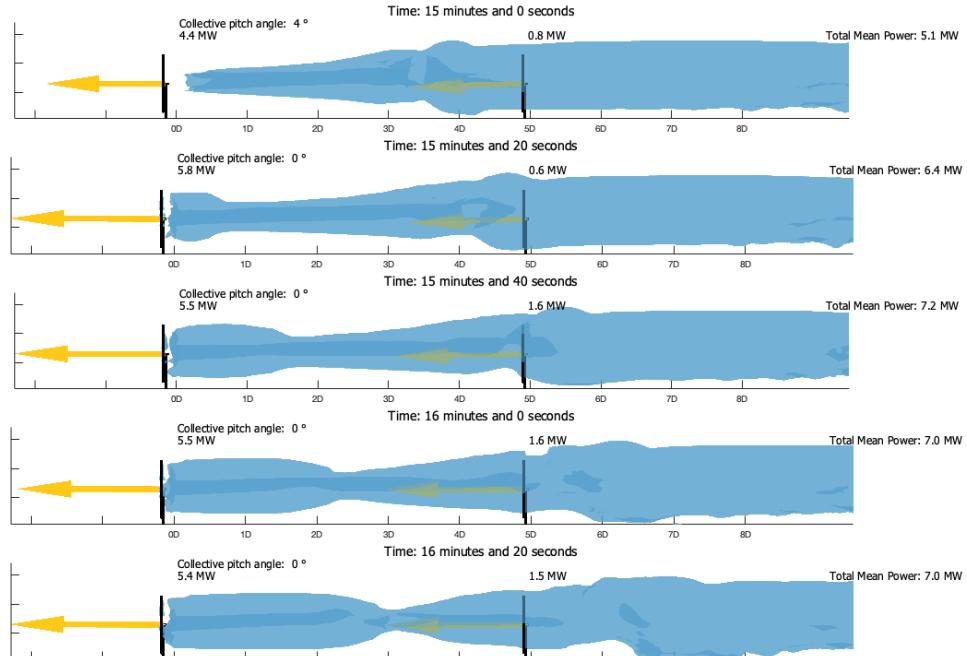


Figure A.4: Wake dynamics: (1) wake continues to move downstream (2) upstream turbine blades pitch back to 0 degrees and collective captured power increases to greedy control levels (3) altered wake hits downstream rotor and collective power increases (4) and (5) altered wake continues to propagate downstream. This latest control configuration explains the momentarily gains.

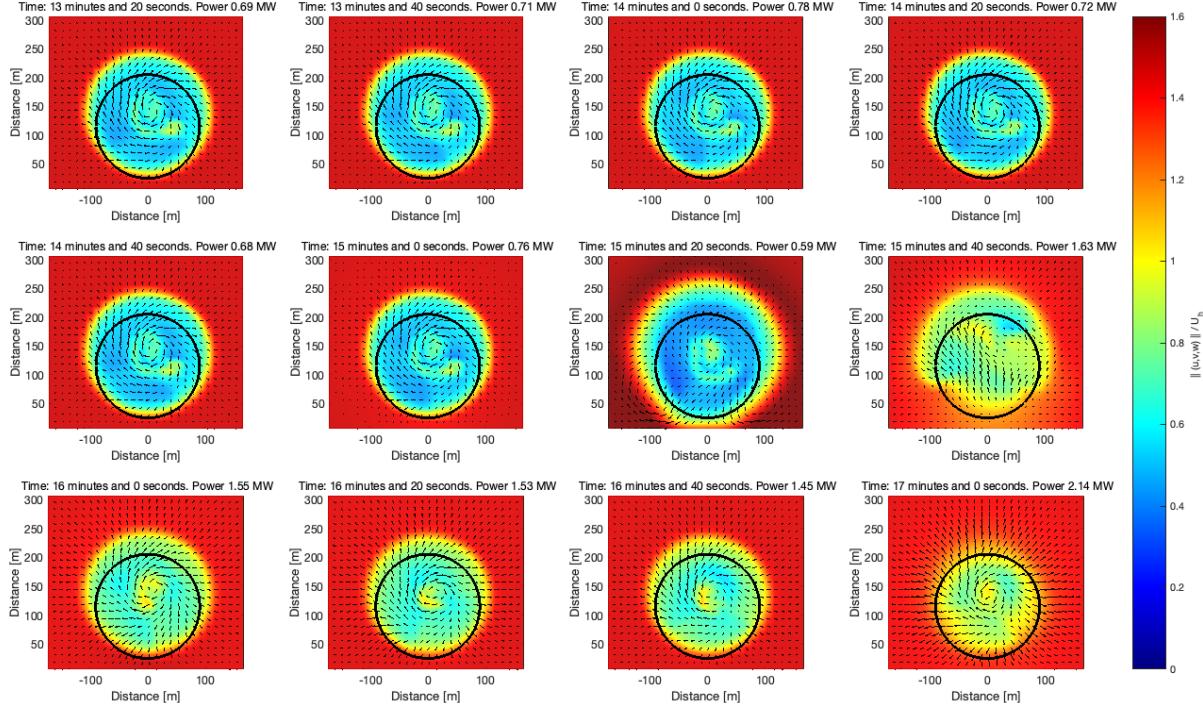


Figure A.5: Wake dynamics at downstream rotor plane: (1) first turbine blades pitch 4 degrees and wake will travel downstream (2) - (6) wake reaches downstream rotor almost 2 minutes later (7) centre of the wake presents decreased velocity deficit, leading to higher power capture (9) - (11) higher power captured by second turbine (12) altered wake due to pitching back to 0 degrees by upstream turbine reaches downstream rotor and a peak in power is registered.

**D. Data is pre processed before calculating models.** More specifically, turbine data is resampled and detrended. The processed information is visible in Figure A.6, which is directly used, along with the states, for the systems identification task.

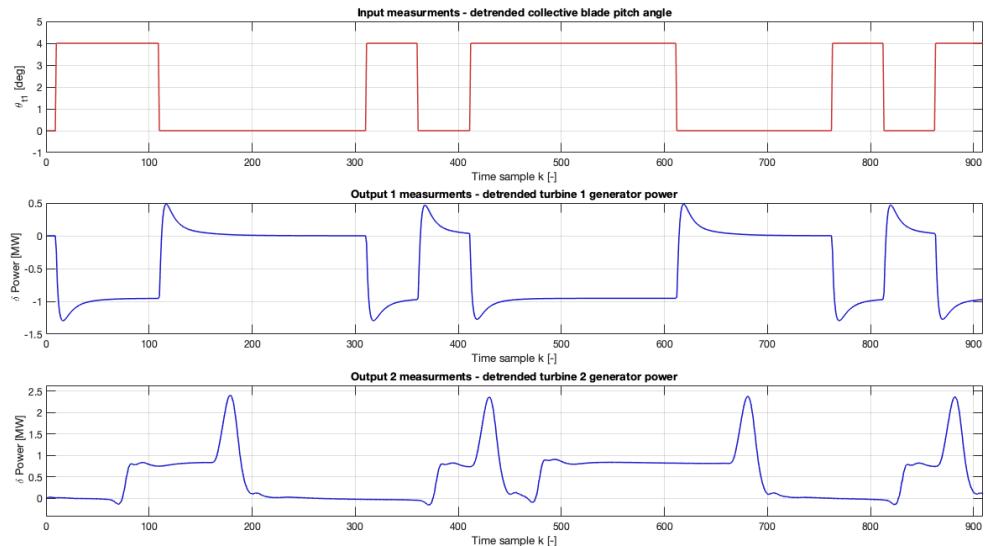


Figure A.6: Input-output measurements to be used for systems identification: detrended and resampled upstream turbine collective blades pitch angle and generator power of both turbines during simulation.

**E. A set of validation data is also generated, and then resampled and detrended.**

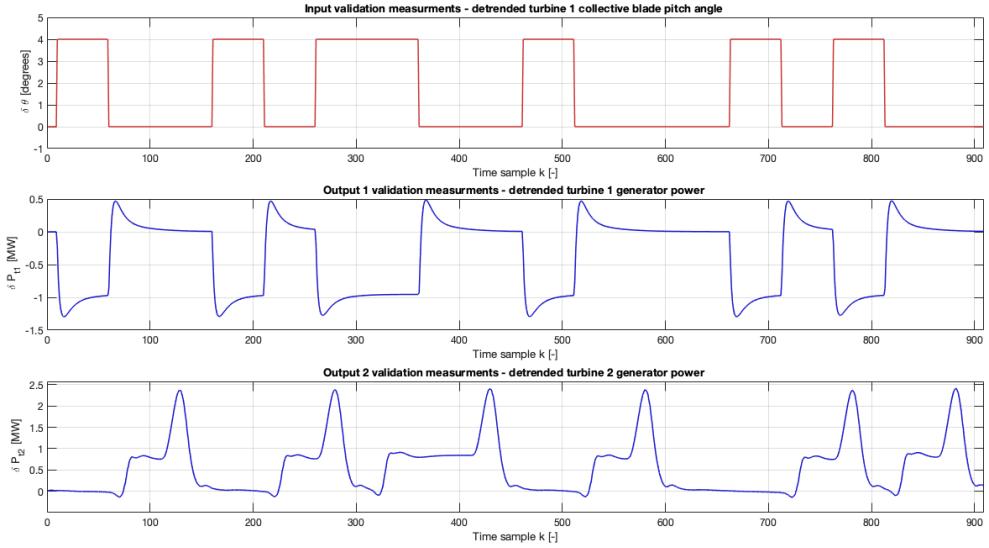


Figure A.7: Input-output measurements to be used for model validation: detrended and resampled upstream turbine collective pitch angle and generator power of both turbines during simulation.

**F. IODMD is used to compute models, using  $u$  component for the states.** This serves as the baseline result.

Table A.1: Models validation results using IODMD algorithm

Model properties	Model obtained by IODMD					
	Observable	VAF(WT2) <sub>max</sub>	Model size	VAF(WT1)  <sub>r</sub>	NRMSE (id)	NRMSE (val)
$u$		73.59 %	38	98.08 %	3.66%	3.75%

- Best model has VAF of 73.59% of the downstream turbine generated power.** The performance of the best model can be seen in Figure A.9. Using IODMD<sub>u</sub>, where velocity  $u$  is chosen as state, the best model has a VAF of 74%, which is lower compared to the model used for yaw control computed in chapter 4, which presented a VAF of 88%. It is also important to note that the spikes in power are the most difficult to model.
- A certain number of modes is required to properly recover the dynamics.** In Figure A.8 it is shown that to reach a VAF of 70%, at least 28 modes are needed. This again indicates the increasingly difficult task of modelling the dynamic for pitch control, when compared to yaw control. In the later, only 4 modes are needed to reach a VAF of 80%.
- Adding more modes does not directly lead to more accurate models.** This pattern is recognizable in Figure A.9, where the gap between VAF for the same models under identification and validation data widens as more modes are used. This is possibly explained by overfitting, where the models correspond to closely to the data set used for identification.

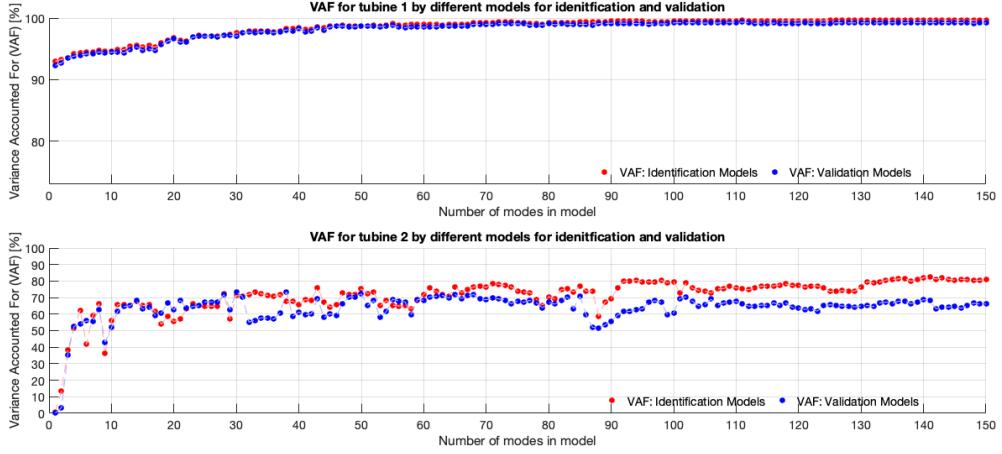


Figure A.8: VAF of models. Each model, from left to right, has increasing number of states. In red, output performance using identification data and in blue using validation data

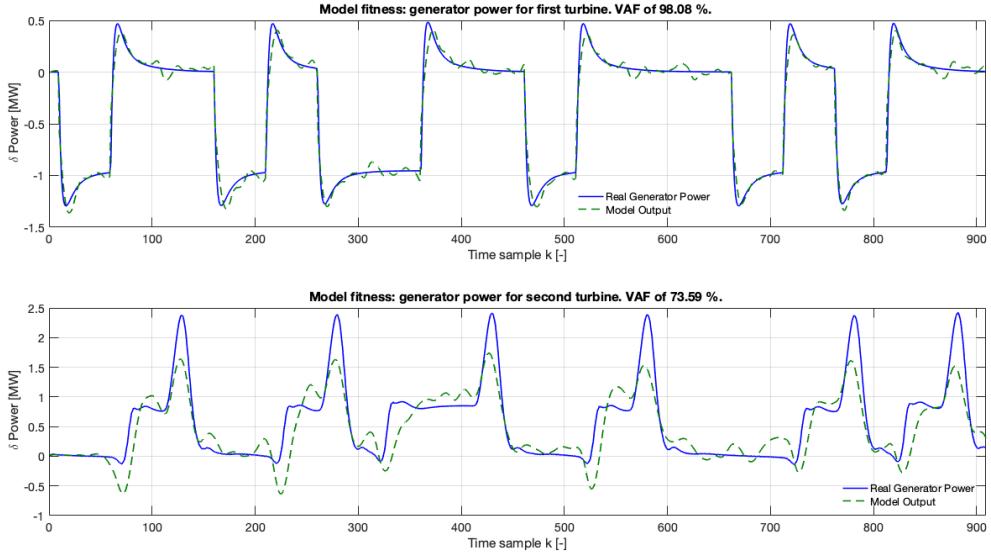


Figure A.9: Best performing model validation results. In blue, real generated power in upstream and downstream turbines retrieved from SOWFA. In green, response from model.

The IODMD<sub>u</sub> model can also be evaluated in its ability to reproduce the wake's characteristics. Similarly to the analysis performed for the yaw control strategy, the full velocity information at all points throughout the complete linear simulation is obtained from the reduced state vector. A mean NRMSE of 3.75% for validation data shows that the model can be used to predict the wake's characteristics, which is depicted in Figures A.10 and A.11. These Figures illustrate how **IODMD reconstruction compares to SOWFA data at the downstream turbine rotor plane**. An important observation to note is that the **gap between SOWFA information of streamwise velocity and corresponding model reconstruction widens at certain instant, namely when the changed wake hits the downstream rotor**. This is specially noticeable in Figure A.11, in the middle snapshot. In the other snapshots the wake is fully developed and IODMD<sub>u</sub> appears to better reconstruct its properties.

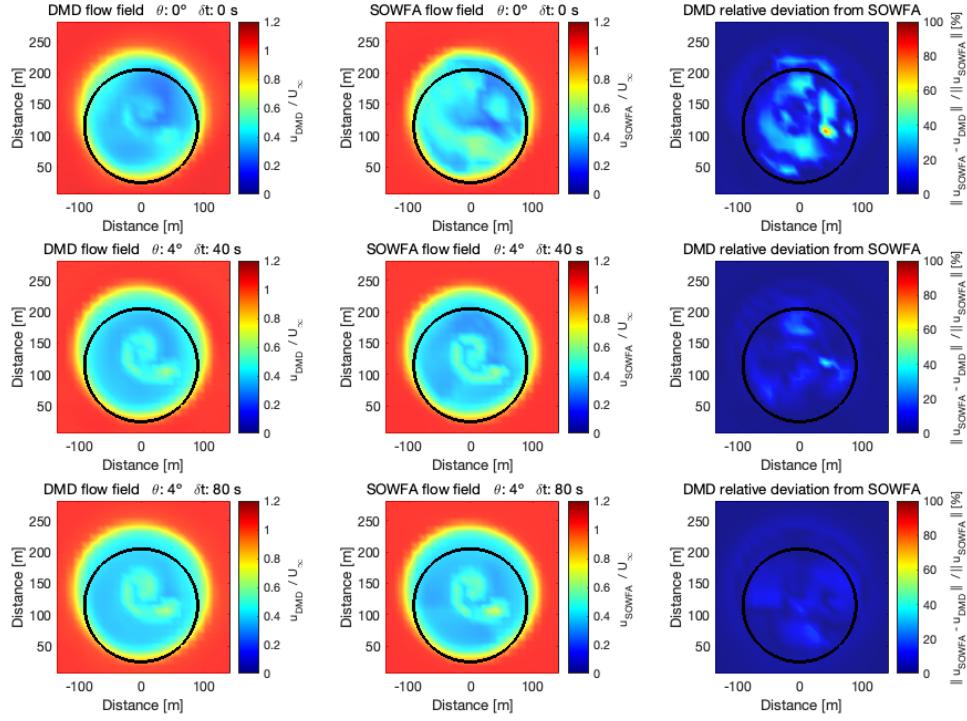


Figure A.10: Comparison of flow field reconstruction using reduced linear model (left) and data from SOWFA (middle) at downstream turbine rotor plane at different instances (during a total of 80 seconds). IODMD<sub>u</sub> flow field deviation with relation to SOWFA data is shown on the right.

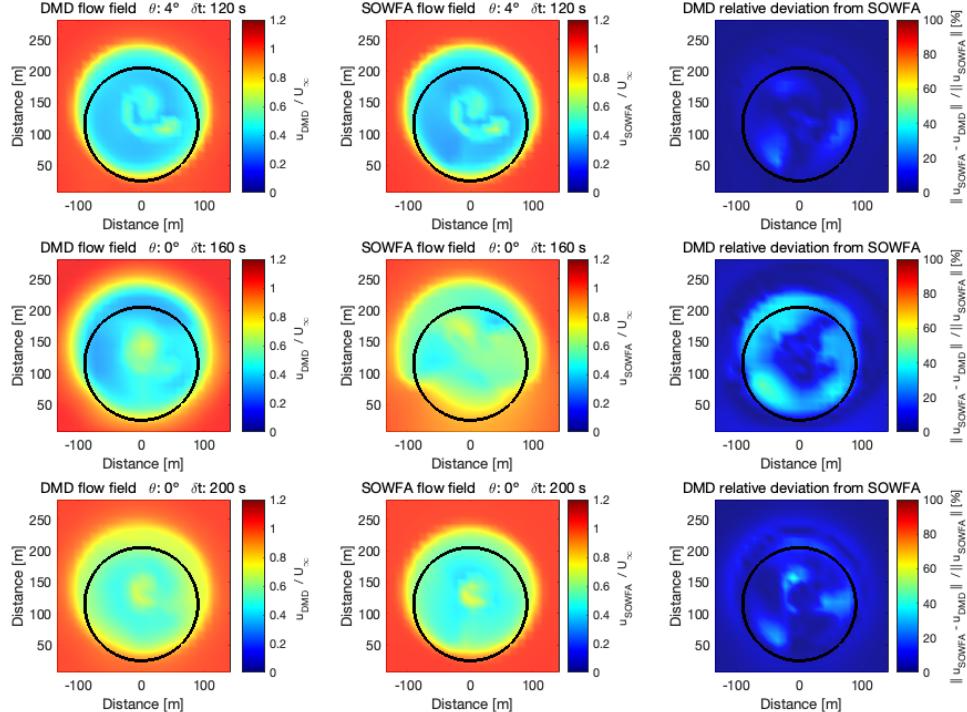


Figure A.11: Comparison of flow field reconstruction using reduced linear model (left) and data from SOWFA (middle) at downstream turbine rotor plane at different instances (additional 80 seconds). IODMD<sub>u</sub> flow field deviation with relation to SOWFA data is shown on the right.

The dynamical properties of the ROM are summarized in Table A.2. These properties are taken from the corresponding poles on the complex plane, whose location dictates the dynamics of the corresponding mode. The poles location can be seen in Figure A.12. DMD modes have been ordered by their natural frequency.

Table A.2: Dynamical properties of IODMD model.

Modes		Dynamical properties		
#	DMD	$\omega_n$ [Hz]	$\omega_n$ St [-]	$\xi$
1	27	0	0	1
2	34	0.0014	0.0283	1
3	37-38	0.0030	0.0585	0.7608
4	35-36	0.0032	0.0637	0.2041
5	32-33	0.0064	0.1268	0.3072
6	30-31	0.0088	0.1735	0.0275
7	28-29	0.0109	0.2162	0.1192
8	25-26	0.0129	0.2559	0.0497
9	23-24	0.0158	0.3124	0.0712
10	21-22	0.0194	0.3830	0.0140
11	19-20	0.0207	0.4087	0.0278
12	17-18	0.0247	0.4884	0.0358
13	15-16	0.0263	0.5204	0.0280
14	13-14	0.0301	0.5959	0.0066
15	11-12	0.0343	0.6787	0.0025
16	9-10	0.0404	0.7996	0.0026
17	7-8	0.0472	0.9331	0.0100
18	5-6	0.0560	1.1071	0.0040
19	3-4	0.1479	2.9251	0.0091
20	1-2	0.2249	4.4473	0.0482

Each of the modes corresponds to a coherent structure, meaning a spatial pattern of the fluid which evolves in time. These coherent structures can be visualised and add value to the interpretation of the dynamics of the wake, a fundamental step in the further development and usage of state of the art wind farm control strategies. Some of the modes have been represented in Figure A.14, A.15 and A.16, for different frequency domains.

It is important to note that the action of the control action is not represented in these figures. They are merely representations of the high order eigenvectors of state matrix **A**. Isosurface of the different values of the velocity field are sketched so that the complete structure may be visualized.

The eigenvalues, in both discrete time and continuous time, are represented in Figure A.12. For each pole, a mode in space which evolves in time exists, corresponding to a coherent structure that explains the dynamics of the wake. These structures, whose corresponding poles have been labelled with numbers in Figure A.12, are represented in Figures A.14, A.15 and A.16.

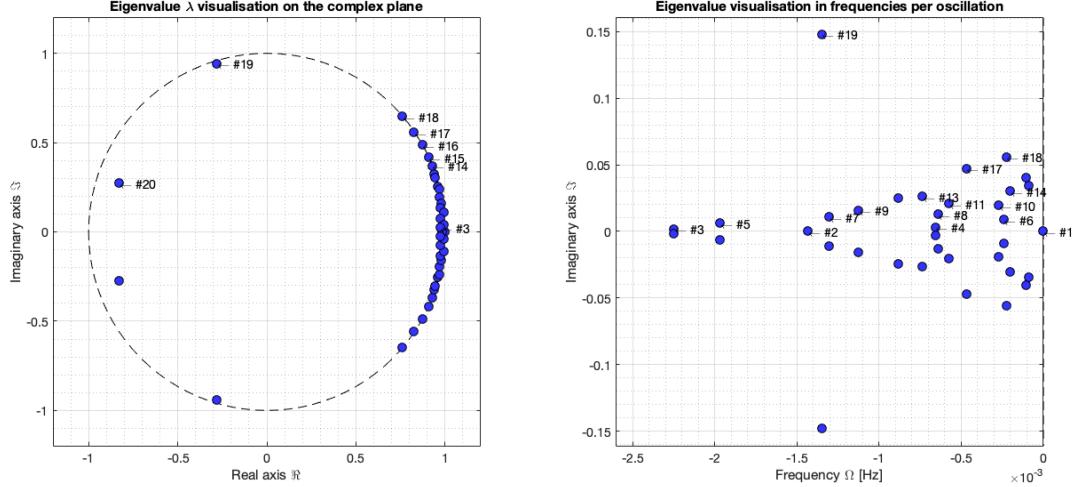


Figure A.12: DMD eigenvalues of the pitch model. On the left, the discrete time eigenvalues  $\lambda$  on the complex plane are represented relative to the unit circle (dashed line). On the right, the eigenvalues transformed to continuous time are represented. The area where the majority of modes are presented has been zoomed in.

The frequency response of the model can also be analysed by plotting the corresponding bode diagram. Figure A.13 shows the gain of the system for different frequencies, where the Nyquist frequency is sketched as a black line.

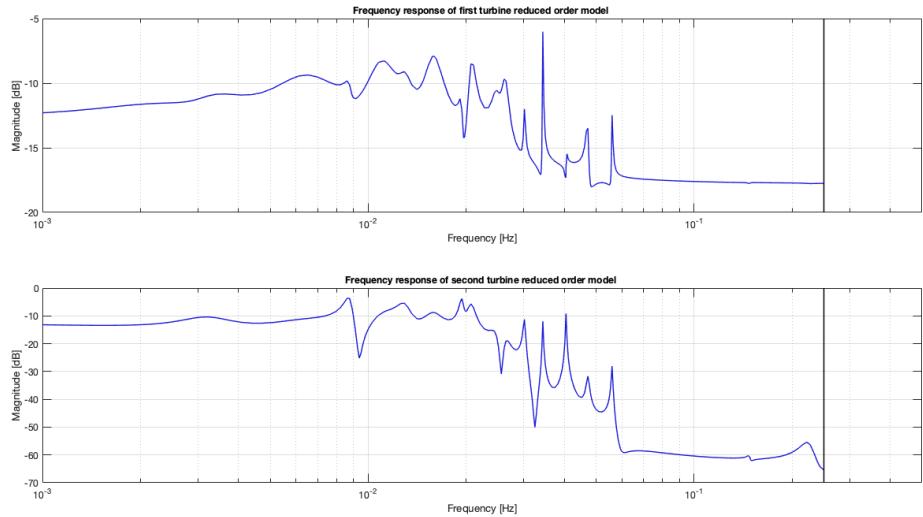


Figure A.13: Bode magnitude plot of LTI IODMD<sub>u</sub> model for axial induction control

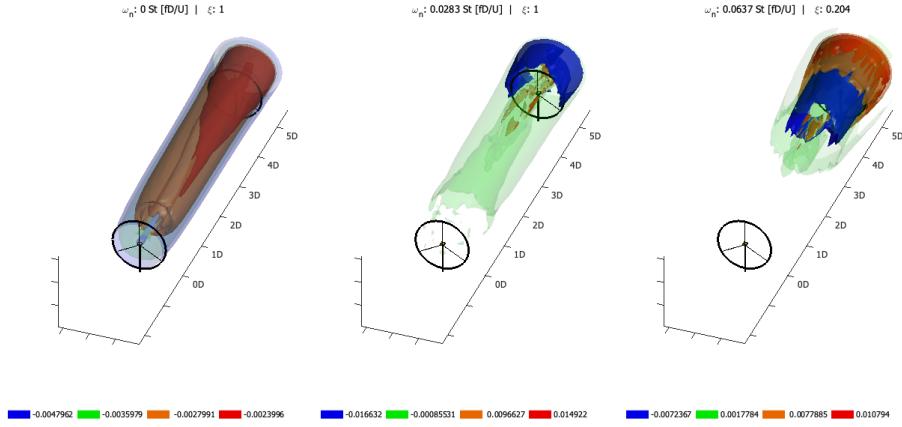


Figure A.14: DMD modes #1, #2 and #4. Mode #1 corresponds to a background mode that is not changing (i.e. it has zero eigenvalue). By analysing the isosurfaces, it is seen that this mode represents the deficit velocities always present behind turbine 1.

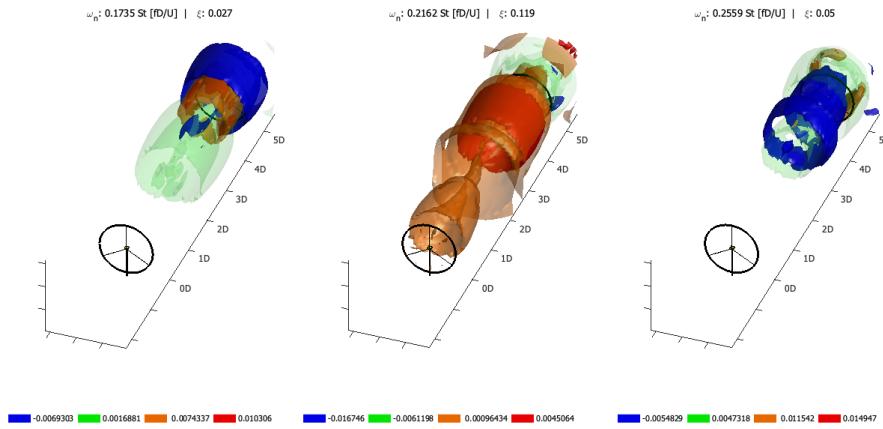


Figure A.15: DMD modes #6, #7 and #8, with natural frequencies comprised between 0.17 St and 0.26 St. The spatial patterns present here appear to also explain some of the dynamics more related with the downstream turbine, as opposite to background and low frequency modes.

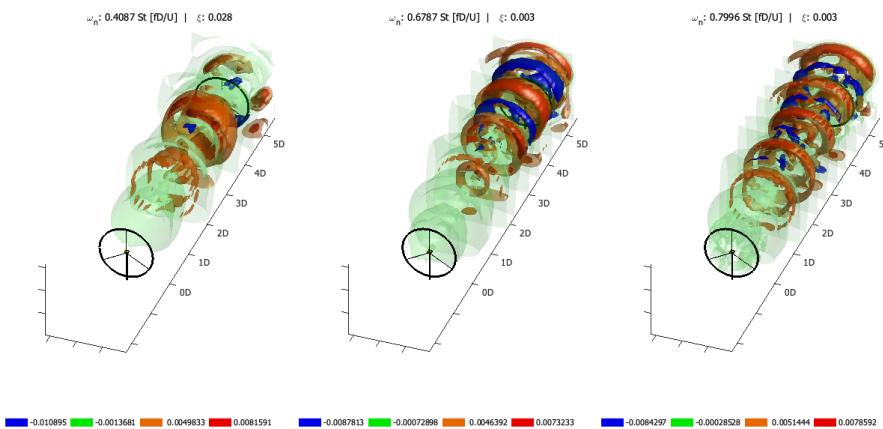


Figure A.16: DMD modes #11, #15 and #16, with natural frequencies comprised between 0.41 St and 0.80 St, showing pulsating structures inbetween upstream and downstream turbines.

**G. Other states are experimented with. It leads to the conclusion that for axial induction control  $w$  serves as a better candidate for the states.**

Several different states are attempted at first in alternative to using the  $u$  velocity component. The results are compiled in Table A.3.

Table A.3: Models validation results using IODMD algorithm with different observables. First column has the data chosen for the states. Second, third and forth columns have the fit criteria VAF and model size, where VAF of the second turbine generator power is the most relevant. The last two columns show the improved performance in terms of VAF (green) and the improvements in terms of the size of model. Reduced models are less computationally expensive for the purposes of model predictive control, hence they are written in green.

Model properties	Model obtained by $\text{IODMD}_u$			$\Delta \text{IODMD}_u$	
Observable	VAF(WT2) <sub>max</sub>	Model size	VAF(WT1)  <sub>r</sub>	$\Delta$ VAF (WT2) <sub>max</sub>	$\Delta$ size
$v$	74.78 %	39	97.64 %	+1.19%	+2.63%
$w$	81.71 %	39	96.99 %	+8.12%	+2.63%
$u'$	43.16 %	29	93.99 %	-30.43%	-23.68%
$v'$	43.09 %	49	96.71 %	-30.50%	+28.95%
$w'$	55.57 %	49	96.69 %	-18.02%	+28.95%
$u'^2$	51.83 %	57	95.58 %	-21.76%	+50.00%
$w'^2$	35.18 %	104	94.78 %	-38.41%	+41.32%
$u^2$	74.71 %	24	96.76 %	+1.12%	-36.84%
$v^2$	72.38 %	112	99.33 %	-1.21%	+194.74%
$w^2$	77.67 %	106	99.29 %	+4.08%	+178.95%
$u \cdot v$	75.56 %	59	99.03 %	+1.97%	+55.26%
$u \cdot w$	76.54 %	35	97.42 %	+2.95%	-7.89%
$v \cdot w$	75 %	120	99.25 %	+1.41%	+63.07%
$u \cdot v \cdot w$	72.83 %	72	99.25 %	-0.76%	+89.47%
$u^2 + w^2$	75.67 %	24	96.79%	+2.08%	-36.84%
$u^2 + v^2 + w^2$	78.65 %	24	96.89%	+5.09%	-36.84%

The attempt to find coordinates which can better reproduce the dynamics of the wind farm system, in light of the Koopman operator theory, appears to be fruitful, showing that gains in VAF of 8.13%, when compared to results yielded by  $\text{IODMD}_u$ . Choosing an observable that captures the complete fluid dynamics, by simultaneously including measurements of all velocity components, such as  $u^2 + w^2$  and  $u^2 + v^2 + w^2$  also appears to be advantageous, showing increases in VAF of 2.08% and 5.09% respectively, and decreases in size of -36.84% for both cases.

Leveraging the Koopman operator theory can also be achieved by EIODMD. Following the EIODMD algorithm, the base state  $u$  can be augmented with other observables. The results from this procedure are summarized in Table A.4

Table A.4: Models validation results using EIODMD algorithm with different observables. The second observables column has the state chosen to augment the initial data matrix with  $u$  information. The last two columns represent the average NRMSE of the first observable, using both identification and validation data.

Model properties		Model obtained by EIODMD			$\Delta \text{IODMD}_u$			
Observable	Second observables	VAF(WT2) <sub>max</sub>	Model size	VAF(WT1)  <sub>r</sub>	$\Delta \text{VAF}$ (WT2) <sub>max</sub>	$\Delta$ size	NRMSE (id)	NRMSE (val)
$u$	$v$	79.00 %	32	96.70 %	+5.41%	-15.79%	-0.05%	-0.07%
$u$	$w$	81.74 %	30	97.27 %	+8.15%	-21.05%	-0.12%	-0.33%
$u$	$v, w$	79.76 %	28	97.22 %	+6.17%	-2.19%	-0.08%	-0.13%

The results indicate that augmenting the data matrix with additional information can have benefits. The additional benefits of augmenting  $u$  with  $w$  information translates in an increase in VAF of 8.15% and a reduction in model size to of 21.05%. Furthermore, the mean NRMSE, which measures that ability of the model to reconstruct the full state information, is reduced to 3.54% and 3.42%, for identification and validation measurements.

It should be noted that this process can become computationally quite expensive, specially in the case of high order systems. By augmenting, the number of points increases by a factor of 2 from 43.907 to 87.814 data points. Consequently, the SVD is performed on a matrix with a total of almost 80 million entries, as opposed to 40.

A comparison of both model's performance is depicted in Figure A.17. It is interesting to note how the EIODMD <sub>$u,w$</sub>  better captures the spikes in power as a consequence of the blades pitching back to zero degrees.

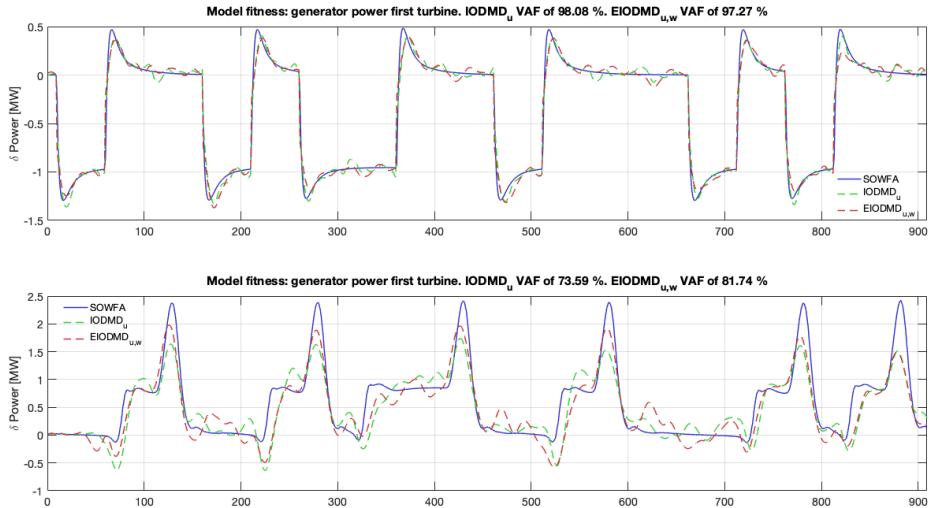


Figure A.17: Comparison of IODMD <sub>$u$</sub>  and EIODMD <sub>$u,w$</sub>  model's performance.

Both models can also be compared based on their response in the frequency domain. Another result of interest is the resonance frequency  $\omega_n$  at approximate 0.2 St which is captured and can be observed in the bode magnitude plot in Figure A.18. The system's response at this frequency shows a magnitude of 10.43 dB, higher than any other frequency.

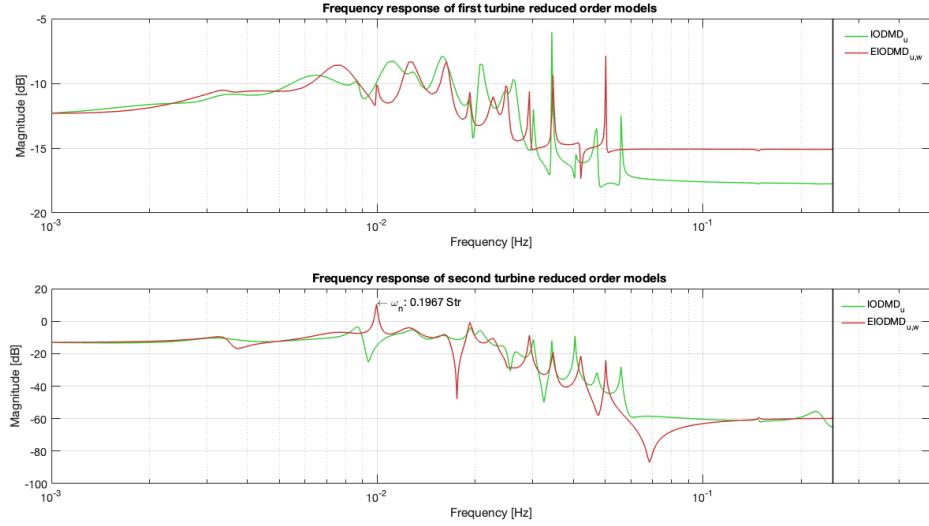


Figure A.18: Comparison of  $\text{IODMD}_u$  and  $\text{EIODMD}_{u,w}$  model's response in the frequency domain.

In addition to augmenting the initial data matrices with flow field, other information related with turbine performance can be experimented with. The results are summarized in Table A.5.

Table A.5: Models validation results using EIODMD algorithm with different base observable and deterministic states

Model properties		Model obtained by EIODMD				$\Delta \text{IODMD}_u$		
Observable	Deterministic	VAF(WT2) <sub>max</sub>	Model size	VAF(WT1)  <sub>r</sub>	$\Delta \text{VAF}(\text{WT2})_{\text{max}}$	$\Delta \text{size}$	$\overline{\text{NRMSE}}(\text{id})$	$\overline{\text{NRMSE}}(\text{val})$
$u$	$\Omega_1, \Omega_2$	76.71 %	12	98.97 %	+3.12%	-68.42%	+0.17%	+0.45%
$w$	$\Omega_1, \Omega_2$	76.79 %	30	98.93 %	+3.20%	-21.05%	-	-

The results point to the conclusion that augmenting data matrices with turbine information is a trade-off. For the purposes of prediction and control, it does provide models which are better able to capture the generator power dynamics. This procedure does not incur on an extra computational cost, as only few lines of data are added to the snapshot matrices. Nevertheless, increases in mean NRMSE were registered, to 3.83% and 4.20% for identification and validation, respectively.

## APPENDIX B

# MATHEMATICAL OPERATORS

---

## B.1 Linear Algebra

### $l_2$ norm

The  $l_2$  norm is a standard method to compute the length of a vector,  $\mathbf{x} = \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix}^T$ , and is defined as the square root of the sum of the squares of the values in each dimension:

$$|\mathbf{x}| = \sqrt{\sum_{k=1}^n |x_k|^2} \quad (\text{B.1})$$

### Frobenius norm

The Frobenius norm, or Hilbert-Schmidt norm, is a matrix norm defined as the square root of the sum of the absolute square os its elements [29]:

$$\|\mathbf{X}\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2} \quad (\text{B.2})$$

### Singular Value Decomposition

The Singular Value Decomposition (SVD) is a matrix factorization whose computation is a step in many algorithms. The SVD of  $\mathbf{A}$  is a factorization:

$$\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^* \quad (\text{B.3})$$

where

$$\begin{aligned} \mathbf{U} &\in \mathbb{C}^{n \times n} \text{ is unitary} \\ \mathbf{V} &\in \mathbb{C}^{m \times m} \text{ is unitary} \\ \Sigma &\in \mathbb{C}^{n \times m} \text{ is diagonal} \end{aligned} \quad (\text{B.4})$$

$\mathbf{U}$  and  $\mathbf{V}$  are unitary matrices<sup>1</sup> such that  $\mathbf{U}^*\mathbf{U} = \mathbf{U}\mathbf{U}^* = \mathbf{I}$  and  $\mathbf{V}^*\mathbf{V} = \mathbf{V}\mathbf{V}^* = \mathbf{I}$ . The matrix  $\Sigma$  has entries along the diagonal corresponding to the singular values ordered from largest to smallest, which are

<sup>1</sup>A complex square matrix is unitary if its conjugate transpose is also its inverse.

assumed to be non-negative. The columns in  $\mathbf{U}$  are referred to as the left singular vectors and the columns in  $\mathbf{V}$  as the right singular vectors.

This procedure produces a hierarchical matrix decomposition which spits a matrix into a sum of rank-one matrices given by the outer product of a column vector (left singular vector) with a row vector (conjugate transpose of right singular vector). These rank-one matrices are ordered by the singular value so that the first  $r$  rank-one matrices form the *best* rank- $r$  matrix approximation of the original matrix in a least squares sense. For more information on the SVD, as well as the geometric interpretation, please refer to [66].

## B.2 Vector Calculus

### Gradient

The gradient vector function of a function  $f : \in \mathbb{R}^n \rightarrow \mathbb{R}$  at a point  $x \in D_f$  is defined as [48]:

$$\text{grad } f(\mathbf{x}) = \left( \frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \dots, \frac{\partial f}{\partial x_n} \right) \equiv \nabla f(\mathbf{x}) \quad (\text{B.5})$$

### Divergence

Divergence is a differential operator applied to a three dimensional vector-valued function  $\mathbf{F} = F_x \mathbf{i} + F_y \mathbf{j} + F_z \mathbf{k}$ .

The divergence of a vector  $\mathbf{v}$ , in Cartesian coordinates  $x, y, z$  is given by:

$$\nabla \cdot \mathbf{v} = \frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z} \quad (\text{B.6})$$

# APPENDIX C

## MAIN ALGORITHMS

---

### Algorithm 1 *dynamicmodedecomposition.m*

---

#### Input:

*states*: states to be used in Singular Value Decomposition (SVD)  
*Inputs*: inputs used for model identification  
*Outputs*: outputs used for model identification  
*Deterministic*: deterministic used for model identification  
*method*: variant of dynamic mode decomposition to use  
*r*: truncation value for the SVD  
*maindir*: main directory to save results  
*dt*: sampling period, in seconds

#### Output:

*sys\_red*: #*r* reduced order models (rom) computed by dmd variant  
*sys\_red.fil*: #*r* roms computed by dmd variant where actuator is modelled as low pass filter  
*FITje*: VAF of models in *sys\_red* using identification data  
*FITje.fil*: VAF of models in *sys\_red.fil* using identification data  
*U,S,V*: Matrices resulting from SVD  
*X, X\_p*: Snapshot matrices used for DMD  
*X\_d*: Deterministic states  
*dirdmd*: Subdirectory within *miandir* where results are saved  
*x*: States predicted by models using identification data

```

1:  $X \leftarrow states(:, 1 : end - 1)$                                      ▷ Generate snapshot matrix
2:  $X' \leftarrow states(:, 2 : end)$                                     ▷ Generate time shifted snapshot matrix
3:  $Y \leftarrow Outputs(:, 1 : end - 1)$                                 ▷ Generate output data matrix
4:  $Y' \leftarrow Outputs(:, 2 : end)$                                 ▷ Generate time shifted output data matrix
5:  $U \leftarrow Inputs(:, 1 : end - 1)$                                 ▷ Generate input matrix
6:  $U' \leftarrow Inputs(:, 2 : end)$                                 ▷ Generate time shifted input data matrix
7:  $D \leftarrow Deterministic(:, 1 : end - 1)$                             ▷ Generate deterministic data matrix
8:  $D' \leftarrow Deterministic(:, 2 : end)$                             ▷ Generate time shifted deterministic data matrix
9:  $lpf \leftarrow ss(1/dt, 1, 1/dt, 0, dt)$                          ▷ Generate state space low pass filter to model actuator
10:
11: if method=0 then
12:   maindir  $\leftarrow DMDresults.DMD$                                 ▷ Standard DMD: no control action
13:    $U, S, V \leftarrow SVD(X, r)$                                          ▷ Create directory for standard DMD
14:   for  $si = 1 \rightarrow r$  do
15:      $\tilde{U} \leftarrow U(:, 1 : si)$                                          ▷ SVD of snapshot matrix truncated at r singular values
16:      $\tilde{S} \leftarrow S(1 : si, 1 : si)$ 
  
```

---

---

```

17:    $\tilde{V} \leftarrow V(:, 1 : si)$ 
18:    $A \leftarrow \tilde{U}X'\tilde{V}\tilde{S}^{-1}$ 
19:    $sys\_red(si) \leftarrow ss(A, \mathbf{0}, \mathbf{0}, \mathbf{0}, dt)$            ▷ Create state space model with only matrix A
20: end for
21: else if method=1 then                                ▷ DMDc: control action and no mapping to specific output
22:    $\Omega \leftarrow \begin{bmatrix} X \\ U \end{bmatrix}$ 
23:    $U, S, V \leftarrow SVD(\Omega, r)$ 
24:    $U_o, S_o, V_o \leftarrow SVD(X', r)$ 
25:    $n \leftarrow size(X, 1)$                                 ▷ Designate size of states (row-space)
26:    $q \leftarrow size(U, 1)$                                 ▷ Designate size of inputs (row-space)
27:   for  $si = 1 \rightarrow r$  do
28:      $\tilde{U} \leftarrow U(:, 1 : si)$ 
29:      $\tilde{S} \leftarrow S(1 : si, 1 : si)$ 
30:      $\tilde{V} \leftarrow V(:, 1 : si)$ 
31:      $\tilde{U}_1 \leftarrow U(1 : n, :)$ 
32:      $\tilde{U}_2 \leftarrow U(n + 1 : n + q, :)$ 
33:      $\hat{U} \leftarrow U_o(:, 1 : si)$ 
34:      $A \leftarrow \hat{U}X'\tilde{V}\tilde{S}^{-1}\tilde{U}_1^*\hat{U}$ 
35:      $B \leftarrow \hat{U}X'\tilde{V}\tilde{S}^{-1}\tilde{U}_2$ 
36:      $sys\_red(si) \leftarrow ss(A, B, \mathbf{I}, \mathbf{0}, dt)$ 
37:   end for
38:
39: else if method=2 then                                ▷ IODMD: control action and mapping to specific output
40:    $maindir \leftarrow DMDresults\_IODMD$                   ▷ Create directory for IODMD
41:    $U, S, V \leftarrow SVD(X, r)$                           ▷ SVD of snapshot matrix truncated at r singular values
42:   for  $si = 1 \rightarrow r$  do
43:      $\tilde{U} \leftarrow U(:, 1 : si)$ 
44:      $\tilde{S} \leftarrow S(1 : si, 1 : si)$ 
45:      $\tilde{V} \leftarrow V(:, 1 : si)$ 
46:      $\begin{bmatrix} A & B \\ C & D \end{bmatrix} \leftarrow \begin{bmatrix} \tilde{U}^*X' \\ Y \end{bmatrix} \begin{bmatrix} \tilde{S}\tilde{V} \\ U \end{bmatrix}^\dagger$           ▷ Compute least squares approximation
47:      $sys\_red(si) \leftarrow ss(A, B, C, D, dt)$           ▷ Create state space model A,B,C,D
48:     Evaluate fitness of models  $sys\_red(si)$ 
49:     Save results of models performance
50:   end for
51:    $Xd = \{\}$                                          ▷ Create empty variable for deterministic states, as they were not used
52:
53: else if method=3 then                                ▷ EIODMD: control action and improved mapping based on Koopman
54:    $maindir \leftarrow DMDresults\_EIODMD$                   ▷ Create directory for EIODMD
55:    $U, S, V \leftarrow SVD(X, r)$ 
56:   for  $si = 1 \rightarrow r$  do
57:      $\tilde{U} \leftarrow U(:, 1 : si)$ 
58:      $\tilde{S} \leftarrow S(1 : si, 1 : si)$ 
59:      $\tilde{V} \leftarrow V(:, 1 : si)$ 
60:      $\begin{bmatrix} A & B \\ C & D \end{bmatrix} \leftarrow \begin{bmatrix} D' \\ \tilde{U}^*X' \\ Y \end{bmatrix} \begin{bmatrix} D \\ \tilde{S}\tilde{V} \\ U \end{bmatrix}^\dagger$           ▷ Compute least squares approximation
61:      $sys\_red(si) \leftarrow ss(A, B, C, D, dt)$           ▷ Create state space model A,B,C,D
62:     Evaluate fitness of models  $sys\_red(si)$ 
63:     Save results of models performance
64:   end for
65: end if

```

---

---

**Algorithm 2** powerreferencetracking.m

---

**Input:**

*model*: Model to be used for predictive control  
*H<sub>p</sub>, H<sub>c</sub>*: Prediction and control horizon, respectively  
*Inputs, Outputs*: Simulation data  
*scalingfactors*: Scalar factors used to scale data

**Output:**

*U*: Optimal control action

```
1: Setup controller: Define prediction horizon  $H_p$  and control horizon  $H_c$ .  
2: Setup model: Define state space matrices ( $A, B, C, D$ ) of model to be used.  
3: Setup incremental model: Define state space matrices ( $A_e, B_e, C_e, D_e$ ) of model to be used.  
4: Construct matrices: Define matrices  $\bar{H}_e, F$  and  $\Gamma_e$ .  
5:  $t \leftarrow [t_i, t_i + dt, \dots, t_f + H_p]$  ▷ Define vector from initial time  $t_i$  to final instant  $t_f + H_p$   
6:  $P_{ref} \leftarrow [P_1(\Delta t); P_2(\Delta t); \dots; P_m(\Delta t + H_p)]$  ▷ Define reference for wind farm total power  
7:  $x_o \leftarrow dinit(A, B, C, D, Inputs, Outputs)$  ▷ Compute initial states using identification data:  
8:  $x(1) \leftarrow x_o$   
9:  $t \leftarrow 0$   
10:  $Q \leftarrow$  Diagonal matrix of size  $H_p \times H_p$  with equal entries of 10.  
11:  $R \leftarrow$  Identity matrix of size  $H_c \times H_c$   
12:  $\mathcal{H} \leftarrow 2(\bar{H}_e^T F^T \bar{H}_e + R)$  ▷ Compute Hessian matrix for QP problem formulation  
13:  $u_{min} \leftarrow -25$  ▷ Define lower limit for control action  
14:  $u_{max} \leftarrow 10$  ▷ Define upper limit for control action  
15: Construct matrices: Define matrices  $M$  and  $\Lambda$  ▷ QP restrictions matrix  
16:  $u_0 \leftarrow 0$   
17: for  $k = 1 \rightarrow m - H_p$  do  
18:    $\tilde{P} \leftarrow []$   
19:   for  $v = 1 \rightarrow H_p$  do  
20:      $\tilde{P} = [\tilde{P}; P_{ref}(k + v)]$  ▷ Compute reference at time instant  $k$   
21:   end for  
22:    $C = -F\Gamma_e x_e(k) - \tilde{P}$  ▷ Compute constant term for every time step  
23:    $f = -2(\bar{H}_e^T F^T Q C)$   
24:    $\Delta u_{opt} \leftarrow quadprog(\mathcal{H}, f, M, \Lambda)$  ▷ Compute optimal future control sequence  
25:    $\Delta u(k) \leftarrow \Delta u_{opt}(1)$  ▷ Designate next incremental control action  
26:    $u(k) \leftarrow u(k - 1) + \Delta u(k)$   
27:    $x_e(k + 1) \leftarrow A_e x_e(k) + B_e \Delta u(k)$   
28:    $P(k) \leftarrow C_e x_e(k)$   
29: end for
```

---

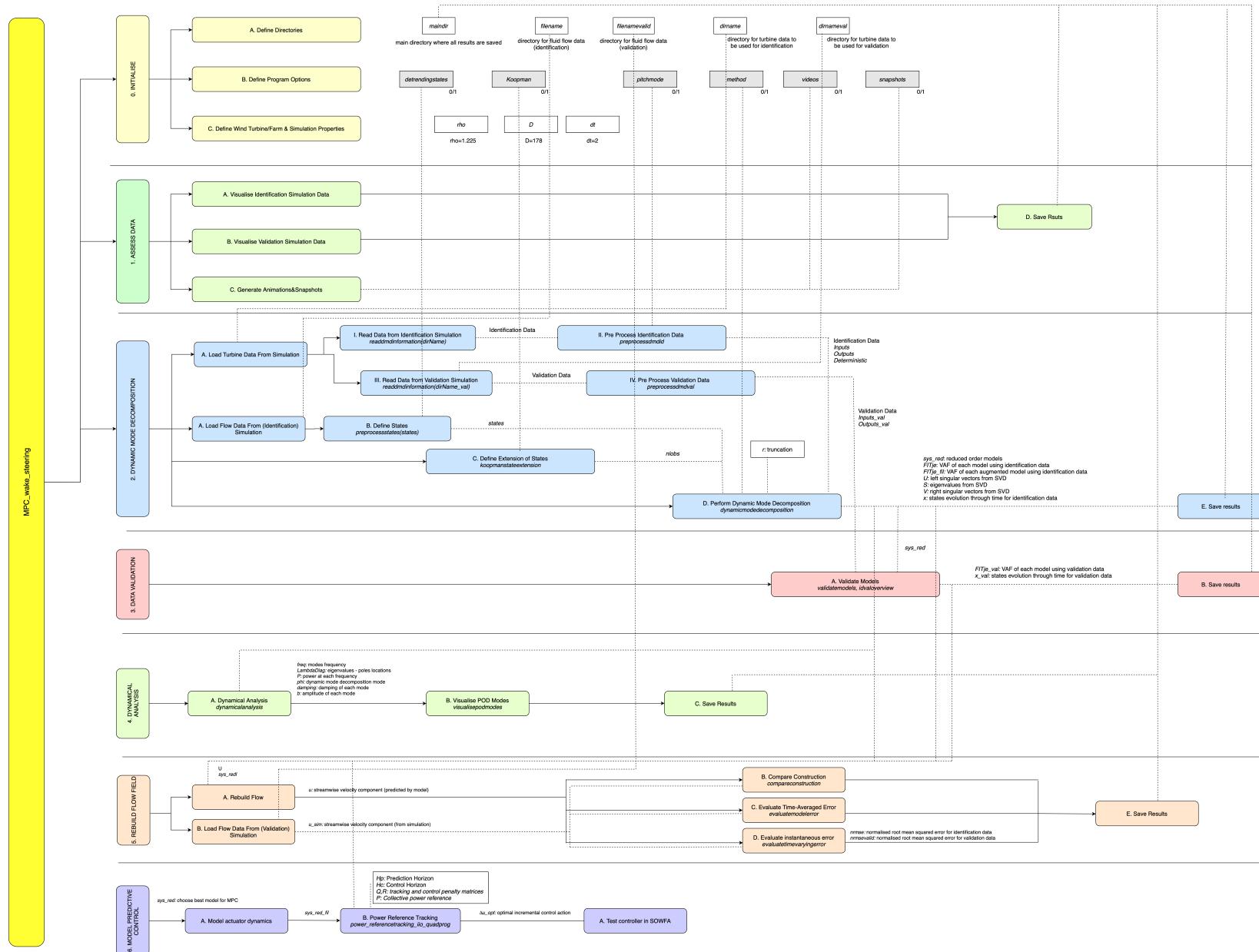


Figure C.1: MALTAB program structure