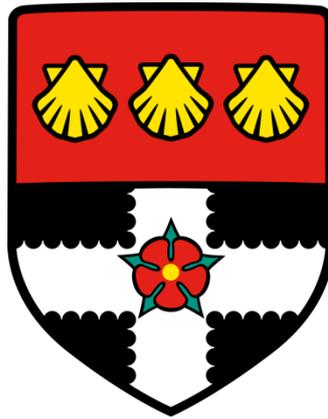


**UNIVERSITY OF READING**

**Fractional-Order System Identification in Massive  
MIMO Systems**



Thesis submitted in partial fulfilment of the requirements for the degree of  
Doctor of Philosophy

Department of Biomedical Engineering,  
School of Biological Sciences

Mzabalazo Lupupa  
December 2018

# **Declaration**

I confirm that this is my own work and the use of all material from other sources has been properly and fully acknowledged.

Mzabalazo Lupupa

21<sup>st</sup> December 2018

# **Dedication**

To my family, relatives and friends.

## **Acknowledgements**

My sincere appreciation goes to my supervisors Dr. Sillas Hadjiloucas and Dr. John Bowen and my former supervisors, Prof. Fu-Chun Zheng and Dr. Jian Qi for their guidance.

I would like to express my gratitude to my mother, Mrs. T. J. Lupupa for her financial support and encouragement. I would also like to thank my entire family, especially my grandmother, Mrs. E.T. Lupupa and my friends all over the world for their support and prayers. To my late father, Mr. B. M. Lupupa, thank you for believing in the power of education, "*Kuliwa ngempenseli*".

I am also grateful to my colleagues, flatmates and friends I met whilst in Reading for their advice, and support. Most of all, thank you Lord, God Almighty!

# Abstract

Fifth generation (5G) wireless communication systems promise increased capacity, increased data rate, enhanced reliability, reduced latency, improved energy efficiency, improved spectrum efficiency and reduced interference, and massive multiple-input multiple-output (MIMO) has been identified as a driving technique in achieving this. In massive MIMO, the base station is equipped with hundreds of antennas to service tens of terminals in the same time-frequency resource. But there are several challenges associated with massive MIMO that prevent the achievement of these benefits, and these include channel estimation, pilot contamination and radio frequency (RF) impairments, etc. The main focus of this thesis is on the use of continuous-time state-space models to identify the dynamics of massive MIMO wireless channels, i.e. channel estimation.

Two identification models, namely the continuous-time integer-order state-space and continuous-time fractional-order state-space identification models are considered when identifying the massive MIMO frequency-selective wireless channels. These models are designed based on the multiple-input multiple-output output-error state space (MOESP) algorithm, a subspace system identification algorithm that has been proven to successfully identify the dynamics of a system.

Through simulations it is shown that with increase in model order, the continuous-time integer-order state-space model is able to model the massive MIMO channels with increased accuracy. The performance of the continuous-time fractional-order state-space model is also studied for different fractional-order values, and its performance is then compared with that of the continuous-time integer-order state-space model.

Having identified the dynamics of the massive MIMO system, equalizers are then designed to help combat the effects of inter-symbol interference (ISI) caused by the massive multiple-input multiple-output (MIMO) frequency-selective wireless channels. We propose the use of state-space models for channel equalization. The minimum mean square error – decision feedback equalizer (MMSE-DFE) is the equalizer of choice in

addressing the ISI and is built based on the continuous-time integer-order state-space and continuous-time fractional-order state-space identified models.

# Contents

Declaration.....	i
Dedication.....	ii
Acknowledgements.....	iii
Abstract.....	iv
List of Figures.....	x
List of Tables.....	xiv
Abbreviations.....	xv
Chapter 1.....	1
Introduction.....	1
1.1 Motivation.....	1
1.2 System Identification.....	2
1.3 Massive MIMO.....	6
1.3.1 Channel Estimation.....	9
1.3.2 Pilot Contamination.....	11
1.3.3 Radio Frequency Impairments.....	14
1.4 Research Contributions.....	27
1.5 List of Publications.....	28
1.6 Thesis Outline.....	28
Chapter 2.....	31
Massive MIMO Continuous-Time System Identification Using Subspace Algorithm..	31
2.1 Introduction.....	31
2.2 Massive MIMO System Model.....	32
2.3 State-Space Model.....	34
2.4 Subspace Identification Methods.....	36
2.4.1 The Algebraic Geometric Framework.....	38
2.4.2 The Multiple-Input Multiple-Output Output-Error State Space Algorithm...	43
2.4.3 The Numerical Algorithm for Subspace State-Space System Identification .	46
2.4.4 The Canonical Variate Analysis Algorithm.....	51

2.4.5 Constructing Data Matrices .....	52
2.5 Integer-Order System Identification with Measurement Noise Considered .....	56
2.6 Simulation Results .....	63
2.6.1 Selection of Identification Parameters.....	67
2.7 Summary .....	73
Appendix A2.....	75
A2.1 The Poisson Moment Functional Filtering for Continuous-Time Integer-Order System Identification .....	75
Chapter 3.....	79
Massive MIMO Continuous-Time System Identification Using Fractional-Order Algorithm.....	79
3.1 Introduction .....	79
3.2 Fractional-Order System Identification.....	81
3.2.1 Constructing Data Matrices .....	84
3.2.2 Estimating the Extended Observability Matrix .....	87
3.2.3 Analytical Solution of the Fractional-Order Massive MIMO System .....	88
3.3 Fractional-Order Realisations / Integer-Order Approximation.....	92
3.3.1 Carlson’s Realisation.....	94
3.3.2 Charef’s Realisation .....	94
3.3.3 Oustaloup’s Realisation.....	96
3.4 Fractional-Order System Identification with Measurement Noise Considered ....	97
3.5 Simulation Results .....	103
3.5.1 Selection of Identification Parameters.....	106
3.6 Summary .....	113
Appendix A3.....	115
A3.1 Special Functions .....	115
A3.1.1 Gamma Function .....	115
A3.1.2 Mittag-Leffler Function.....	115
A3.2 Properties of Fractional-Order Calculus .....	116
A3.3 Definitions of Fractional-Order Calculus.....	118
A3.3.1 The Grünwald-Letnikov Definition.....	119

A3.3.2 The Riemann-Liouville Definition .....	119
A3.3.3 The Caputo Definition .....	120
A3.4 The Poisson Moment Functional Filtering for Continuous-Time Fractional- Order System Identification .....	121
Chapter 4.....	124
Massive MIMO Channel Equalization Using Integer-Order State-Space Models.....	124
4.1 Introduction .....	124
4.2 Inter-Symbol Interference .....	125
4.3 Channel Equalization .....	127
4.3.1 Matched Filter Equalizer .....	129
4.3.2 Zero Forcing Equalizer.....	130
4.3.3 Minimum Mean Square Error Equalizer .....	131
4.3.4. Decision Feedback Equalizer .....	135
4.3.5 Maximum Likelihood Sequence Estimation Equalizer.....	139
4.4 Minimum Mean Square Error – Decision Feedback Equalizer for the Finite Impulse Response Massive MIMO Channel Model .....	140
4.5 Minimum Mean Square Error – Decision Feedback Equalizer for the Integer- Order State-Space Massive MIMO Channel Model .....	145
4.6 Simulation Results .....	153
4.7 Summary .....	157
Chapter 5.....	158
Massive MIMO Channel Equalization Using Fractional-Order Models.....	158
5.1 Introduction .....	158
5.2 Minimum Mean Square Error – Decision Feedback Equalizer for the Fractional- Order Channel Model.....	158
5.3 Symbol Error Rate Performance .....	166
5.4 Simulation Results .....	169
5.5 Summary .....	174
Chapter 6.....	176
Conclusions and Future Work .....	176
6.1 Conclusions .....	176

6.2 Future Work .....	180
References.....	182
Appendix A.....	201
Nonlinear Models.....	201
Appendix B .....	205
The General-Linear Polynomial Models.....	205
Appendix C .....	210
Definition of a Persistently Exciting Signal.....	210

# List of Figures

Figure 1. 1: Possible antenna configurations for a massive MIMO base station.....	7
Figure 1. 2: Time division duplexing.....	10
Figure 1. 3: Frequency division duplexing .....	10
Figure 1. 4: Pilot contamination concept, (a) Uplink transmission (b) Downlink transmission .....	12
Figure 1. 5: Direct up-conversion .....	15
Figure 1. 6: Block diagram of a direct up-conversion transmitter .....	15
Figure 1. 7: Direct down-conversion .....	16
Figure 1. 8: Block diagram of a direct down-conversion receiver .....	16
Figure 1. 9: The influence of IQ imbalance on the message (wanted) signal.....	18
Figure 1. 10: SER vs. SNR for ideal received signal, received signal with IQ imbalance and received signal with IQ imbalance compensation for BPSK signal .....	23
Figure 1. 11: SER vs. SNR for ideal received signal, received signal with IQ imbalance and received signal with IQ imbalance compensation for QPSK modulated signal .....	24
Figure 1. 12: SER vs. SNR for ideal received signal, received signal with IQ imbalance and received signal with IQ imbalance compensation for 256 QAM modulated signal .....	24
Figure 2. 1: Block diagram of a massive MIMO system.....	33
Figure 2. 2: Stability region for integer-order systems.....	35
Figure 2. 3: Block diagram of a state-space model.....	36
Figure 2. 4: Illustrating skew-diagonal entries .....	38
Figure 2. 5: Orthogonal projection .....	39
Figure 2. 6: Oblique projection.....	41
Figure 2. 7: Chirp signal for the input .....	64
Figure 2. 8: Plot of input and output data .....	65
Figure 2. 9: System order obtained using the MATLAB's System Identification Toolbox.....	66
Figure 2. 10: Comparison of estimation and validation data sets for (system order, $n = 7$ ).....	68

Figure 2. 11: Comparing the performance of the system output obtained using the integer-order MOESP algorithm for $n = 7$ and the actual system output (zoomed in results).....	69
Figure 2. 12: Comparing the performance of the system output obtained using the integer-order MOESP algorithm for $n = 3$ and the actual system output .....	70
Figure 2. 13: Comparing the performance of the system output obtained using the integer-order PO-MOESP algorithm for $n = 7$ and the actual system output .....	71
Figure 2. 14: Comparing the performance of the system output obtained using the integer-order PO-MOESP algorithm for $n = 7$ and the actual system output (zoomed in results).....	72
Figure 2. 15: Comparing the performance of the system output obtained using the integer-order MOESP algorithm, integer-order PO-MOESP algorithm for $n = 7$ and the actual system output (zoomed in results).....	73
Figure 3. 1: Stability region for fractional-order commensurate systems .....	83
Figure 3. 2: Chirp signal for the input .....	104
Figure 3. 3: Plot of input data .....	105
Figure 3. 4: Comparing the performance of the system output obtained using the fractional-order MOESP algorithm for $\alpha = 0.1$ and the actual system output .....	107
Figure 3. 5: Comparing the performance of the system output obtained using the fractional-order MOESP algorithm for $\alpha = 0.1$ and the actual system output (zoomed in results).....	108
Figure 3. 6: Comparing the performance of the system output obtained using the fractional-order PO-MOESP algorithm for $\alpha = 0.1$ and the actual system output .....	109
Figure 3. 7: Comparing the performance of the system output obtained using the fractional-order PO-MOESP algorithm for $\alpha = 0.1$ and the actual system output (zoomed in results).....	110
Figure 3. 8: Comparing the performance of the system output obtained using the fractional-order MOESP algorithm, fractional-order PO-MOESP algorithm for $\alpha = 0.1$ and the actual system output (zoomed in results) .....	110
Figure 3. 9: Comparing the performance of the system output obtained using the Oustaloup's realisation for $\alpha = 0.1$ and the actual system output .....	112

Figure 3. 10: Comparing the performance of the system output obtained using the Oustaloup's realisation for $\alpha = 0.1$ and the actual system output .....	112
Figure 4. 1: Ideal transmission.....	125
Figure 4. 2: Real transmission .....	126
Figure 4. 3: Effects of inter-symbol interference (ISI) .....	126
Figure 4. 4: Block diagram of a multipath fading channel .....	127
Figure 4. 5: Summary of linear and nonlinear equalizers .....	128
Figure 4. 6: Block diagram for linear equalizers .....	129
Figure 4. 7: Block diagram for DFE .....	135
Figure 4. 8: DFE for FIR channel models .....	141
Figure 4. 9: DFE for state-space channel models .....	148
Figure 4. 10: Plot of the input signal, MMSE-DFE signal for the FIR massive MIMO channel model and MMSE-DFE signal for the integer-order state-space massive MIMO channel model in each transmitting antenna.....	154
Figure 4. 11: Plot of the input signal, MMSE-DFE signal for the FIR massive MIMO channel model and MMSE-DFE signal for the integer-order state-space massive MIMO channel model in each transmitting antenna (zoomed in) .....	155
Figure 5. 1: DFE for fractional-order channel models.....	161
Figure 5. 2: Plot of the input signal and MMSE-DFE signal for the fractional-order massive MIMO channel model in each transmitting antenna.....	170
Figure 5. 3: Plot of the input signal and MMSE-DFE signal for the fractional-order massive MIMO channel model in each transmitting antenna (zoomed in) .....	171
Figure 5. 4: Plot of SER vs. SNR for the MMSE-DFE signal for the FIR, integer-order and fractional-order massive MIMO channel models for BPSK modulated signals....	173
Figure 5. 5: Plot of SER vs. SNR for the MMSE-DFE signal for the FIR, integer-order and fractional-order massive MIMO channel models for QPSK modulated signals ...	173
Figure 5. 6: Plot of SER vs. SNR for the MMSE-DFE signal for the FIR, integer-order and fractional-order massive MIMO channel models for 256QAM modulated signals .....	174
Figure A. 1: Hammerstein model .....	202

Figure A. 2: Wiener model .....	202
Figure A. 3: The Wiener-Hammerstein model .....	203
Figure A. 4: Neural networks model .....	204
Figure B. 1: Structure of a general-linear model .....	205
Figure B. 2: Structure of an autoregressive model .....	206
Figure B. 3: Structure of an autoregressive with exogenous input model .....	207
Figure B. 4: Structure of an autoregressive moving average with exogenous input model .....	208
Figure B. 5: Structure of a Box-Jenkins model .....	208
Figure B. 6: Structure of an output-error model .....	209

## List of Tables

Table 2. 1: System model and system identification parameters for integer-order model .....	67
Table 2. 2: MSE for system orders $n = 3$ and $n = 7$ .....	71
Table 3. 1: System model and system identification parameters for fractional-order model .....	106
Table 3. 2: Simulation parameters for the Oustaloup's realisation.....	111
Table 4. 1: System model and channel equalization parameters for integer-order model .....	153
Table 4. 2: MSE for $N_f = 6$ , $\Delta = 5$ and different $N_b$ .....	156
Table 4. 3: MSE for $N_b = 3$ , $\Delta = 5$ and different $N_f$ .....	156
Table 4. 4: MSE for $N_f = 6$ , $N_b = 3$ and different $\Delta$ .....	156
Table 5. 1: System model and channel equalization parameters for fractional-order model .....	170
Table 5. 2: MSE for $N_f = 6$ , $\Delta = 5$ and different $N_b$ .....	172
Table 5. 3: MSE for $N_b = 3$ , $\Delta = 5$ and different $N_f$ .....	172
Table 5. 4: MSE for $N_f = 6$ , $N_b = 3$ and different $\Delta$ .....	172

# Abbreviations

Multiple-input multiple-output	MIMO
Radio frequency	RF
Multiple-input multiple-output output-error state space or Multivariable output-error state space	MOESP
Inter-symbol interference	ISI
Minimum mean square error – decision feedback equalizer	MMSE-DFE
Nonlinear autoregressive moving average with exogenous input	NARMAX
Neural networks	NN
Autoregressive	AR
Autoregressive with exogenous input	ARX
Autoregressive moving average with exogenous input	ARMAX
Box-Jenkins	BJ
Output-error	OE
Finite impulse response	FIR
Discrete Fourier Transform	DFT

Maximum-ratio combining	MRC
Zero forcing	ZF
Channel state information	CSI
Fifth generation	5G
Maximum-ratio transmission	MRT
Signal-to-noise ratio	SNR
Time division duplexing	TDD
Frequency division duplexing	FDD
User equipment	UE
Minimum mean square error	MMSE
In-phase and quadrature	IQ
Intermediate frequency	IF
Digital-to-analogue-converto	DAC
Low pass filter	LPF
Power amplifier	PA
Bandpass filter	BPF
Low-noise amplifier	LNA

Decibels	dB
Image rejection ratio	IRR
Image leakage ratio	ILR
Local oscillator	LO
Common phase error	CPE
Inter-carrier interference	ICI
Orthogonal frequency-division multiplexing	OFDM
Peak-to-average-power ratio	PAPR
Single-carrier frequency-division multiple access	SC-FDMA
Linear-time invariant	LTI
Poisson moment functional	PMF
Subspace system identification	SSI
Numerical algorithm for subspace state-space system identification	N4SID
Canonical variate analysis	CVA
Linear quadratic	LQ
Singular value decomposition	SVD

Past output multiple-input multiple-output output-error state space PO-MOESP

Mean-square error MSE

Grünwald-Letnikov G-L

Riemann-Liouville R-L

Caputo C

Matched filter MF

Maximum likelihood sequence estimation MLSE

Decision feedback equalizer DFE

Maximum likelihood ML

Maximum á posteriori MAP

Symbol error rate SER

Quadrature phase shift keying QPSK

Binary phase shift keying BPSK

Quadrature amplitude modulation QAM

# Chapter 1

## Introduction

### 1.1 Motivation

With advances in wireless communications there is need for systems that will address the issues associated with capacity, data rates, reliability, latency, energy and spectrum efficiency. Massive multiple-input multiple-output (MIMO), an emerging technology that uses antenna arrays which have hundreds of antennas at the base station to service tens of terminals has been found to play a great role in addressing these issues. But for efficient performance the wireless channel of the massive MIMO system must be efficiently identified, and this brings about the idea of system identification. The area of system identification is one of the important areas in engineering as it can be applied to a wide range of problems. System identification is widely used in many areas such as broadcasting theory, geology, hydrology, communications, control, etc [1]. There is need for exploring novel system identification methods or improving on the already existing system identification methods. This research will focus on the multiple-input multiple-output output-error state space (MOESP) continuous-time fractional-order system identification algorithm which is an extension of the extensively studied MOESP continuous-time integer-order system identification algorithm, a subspace system identification algorithm. Subspace system identification algorithms have been proven to successfully identify the dynamics of a system and this has motivated their application in this thesis [2]. The developed continuous-time fractional-order system identification will then be used to identify the massive multiple-input multiple-output (MIMO) frequency-selective wireless channels.

The reasons of focusing on continuous-time system identification are: i) In most cases, dynamic systems in the real world environment are naturally described in the continuous-time domain, but most system identification methods have been based on discrete-time models with less focus on continuous-time models [3]. ii) The authors in

[4] showed that when continuous-time models are directly used to identify a system their performance is superior to when the indirect approach is used where discrete-time models are first identified and then transformed into continuous-time models. iii) In [5] their results showed that discrete-time system identification may be good enough in some instances, but if the conditions of the identification experiment are not adequately in favour of discrete-time methods then the results may not be reliable in the sense that the resultant models may be unstable, or even if they are stable they may not be accurate, and rapid sampling can make this problem more pronounced because the eigenvalues will lie close to the unit circle in the complex domain making the model parameters to be poorly defined in statistical terms thus leading to estimation errors. On the other hand, the authors showed that continuous-time system identification may be free from all these problems and may result in stable and more accurate models, especially with rapidly sampled data. iv) According to [6] continuous-time models provide differential equation models whose parameters can be interpreted immediately in physically meaningful terms, and these can be of direct use to environmental scientists and engineers who most often derive models in differential equation terms based on natural laws and who are much less familiar with discrete-time models. All these show some of the advantages of the continuous-time system identification methods.

Fractional-order system identification has been mostly applied in the field of controls and the use of the multiple-input multiple-output output-error state space (MOESP) continuous-time fractional-order system identification algorithm to identify the massive MIMO frequency-selective wireless channels will provide a novel research perspective in the area of wireless communications.

## **1.2 System Identification**

A system is a set of elements or components working together to achieve a particular goal or task. Systems can be classified as linear or nonlinear systems. Linear systems follow the principle of superposition, i.e. linearity and homogeneity and they have one isolated equilibrium point. It is easier to work with linear systems and different methods

can be used to identify the dynamics of the system with respect to the inputs, outputs and unmeasured disturbances/noise [7]. But in most cases, real life systems are nonlinear in nature, as a result the objective is usually to obtain an accurate representation of the nonlinear systems which at times can be obtained using localized linear models [8]. Nonlinear systems do not obey the principle of superposition and may have several isolated equilibrium points.

According to [8], one approach that can be used to model nonlinear systems is to use several linear models which when combined together can cover the operating range of the nonlinear system. The assumption here is that the process is locally linear within each of these operating ranges. Different models can also be used for the identification of nonlinear systems, and these may include: Volterra series models, block structured models, neural network models, nonlinear autoregressive moving average with exogenous input (NARMAX) models and state-space models [9] A brief description of these models is presented in Appendix A.

System identification is a technique used to develop mathematical models based on the input and output data sets of a system to represent the characteristics or dynamics of that particular system. The general form of the developed mathematical model involves a number of ordinary differential or difference equations and a set of parameters which have to be estimated [10]. The advantage of system identification is that models for systems with very complex dynamics or systems with unknown physical parameter values can be developed.

Modelling techniques can be classified as a priori modelling, a posteriori modelling or grey box modelling. In a priori modelling the models are built based on first principles where simple experiments are conducted to find out the physical laws involved, e.g. Newton laws. A priori modelling is also called white box modelling or theoretical modelling. In a posteriori modelling the model is built based only on available data without having previous knowledge about the system, there is no prior model available. It describes how the output data depends on the input data. Most system models are of this type. A posterior modelling is also called black box modelling or experimental modelling. Theoretical modelling can deliver more

information about the system to be identified provided that the internal behaviour of the system is known and can be described mathematically, but there has been an increased attention in experimental modelling because i) theoretical modelling can become quite complex even for simple systems, ii) model coefficients derived from the theoretical modelling are not precise enough and not all actions taking place inside the system are known [11]. In grey box modelling the dynamics of what is going on inside the system are not fully known, so it is based on both insight into the system and on experimental data analysis [12].

Linear system identification methods can be broadly categorised into two classes, namely parametric and non-parametric identification. Parametric identification estimates the unknown parameters in the model that govern a system. In parametric identification the system models are completely defined by the system coefficients or parameters [13]. Parametric identification models are used to determine the system coefficients or parameters and these models include the transfer functions autoregressive (AR) models, autoregressive with exogenous input (ARX) models, autoregressive moving average with exogenous input (ARMAX) models, Box-Jenkins (BJ) models, output-error (OE) models and models and state-space models. A brief description of these models is presented in Appendix B. Due to the fact that linear systems are easier to work with compared to nonlinear systems and in this case offer a good starting point for system identification for the massive MIMO system, the focus of this thesis is on linear systems that employ the state-space models for system identification. An expanded discussion on state-space models is presented in the chapters that follow.

On the other hand, non-parametric identification assumes that the model that governs a system is unknown and it tries to estimate the generic model using step response, finite impulse response (FIR) and frequency response [14]. Non-parametric identification techniques provide a very effective and simple way of finding model structure in data sets without the imposition of a parametric one [12].

In system identification the choice of the model to use to identify the system depends on the dynamics and the noise characteristics of the system. This is because

some models may not consider the noise characteristics when it comes to system identification, for example deterministic models ignore the effects of the process noise and additive noise.

In general, system identification may follow these steps:

1. Experiment design - Setting up the system in terms of what signals to measure, choice of sampling rate and choice of persistently excitation signals.
2. Data acquisition – Getting the system input and output data.
3. Model selection - Determining the type of model that is required for the observation, e.g. open loop model or closed loop model. In the open loop model the direct identification of the output from the input is considered. But in the closed loop model there is a feedback where the output is fed back to the input before identification.
4. Model estimation - Identifying which suitable model to use from the different available. This model should be the one that provides the best approximation of the observed data and minimises the cost function.
5. Model validation - Ascertain how satisfactory/ accurate the specified model is in describing the system. Simulate and compare the model with sets of data.

System identification can be performed in either time domain or frequency domain. In time domain identification the measured data is directly used to estimate the model parameters. Whereas in frequency domain identification the measured data is first transformed to frequency domain using Discrete Fourier transform (DFT) and then the model parameters are estimated in the frequency domain transformed data.

The identification process can also be classified as online or offline identification [11]. In online identification the system identification process is carried out the same time that the input and output data are being collected, i.e. the system identification

process is performed parallel to the experiment. Here the data is processed immediately after becoming available. This is the typical identification process for real-time systems. Online identification can also be referred to as real-time or recursive identification. Recursive identification updates the model as each measurement becomes available. Hence, the new measurement is always used to improve the model derived in the previous step. The old measurements do not need to be stored. On the other hand in offline identification the measured data is first stored and then the system identification process is carried out on the stored data, here the data is processed in one go. In offline identification the identification process usually takes a longer time compared to online identification. Offline identification can also be referred to as batch or non-recursive identification [15]. Non-recursive identification determines the model from the previously stored measurements. Non-recursive identification can be further divided into direct processing or iterative processing. In direct processing the model is identified in one pass or one step whereas in iterative processing the model is identified in several steps.

In system identification the system to be modelled must be excited by an operational input signal or an artificially generated signal known as a test signal. The operational input signal should excite all relevant frequencies. The test signal is usually applied if the operational input signal does not sufficiently or persistently excite the system. Persistent excitation plays an important role in establishing parameter convergence. The definition of a persistently exciting signal is given in Appendix C. Commonly used excitation signals include pseudo random binary sequence, chirp, unit step. The estimated model is usually more accurate in frequency ranges where the input signal has high energy.

### **1.3 Massive MIMO**

Massive multiple-input multiple-output (MIMO) also known as ARGOS, Very Large MIMO, Large-Scale Antenna Systems, Full-Dimension MIMO or Hyper MIMO is an emerging technology that uses antenna arrays which have hundreds of antennas at the base station to service tens of terminals in the same time-frequency resource [16], [17],

[18], [19]. The term massive in this context means that the number of base station antennas is one hundred or more. Figure 1.1 shows possible antenna configurations for a massive MIMO base station [16].

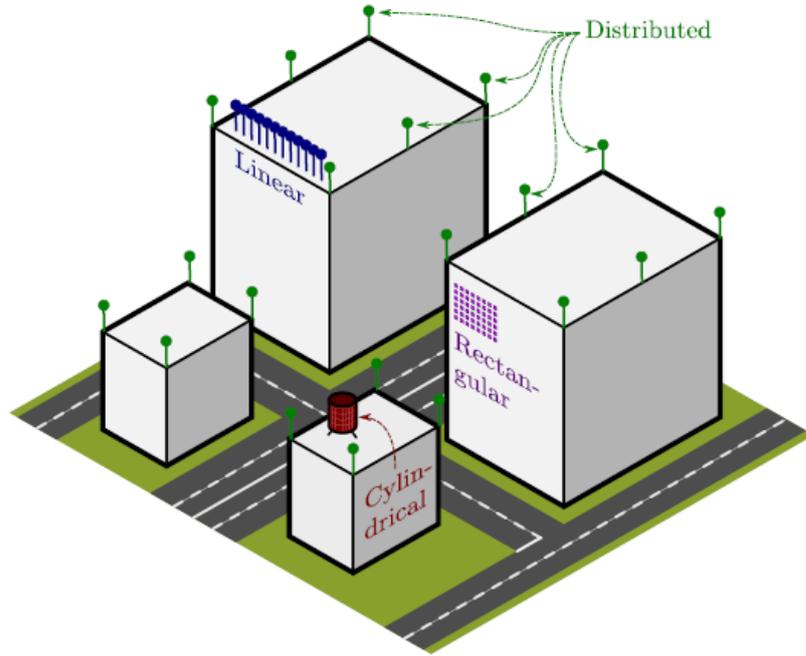


Figure 1. 1: Possible antenna configurations for a massive MIMO base station

The cylindrical antenna configuration is compact and relatively small in size compared to the linear antenna array which is large in size [20]. Based on a design point of view, it is more desirable to have a compact array with a large number of antennas at the base station end; but making the arrays smaller in size results in high antenna correlation which then affects the performance of the wireless system.

With an increase in the number of base station antennas arrays, the channel vectors between the terminals and the base station become very large-size random vectors and are under favourable propagation conditions, meaning that they become pairwise orthogonal [21], [22]. As a result, with linear processing such as maximum-ratio combining (MRC) or zero forcing (ZF), assuming perfect channel state information (CSI) at the base station, the interference from other users can be cancelled without consuming more time-frequency resources. The limiting factor on the number of terminals that can be serviced in massive MIMO is on the inability to obtain channel state information for an unlimited number of terminals [17].

Massive MIMO is a technology that is being proposed for fifth generation (5G) wireless communications, and is said to achieve the benefits of multiuser MIMO such as increased capacity, increased data rate, enhanced reliability, reduced latency, improved energy efficiency, improved spectrum efficiency and reduced interference but at a greater extent [23], [24] with simple linear processing such as maximum-ratio combining (MRC) or zero-forcing (ZF) on the uplink and maximum-ratio transmission (MRT) or ZF on the downlink [25].

Spectrum efficiency in a wireless communication system is limited by the information-theoretic capacity, which depends not only on the signal-to-noise ratio (SNR) but also on the spatial correlation in the propagation environment, channel estimation accuracy, transceiver hardware impairments and signal processing resources. Thus massive MIMO offers the remarkable potential of both increasing the spectral efficiency and relaxing implementation issues such as lowering the high price of having stricter hardware and overhead requirements. [26].

In massive MIMO the radiated energy can be focused to specific terminals within the network thus leading to reduced intra and intercell interference [18] and much higher energy efficiency. Another advantage of massive MIMO includes the use of cheap, low quality and low power components in its design, as the use of coaxial cables will be eliminated in its design [19]. If expensive components were used then the massive MIMO deployment cost would scale up with the number of radio frequency (RF) front ends and components [27]. According to [17], with massive MIMO the effects of small scale Rayleigh fading and thermal noise can be averaged out so that the system performance is mainly affected by interference from neighbouring cells. They found out that this effect is more pronounced in the uplink as compared to the downlink. They also stated that massive MIMO increased the resolution of the antenna arrays, resulting in high precision when it comes to resolving individual scattering centres. As the number of antennas gets sufficiently large, interference between co-scheduled users and the impact of fast fading vanishes [28].

Massive MIMO relies on spatial multiplexing and this is highly dependent on the base stations having good channel state information, in both the downlink and the uplink [16]. Acquiring the channel state information on the uplink depends on the terminals sending pilots to the base stations from which the base stations then estimates the channel state information. But the downlink channel estimation is the one which is a bit complicated, thus massive MIMO system relies on reciprocity for the channel state information [29].

There are several challenges associated with massive MIMO. These may include, channel estimation, pilot contamination and radio frequency (RF) impairments [28] and they are discussed in the following sections.

### **1.3.1 Channel Estimation**

Massive MIMO basically operates in the time division duplexing (TDD) mode although the frequency division duplexing (FDD) mode may be possible in certain cases [16], and relies on reciprocity for channel state information where the downlink channel can be acquired by simple processing of the estimated uplink channel and vice-versa. With reciprocity the base station can make use of the channel state information estimated from the uplink training and in turn the user terminals can also make use of the channel state information estimated from the downlink training.

Time division duplexing uses a single frequency band for both transmitting and receiving data. This single frequency band is shared by assigning alternative time slots for transmission and reception, Figure 1.2.

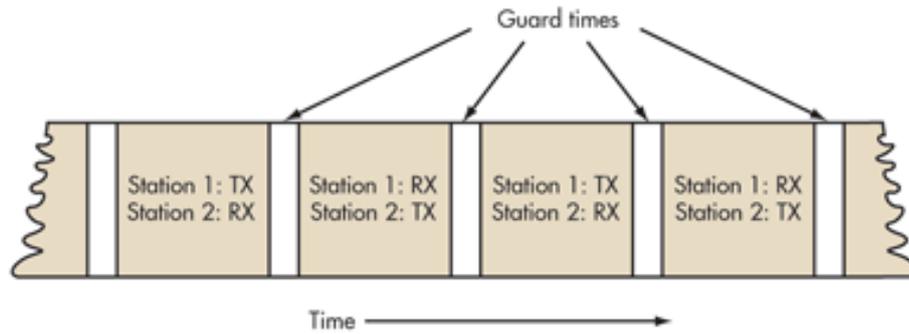


Figure 1. 2: Time division duplexing

Frequency division duplexing uses two different frequency bands, one for transmitting and the other one for receiving the data, Figure 1.3. Signal transmission and reception can occur simultaneously.

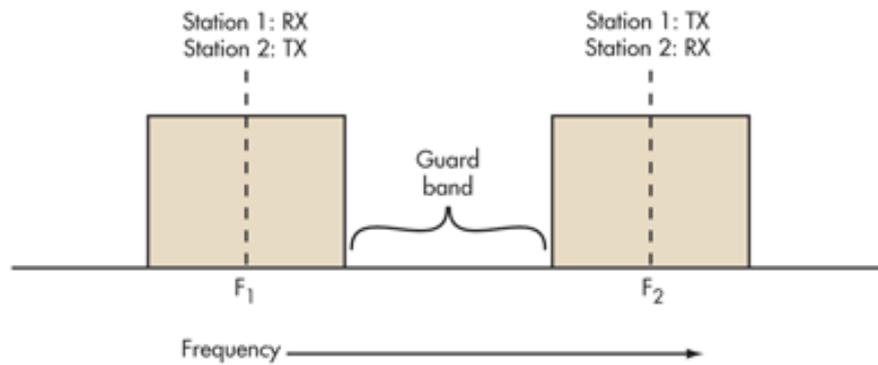


Figure 1. 3: Frequency division duplexing

In the FDD mode, the user equipment (UE) provides the network with quantised channel state information through a feedback channel.

Exact channel state information (CSI) is very important in massive MIMO systems as it has great impact on the accuracy of signal detection. According to [30], channel reciprocity ideally follows this pattern:

- i) The transmitting station sends pilot symbols to the receiving station.
- ii) The receiving station then uses these pilot symbols to estimate the corresponding channel.

- iii) After estimation the receiving station then transmits the estimated channel information back to the transmitting station.

To be able to use reciprocity and to calculate pre-coding coefficients in massive MIMO, it is important to know or estimate the differences in the (frequency) responses between the downlink and uplink parts of the hardware chains. This process of estimation is called reciprocity calibration [29]. Reference [29] showed that in massive MIMO, reciprocity calibration could be done on the base station side by sounding the base station antennas one by one while receiving with the other base station antennas. They showed that when having an  $m$  antenna base station,  $m(m-1)$  signals are generated that could be used for calibration purposes.

### 1.3.2 Pilot Contamination

Another challenge associated with massive MIMO is pilot contamination. Pilot contamination results when one pilot symbol from one cell is reused in another neighbouring cell resulting in interference between the cells [31]. The reuse of pilot symbols between cells is a result of the limited number of orthogonal pilot symbols that can be used with increase in the number of base station antennas. During the uplink transmission we let users in neighbouring cells be assigned identical pilot symbols as shown in Figure 1.4 (a), [32], where  $h_{i,k,l}$  is the channel between user terminal  $k$  which is located in cell  $l$  and the neighbouring base station in cell  $i$ . The neighbouring base stations will then receive these pilot symbols and during the downlink transmission these base stations will transmit signals not only to their own users but also to the neighbouring users having the same identical pilot symbols as the intended users, as shown in Figure 1.4 (b), [32] thus causing interference. Furthermore, due to the limited bandwidth, it is not possible to allocate unique pilot symbols for the users in all the cells. This restriction on the availability of orthogonal resources forces the reuse of pilot symbols for users in different cells [33].

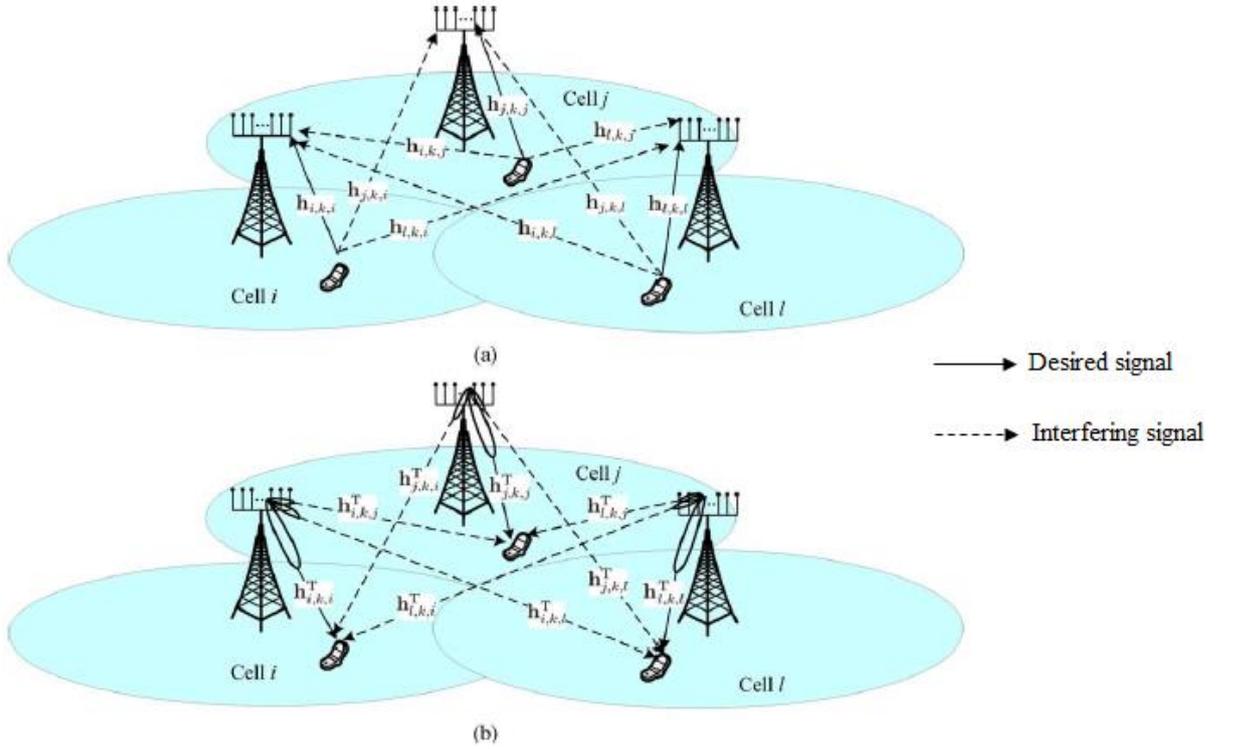


Figure 1. 4: Pilot contamination concept, (a) Uplink transmission (b) Downlink transmission

When the CSI is estimated from the uplink pilot symbols the uplink transmit power per user can only be reduced inversely proportionally to the square-root of the number of base station antennas [22]. This is because when the transmit power of each user is reduced, channel estimation errors become pronounced. As a result, with CSI estimated from pilot symbols, the benefits of using very large antenna arrays are somehow reduced.

According to [33], the effect of pilot contamination on system performance is more detrimental compared to the effect of additive white noise of similar variance. This is because the noise affecting the signal during data transmission is independent of the noise corrupting the channel estimate. On the other hand, the pilot contamination is correlated during the training and data transmission phase, since the channel does not change significantly during this time interval. As the number of antennas at the base station increases to an unprecedented number, the effects of additive noise and Rayleigh fading disappear and what remains is the inter-cell interference resulting from pilot contamination [34].

Several methods have been suggested in [16] to help deal with the drawback associated with pilot contamination. These include i) the use of a less aggressive frequency reuse factor, i.e. 3 or 7 for the pilot symbols so as to reduce interference between neighbouring cells, ii) the use of smart channel estimation techniques or blind techniques that would avoid the need for pilot symbols, and iii) the development of new precoding schemes such as pilot symbol contamination precoding that would only rely on slow fading coefficients.

Conventional channel estimation schemes cannot effectively mitigate the problem of pilot contamination, and even then, the computational complexity becomes larger due to the large number of antenna arrays employed in massive MIMO systems [21]. The authors in [21] proposed an improved multicell minimum mean square error (MMSE) joint channel estimation scheme for massive MIMO systems. Their scheme aims to mitigate pilot contamination using low computational complexity and in a more realistic situation. Their proposed algorithm was shown to save 61.2% computational cost with respect to that of the full-cell MMSE scheme. In their proposed scheme, they firstly identify highly interfering users in neighbouring cells based on the estimation of the large-scale fading and then included in the joint channel processing.

In [35] they proposed an algorithm to avoid pilot contamination in cellular systems with power controlled handoff. The dominant complexity of their algorithm is a singular value decomposition of the received signal block. They analysed their algorithm by means of random matrix theory. The work in [36] proposed a pilot contamination mitigation technique based on a low-complexity pilot power control and a pilot contamination avoidance (within a group of cells) scheme based on pilot reuse. The authors in [37] addressed the problem of pilot contamination in massive MIMO systems and investigated the effect of cell size reduction on the effect of pilot contamination. They showed that cell size reduction was capable of removing the effect of pilot contamination and improving the performance of massive MIMO systems.

### **1.3.3 Radio Frequency Impairments**

Due to the large number of antenna arrays, massive MIMO is expected to use low cost radio frequency (RF) components which are susceptible to hardware impairments such as in-phase and quadrature (IQ) imbalance, phase noise, amplifier nonlinearity and quantization errors [26]. These impairments can create a mismatch between the intended data signal and what is actually generated and emitted, and they distort the received signal at the receiver end. This causes great system performance degradation as these impairments can affect the downlink and uplink capacity of user equipment [38] and can also affect the accuracy of the channel estimation [39]. Radio frequency impairments can also severely affect the performance of linear detectors [40]. Compensation and calibration schemes which can be implemented by analog and digital signal processing are usually used to combat the effects of these impairments. The compensation techniques do not completely remove these impairments because the time-varying hardware characteristics cannot be accurately or fully estimated and because of the randomness introduced by the different types of noise in the system and as a result we have residual RF impairments [26]. Residual RF impairments also arise from the use of inaccurate models to characterise the behaviour of RF impairments and from imperfect parameters estimation errors due to thermal noise [41].

In [26] they studied a massive MIMO system in the presence of RF impairments at both the base stations and the single antenna users. They showed that hardware impairments limit the accuracy of channel estimation and the uplink/downlink capacity of each user equipment (UE). The capacity was mainly limited by the hardware at the UE, while the impact of impairments in the large-scale arrays vanishes asymptotically and inter-user interference (in particular, pilot contamination) becomes negligible.

#### **1.3.1.1 IQ Imbalance**

Owing to its flexibility as it is able to operate with several different air interfaces, frequency bands and waveforms [42] and low complexity, direct conversion or homodyne or zero intermediate frequency (IF) transceivers are preferred over super heterodyne transceivers. In a direct-conversion transceiver much of the signal

processing is done digitally, implying minimal errors. But once the signal has been converted to the analog domain by the digital-to-analog converter then it is more prone to analog impairments such as IQ imbalance.

In direct up-conversion, the message signal in the in-phase (I)-path is up-converted by the local oscillator. While the quadrature-phase (Q)-path signal is up-converted with the  $90^\circ$  phase-shifted version of the local oscillator signal. The local oscillator is the carrier frequency of the radio frequency (RF) signal. The translated I and Q components are then summed up (or subtracted) to produce the final RF signal as shown in Figure 1.5.

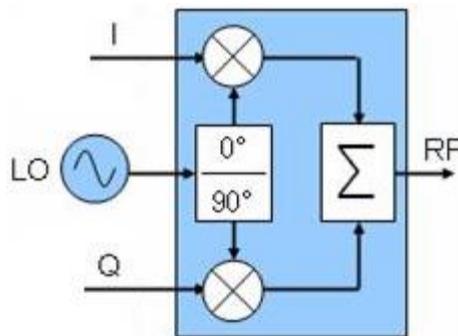


Figure 1. 5: Direct up-conversion

In a direct-conversion transmitter, the real and imaginary parts of the message signal are first passed through a digital-to-analogue-converter (DAC) after which the signal is then passed through a low pass filter (LPF) and then up-converted using the local oscillator. After up-conversion, the resulting RF signal is amplified by the power amplifier (PA) before being transmitted through the communication channel. Figure 1.6 shows the block diagram of a direct up-conversion transmitter.

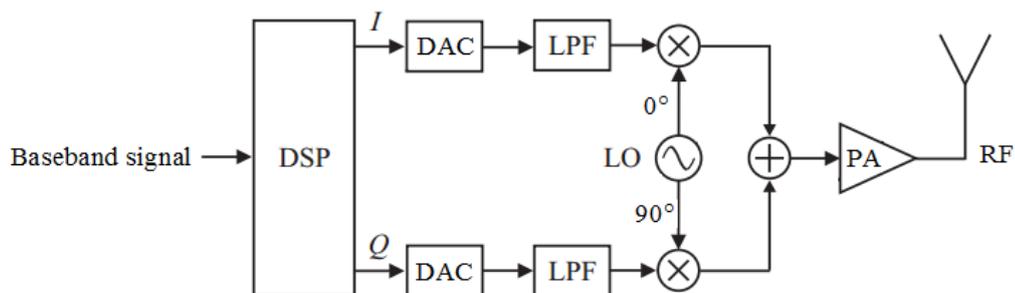


Figure 1. 6: Block diagram of a direct up-conversion transmitter

In the direct down-conversion receiver, the real and imaginary parts of the received RF signal are mixed with the local oscillator signal whose frequency is identical to the carrier frequency and its  $90^\circ$  phase-shifted version, respectively, to produce the baseband signal as shown in Figure 1.7.

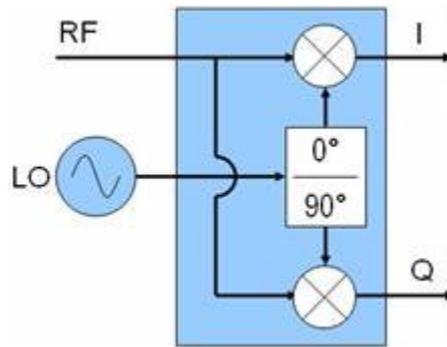


Figure 1. 7: Direct down-conversion

At the receiver end, the received RF signal is passed through a bandpass filter (BPF) and then a low-noise power amplifier (LNA). It is then down-converted using the local oscillator. The real and imaginary parts of the signal are then passed through a low pass filter (LPF), an amplifier (A), an analog-to-digital (ADC) converter and then combined to form the baseband signal. Figure 1.8 shows the block diagram of a direct down-conversion receiver.

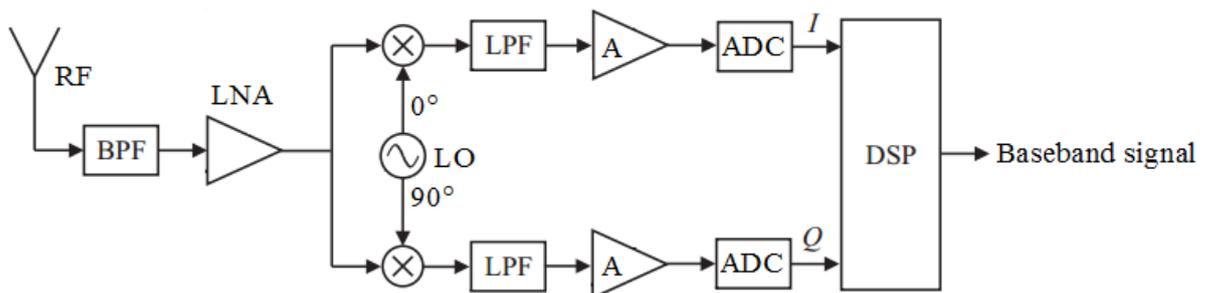


Figure 1. 8: Block diagram of a direct down-conversion receiver

In the direct-conversion transceivers, both the sine (in-phase local oscillator) and cosine (quadrature-phase local oscillator) waveforms are used to perform the conversion. These waveforms multiply the message or received signal so as to perform

the up or down-conversion. The sine and cosine waveforms performing the up or down-conversion need to be orthogonal, i.e., should have exactly  $90^\circ$  phase shift between them. On top of that they should also have the same amplitude, otherwise any mismatches on the I and Q branches after up or down-conversion will contribute to the overall IQ imbalance in the system.

The direct conversion architecture provides image attenuation, thus eliminating the need for image rejection filters [43] since the signal processing provides for image rejection in the in-phase and quadrature arms. This architecture does not have an intermediate frequency stage as is the case in the super heterodyne architecture thus reducing power consumption and implementation costs [44]. But direct-conversion transceivers suffer from DC offset and flicker noise or pink noise ( $1/f$ -noise). Due to manufacturing imperfections, direct-conversion transceivers also suffer from phase and amplitude mismatch in the in-phase and quadrature branches of the IQ arms commonly known as IQ imbalance [45]. As a result, it is important to study the effects of IQ imbalance on system performance and come up with techniques to mitigate them.

### ***Causes of IQ Imbalance***

IQ imbalance can be defined as the mismatch in amplitude and phase of the in-phase and quadrature-phase branches of the signal, i.e. this is a mismatch between the real and imaginary parts of the complex signal. IQ mismatch occurs because of the error in the nominally  $90^\circ$  phase shifter and the mismatch between the amplitudes of the LO in-phase and quadrature-phase outputs [46]. This mismatch is as a result of limited accuracy of analogue hardware such as finite tolerance of capacitors and transistors and temperature variations [27]. IQ imbalance can also be defined as the mismatch between I and Q balances in the IQ modulation /demodulation of the complex valued signals at the transceivers. IQ imbalance can be introduced at both the transmitter end (during frequency up-conversion) and the receiver end (during frequency down-conversion), leading to severe system performance losses thus affecting the user experience.

## *Effects of IQ Imbalance in Massive MIMO Systems*

IQ imbalance causes self-interference between the wanted message signal and its image, i.e. the image signal is the conjugate of the wanted message signal. The image problem is as a result of multiplication by a sinusoid and not a complex exponential. If we could synthesize a complex exponential, we would not have the image problem. Typical communication waveforms are circular but in the presence of IQ imbalance these waveforms become non-circular [47]. This is because the wanted message signal is added up together with its image signal, thus causing self-interference. Figure 1.9 shows a communication system affected by IQ imbalance.

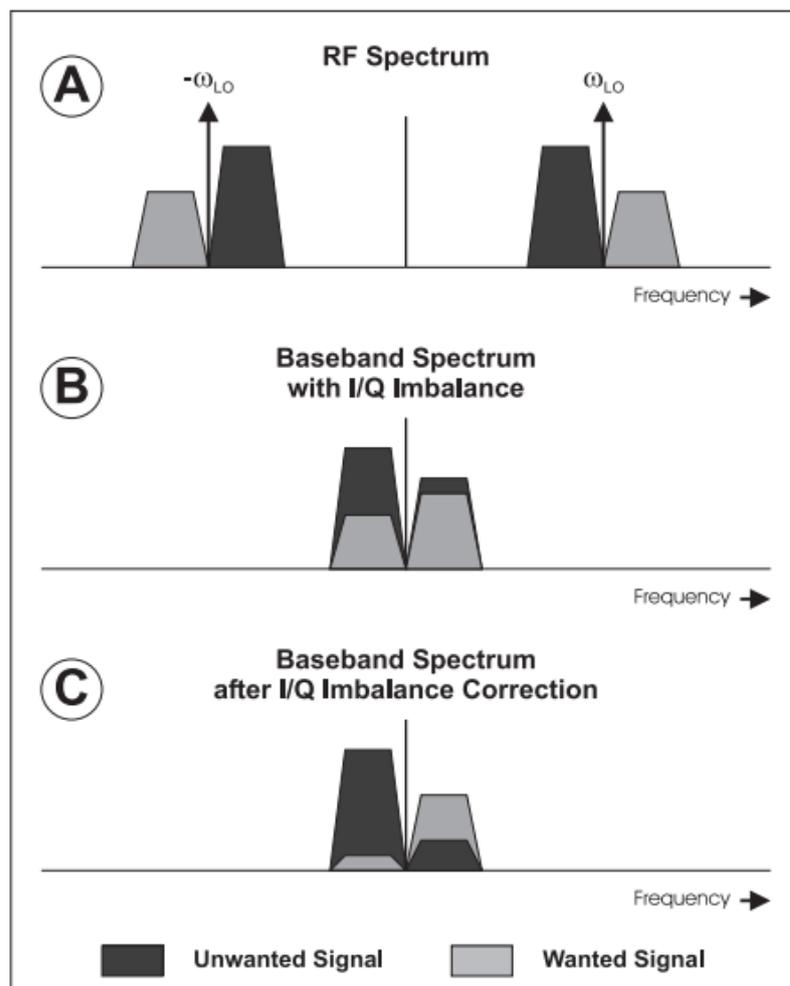


Figure 1. 9: The influence of IQ imbalance on the message (wanted) signal

Figure 1.9 (A) shows the baseband signal after it has been up-converted to the RF signal in the presence of local oscillator imbalances, i.e. IQ imbalance. Both the

message (wanted) signal and the image (unwanted) signal are present. Figure 1.9 (B) shows RF signal after it has been down-converted to the baseband signal. Due to IQ imbalance, the unwanted image signal is not fully rejected, and it mixes with the wanted message signal thus causing interference to it. Figure 1.9 (C) shows the baseband signal after IQ imbalance compensation. Here the IQ imbalance increases the image-rejection ratio of the receiver, as a result the unwanted image signal causes less interference to the wanted message signal.

Transmitter IQ imbalances can sometimes cause severe inter-user interference (IUI) at the receiver end [48]. IQ imbalance becomes more detrimental and more difficult to eliminate as the carrier frequencies increase [46]. An increase in carrier frequencies is expected to be the trend in future communications systems so as to utilize more bandwidth. As future communication systems target higher data rates, higher constellation sizes are needed, and higher operating signal-to-noise ratio are to be achieved to support such high-density constellations. Higher SNR requirements then translate to stricter IQ matching, since small errors of IQ can make the transitions of the RF signal too difficult to distinguish. This can then have a negative effect on the ability of a receiver to successfully demodulate the received signal [49]. On the other hand, adaptive techniques can be developed in the digital domain to track and minimise imbalances.

Since the performance of the baseband digital design is related to the RF analog processing, it is important to consider IQ imbalance in the design and performance evaluation of wireless communication systems [50]. In [46] they stated that different circuit topologies have been used in analog circuit designs that are more robust to component mismatches. But such techniques increase the device sizes, cost and raise the power consumption in the analog domain [48]. Even then, they do not completely remove the IQ mismatches. Initially, compensation techniques were proposed in the analog domain to calibrate the IQ branches, but these were shown to suffer from different offsets, errors in the measurement feedback loop, and a long calibration and they do not meet the target performance requirements. An alternative approach is to estimate and compensate for such distortions in the digital domain by digital signal

processing. Besides the mitigation of multi-user interference, IQ imbalance compensation also provides for the reliable detection of signals [40].

### *Modelling IQ Imbalance*

The received signal at the base station can be expressed as:

$$y = \mathbf{H}u + n \quad (1.1)$$

where  $\mathbf{H}$  is the channel matrix,  $u$  is the transmitted signal and  $n \sim CN(0, N_0)$  is the additive white Gaussian noise vector at the base station

A signal with IQ imbalance can be modelled as [50]:

$$r_{IQ} = K_1 r + K_2 r^* \quad (1.2)$$

where  $r$  is the baseband equivalent signal in the absence of IQ imbalance and the operation  $(\cdot)^*$  denotes the complex conjugate or image signal. The IQ imbalance coefficients,  $K_1$  and  $K_2$  can be expressed as  $K_1 = (1 + ge^{-j\theta})/2$  and  $K_2 = (1 - ge^{j\theta})/2$ , where  $g$  and  $\theta$  denote the amplitude mismatch and the phase mismatch at the receiver, respectively. The coefficients can also be expressed as  $K_1 = \cos(\theta/2) + jg \sin(\theta/2)$  and  $K_2 = g \cos(\theta/2) - j \sin(\theta/2)$ . The relationship between the coefficients  $K_1$  and  $K_2$  can be expressed as  $K_2 = 1 - K_1^*$  [51]. The phase imbalance is any phase deviation from the ideal  $90^\circ$  between the I and Q branches. Amplitude imbalance can be expressed as [46]:

$$g = \frac{a_I - a_Q}{a_I + a_Q} \quad (1.3)$$

where  $a_I$  and  $a_Q$  are the amplitude gains on the I and Q branches respectively. In decibels (dB), the amplitude imbalance can be expressed as  $10\log(1+g)$ .

Under ideal conditions, the imbalance parameters are  $g=1$  and  $\theta=0$ , thus  $K_1=1$  and  $K_2=0$ . Generally, the degree of IQ imbalance is evaluated using the image rejection ratio (IRR) and this can be expressed as [50]:

$$IRR = \frac{|K_1|^2}{|K_2|^2} \quad (1.4)$$

In an ideal case, where there is no IQ imbalance, IRR is equal to infinity. An alternative measure used to quantify the undesirable leakage of the image signal relative to the gain of the desired signal can be defined as image leakage ratio (ILR) [50]:

$$ILR = \frac{|K_2|^2}{|K_1|^2} \quad (1.5)$$

Ideally, the ILR is equal to zero. The difference between the two is that IRR denotes a quality measure, whilst ILR denotes an error measure.

Assuming IQ imbalance is only present at the transmitter end. The signal to be transmitted in the presence of IQ imbalance can be expressed as:

$$u_{IQ_T} = K_{1_T} u + K_{2_T} u^* \quad (1.6)$$

where  $u$  is the transmitted message signal in the absence of IQ imbalance, and  $K_{1_T}$  and  $K_{2_T}$  are the transmitter IQ imbalance coefficients.

Next it is assumed that IQ imbalance is only present at the receiver end. The received signal in the presence of IQ imbalance can be expressed as:

$$y_{IQ_R} = K_{1_R} y + K_{2_R} y^* \quad (1.7)$$

where  $y$  is the received signal in the absence of IQ imbalance, and  $K_{1_R}$  and  $K_{2_R}$  are the receiver IQ imbalance coefficients. Substituting (1.1) into (1.6) results in:

$$y_{IQ_R} = K_{1_R} (\mathbf{H}u + n) + K_{2_R} (\mathbf{H}u + n)^* \quad (1.8)$$

$$y_{IQ_R} = K_{1_R} \mathbf{H}u + K_{2_R} \mathbf{H}^* u^* + K_{1_R} n + K_{2_R} n^* \quad (1.9)$$

According to [50] the received SNR in the presence of IQ imbalance at the receiver end can be expressed as:

$$SNR_{IQ} = \frac{SNR \times |\mathbf{H}|^2}{SNR \times ILR \times |\mathbf{H}|^2 + (1 + ILR)} \quad (1.10)$$

where  $SNR$  is the ideal signal-to-noise ratio or signal-to-noise ratio in the absence of IQ imbalance.

Since the transmitted and received signals, (1.5) and (1.9) respectively are affected by IQ imbalance it is then important to find techniques to reduce the effect of IQ imbalance on these signals. IQ imbalance compensation is one technique that has been identified to help mitigate the problems associated with IQ imbalance. With reference [50] the SNR after IQ imbalance compensation can be written as:

$$SNR_{IQ\_compensation} = \left( \frac{1 - ILR}{1 + ILR} \right)^2 \times SNR \times |\mathbf{H}|^2 \quad (1.11)$$

where  $SNR$  is the ideal signal-to-noise ratio or signal-to-noise ratio in the absence of IQ imbalance.

We then studied the performance of a massive MIMO system in the presence of IQ imbalance with and without compensation in terms of the symbol error rate (SER). The SER is one of the performance parameters used in communication systems and it refers to the ratio of the number of symbols incorrectly received to the total number of symbols transmitted during a specified time interval. Ideally, this ratio should be zero, but we know that real life systems do suffer from inefficiencies and as a result this value cannot be zero. But for good performing systems this value should be as close as possible to zero. We ran our simulations in MATLAB where we considered a massive MIMO system equipped with one hundred transmitting antennas at the base station and one receiving antenna at the user terminal. For our simulations we considered BPSK, QPSK and 256-QAM modulated signals and a quasi-static Rayleigh fading channel. We selected an amplitude mismatch of  $g = 3dB$  and a phase mismatch of  $\theta = 15^\circ$ . These were arbitrary values chosen with reference to [52].

Figures 1.10 to 1.12 depict the SER curves for an ideal received signal, i.e. received signal not affected by IQ imbalance, received signal in the presence of IQ imbalance but without IQ imbalance compensation and received signal after IQ imbalance compensation for different modulation schemes.

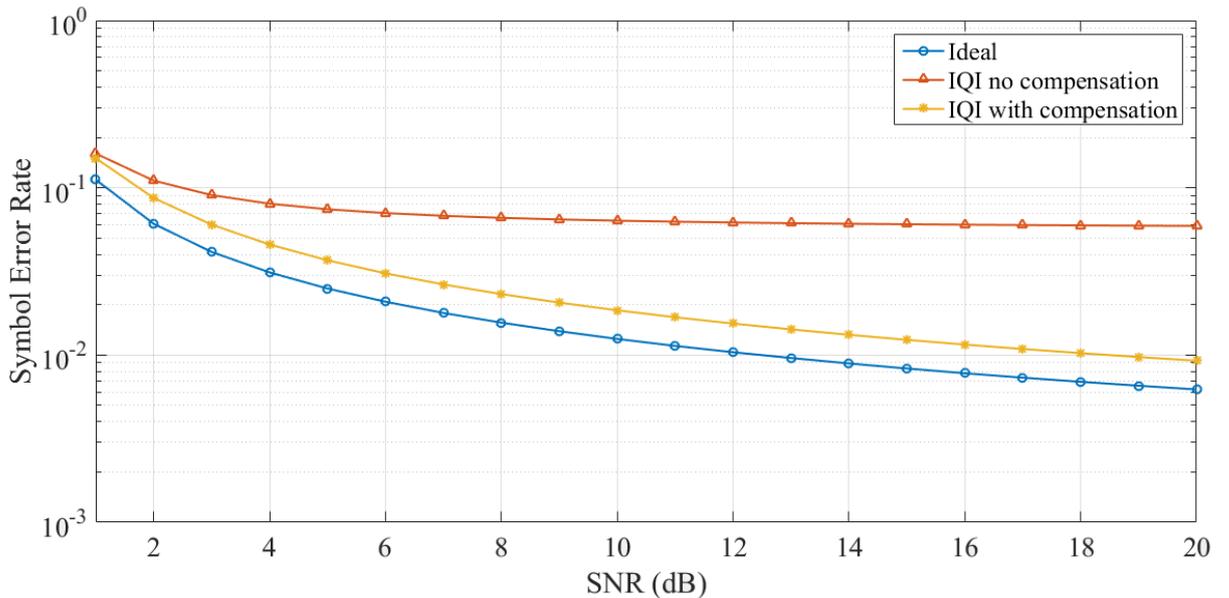


Figure 1. 10: SER vs. SNR for ideal received signal, received signal with IQ imbalance and received signal with IQ imbalance compensation for BPSK signal

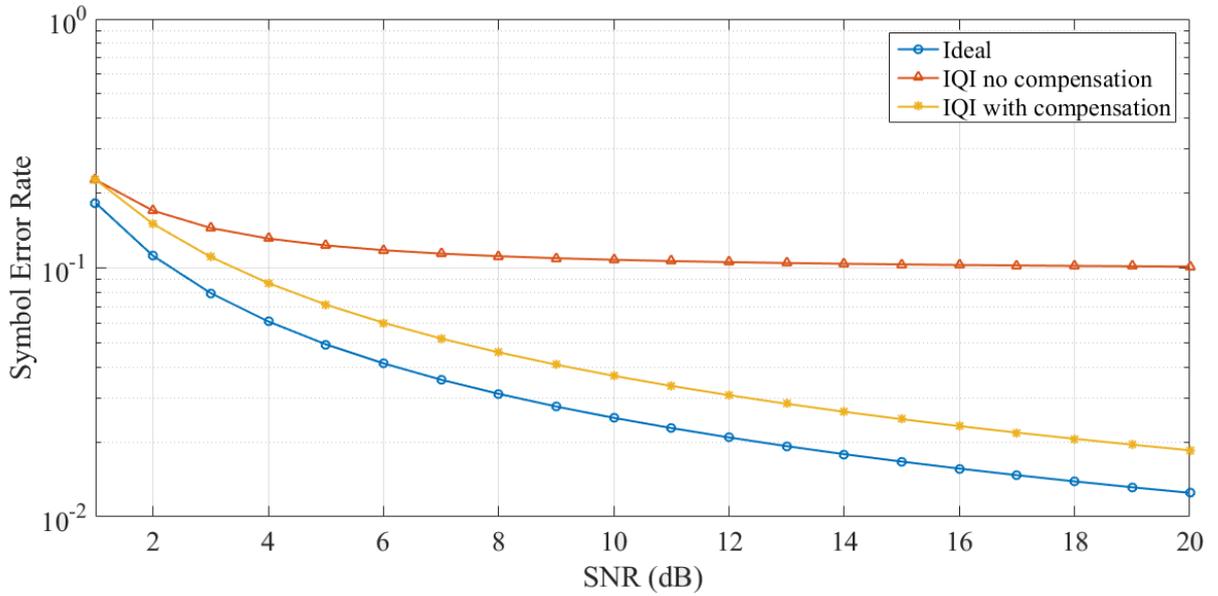


Figure 1. 11: SER vs. SNR for ideal received signal, received signal with IQ imbalance and received signal with IQ imbalance compensation for QPSK modulated signal

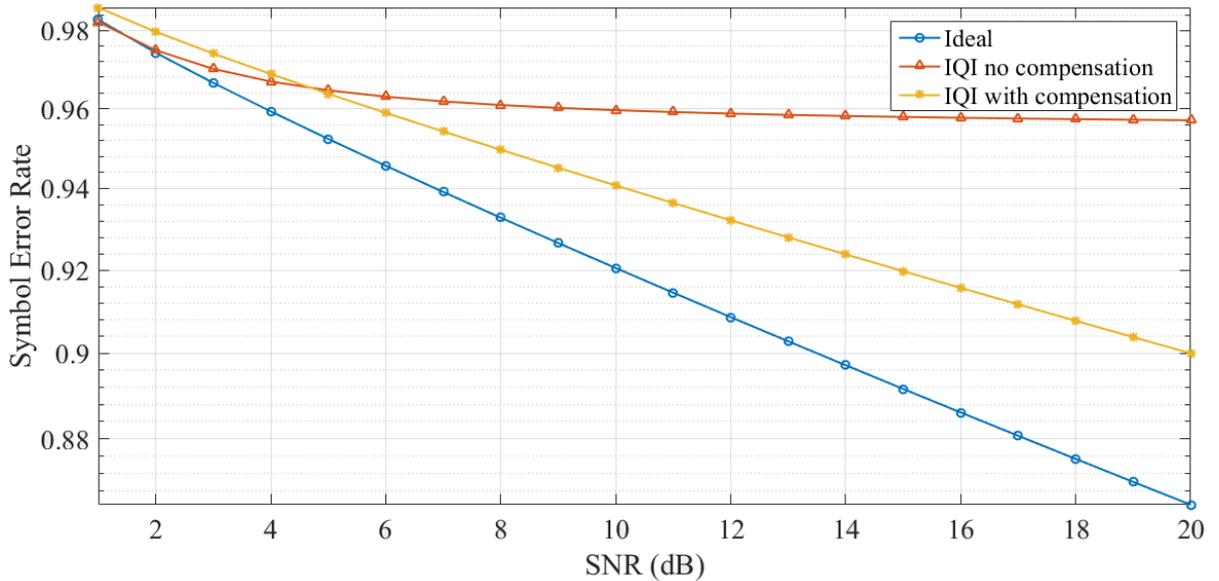


Figure 1. 12: SER vs. SNR for ideal received signal, received signal with IQ imbalance and received signal with IQ imbalance compensation for 256 QAM modulated signal

We can observe from Figures 1.10 to 1.12 that after IQ imbalance compensation the performance of the massive MIMO system improved, i.e. the IQ imbalance compensated case showed better results than the IQ imbalance case without any compensation especially with increase in SNR. This is because IQ imbalance degrades

system performance leading to poor system performance and this drawback can be rectified by using IQ imbalance compensation techniques.

In addressing the issue of IQ imbalance in the massive MIMO system we assumed that the channel state information is known, but in reality, the channel state information is not available meaning that it needs to be estimated. To estimate the channel of the massive MIMO system we propose the use of system identification tools as presented in the following chapters.

To exhaust our discussion on RF impairments we then give a brief overview on phase noise and amplifier nonlinearity

### 1.3.3.2 Phase Noise

According to [53], phase noise is introduced at the transmitter during up conversion of the baseband signal, i.e. when the baseband signal is multiplied with the carrier generated by the local oscillator. The phase of the generated carrier varies randomly, resulting in a phase distortion of the transmitted signal. Phase noise also occurs at the receiver end during the down-conversion of the bandpass signal, i.e. when the bandpass signal is multiplied with the carrier generated by the local oscillator to obtain the baseband signal. It can also cause some inconsistencies between the estimated channel gain and the actual channel gain during data transmission, known as channel-aging phenomenon which can lead to poor system performance [54].

Phase noise is a result of the random mismatch between the phases of the local oscillator (LO) and the input carrier signal and has two effects on the received signal [55]. The first effect is known as the common phase error (CPE), which is the same random phase rotation in all sub-carriers regardless of their index [56]. The second effect is the inter-carrier interference (ICI) caused by the loss of orthogonality among adjacent sub-carriers [57]. Up and down conversion local oscillators are ideally harmonic functions which can be expressed as [57]:

$$x(t) = e^{j2\pi f_c t} \quad (1.12)$$

where  $f_c$  is the carrier frequency. In the presence of phase noise, the local oscillator output can be written as [58]:

$$x_{PN}(t) = e^{j2\pi f_c t + \phi(t)} \quad (1.13)$$

where  $\phi(t)$  is the random phase fluctuations of the local oscillator.

For higher order modulation schemes, phase noise also prevents carrier recovery and causes the constellation plot to spin [49].

### 1.3.3.3 Amplifier Nonlinearity

According to [59], due to the large number of subcarriers and multiplexed streams, massive MIMO OFDM causes a high peak-to-average-power ratio (PAPR) thus amplifier nonlinearity which further degrades the system performance. They went on to suggest that massive MIMO single carrier frequency division multiple access (SC-FDMA) may reduce the PAPR by sacrificing the beamforming gain with subband-precoding matrices. They then compared the performance of massive MIMO OFDM and massive MIMO SC-FDMA systems in the presence of power amplifier nonlinear distortion. Simulation results showed that the throughput for massive MIMO OFDM was higher than that for massive MIMO SC-FDMA owing to inappropriate subband-precoding and the equivalent PAPR of SC-FDMA. Thus the effect of amplifier nonlinearities in a transmitter can be reduced by having a low peak-to-average power ratio (PAPR) [26].

Having given a brief overview on system identification and massive MIMO together with its challenges, the only challenge we will consider in this thesis is channel estimation or system identification.

## 1.4 Research Contributions

The significant research contributions are highlighted as follows.

1. Considering IQ imbalance compensation as a technique to address the issue of IQ imbalance which degrades the performance of massive MIMO systems.
2. Formulating a continuous-time multiple-input multiple-output output-error state space (MOESP) integer-order algorithm to identify linear-time invariant (LTI) continuous-time massive multiple-input multiple-output (MIMO) frequency-selective wireless channels. Evaluating the performance of the proposed integer-order identification algorithm.
3. Formulating a continuous-time multiple-input multiple-output output-error state space (MOESP) fractional-order algorithm to identify linear-time invariant (LTI) continuous-time massive multiple-input multiple-output (MIMO) frequency-selective wireless channels. Evaluating the performance of the proposed fractional-order identification algorithm and compare it with that of the integer-order identification algorithm.
4. Developing a massive MIMO channel equalization algorithm using the identified integer-order channel models. Evaluating the performance of the integer-order equalization algorithm.
5. Developing a massive MIMO channel equalization algorithm using the identified fractional-order channel models. Evaluating the performance of the fractional-order equalization algorithm and compared it with that of the integer-order channel models.

## 1.5 List of Publications

M. Lupupa, and J. Qi, “I/Q imbalance in generalized frequency division multiplexing under Weibull fading”. *PIMRC 2015*: 471-476.

M. Lupupa, F. Zheng, and S. Hadjiloucas, “Joint transmitter/receiver IQ imbalance compensation in massive MIMO Systems”. *Sensors and their Application XVIII 2016*.

M. Lupupa, and S. Hadjiloucas, “Fractional-order system identification in massive MIMO systems”. *XXII World Congress of the International Measurement Confederation (IMEKO) 2018*.

## 1.6 Thesis Outline

This thesis is organised as follows.

Chapter 2 introduces the idea of state-space modelling and the massive MIMO system model. It then discusses on linear-time invariant (LTI), continuous-time massive MIMO system identification using the integer-order state-space modelling approach. A brief overview is given on different subspace identification algorithms with much emphasis on the multiple-input multiple-output output-error state space (MOESP) algorithm. The Poisson moment functional (PMF) filtering approach is then applied to the MOESP subspace algorithm to develop the identification model for the continuous-time massive MIMO system. The PMF approach pre-filters the input-output data to help overcome the problem associated with the time-derivatives of the data in continuous-time system identification. The performance of the proposed integer-order algorithm is then evaluated using MATLAB through analytical modelling.

Chapter 3 discusses on linear-time invariant (LTI) continuous-time massive MIMO system identification using fractional-order models. The application of fractional-order system identification in wireless communications is a novelty and is one of the main contributions of this thesis. The chapter looks at the basics of

fractional-order calculus and commonly used definitions to solve fractional-order differentiations and integral. The fractional-order MOESP algorithm is used to identify the dynamics of the massive MIMO system. The Poisson moment functional (PMF) approach is used to pre-filter the input-output data before developing the fractional-order continuous-time identification models. The PMF approach helps overcome the problem associated with fractional-order time-derivatives of the data in continuous-time system identification. Given the nature of fractional-order systems, their presentation and implementation in the practical sense is not easy and this chapter also discusses on different approximation methods that can be used to represent fractional-order systems given their complicated nature. The performances of the proposed fractional-order algorithm and the different approximation methods are then evaluated using MATLAB through analytical modelling.

Chapter 4 explores the use of integer-order state-space models in linear-time invariant (LTI) continuous-time massive multiple-input multiple-output (MIMO) frequency-selective wireless channel equalization. The chapter first looks at inter-symbol interference (ISI) and the different linear and nonlinear equalizers used to mitigate the effects on ISI on system performance. It also discusses on the minimum mean square error – decision feedback equalizer (MMSE-DFE), a hybrid equalizer which is constructed by combining linear and nonlinear equalizers thus benefiting from both equalizers. This hybrid equalizer is then implemented for the FIR massive MIMO channel model using the blockwise data model. It is then extended to the integer-order state-space massive MIMO channel model. The performance of the FIR massive MIMO channel model equalizer is then compared with that of the integer-order state-space massive MIMO channel model equalizer in MATLAB through analytical modelling.

Chapter 5 investigates the use of fractional-order models in linear-time invariant (LTI) continuous-time massive multiple-input multiple-output (MIMO) frequency-selective wireless channel equalization. Following from the build-up of the fractional-order system identification model, the fractional-order system equalization model in massive MIMO systems is also another novel contribution of this thesis. The minimum mean square error – decision feedback equalizer (MMSE-DFE) is implemented using the the blockwise data model to address ISI in the fractional-order state-space model.

The performance of this model is then compared with the performances of the FIR massive MIMO channel equalizer and integer-order state-space massive MIMO channel equalizer models in MATLAB through analytical modelling.

The conclusions and future work are presented in Chapter 6.

## Chapter 2

# Massive MIMO Continuous-Time System Identification Using Subspace Algorithm

### 2.1 Introduction

This chapter discusses on the use of the continuous-time state-space model to represent massive multiple-input multiple-output (MIMO) frequency-selective wireless channels for better model approximation or better system identification or better channel estimation.

The multiple-input multiple-output output-error state space (MOESP) algorithm which is a subspace system identification algorithm is used to identify the linear-time invariant continuous-time massive MIMO frequency-selective wireless channels. The use of state-space models in system identification in communications is motivated by the work by Li [60] and the work by Zhang and Bitmead [61]. In [60], Li used the state-space technique to estimate the fading wireless communication channels and in [61], the authors used the MOESP algorithm to identify the MIMO frequency-selective wireless channels. In their study all these authors focused on the discrete-time state-space model and we will extend the study of system identification in wireless communication to the continuous-time state-space model. Obtaining an equivalent continuous-time model is not always easy and difficulties are experienced whenever the sampling time is either too small or too large. Making the sampling time too small may create numerical problems because the poles are constrained to lie in a very small area of the  $s$ -plane. Making it too large may lead to loss of information, thus making the identification process unreliable. Some conversion methods use the matrix logarithm which may produce complex arithmetic when the matrix has negative eigenvalues. In addition, the zeros of the discrete-time system are not easily translatable to continuous-time equivalents as is the case with the poles [62].

To address these issues, this calls for the direct identification of the continuous-time state-space model from the sampled input-output data. The main problem with direct identification of the continuous-time state-space model is dealing with the non-measurable input-output data time-derivatives. In [63] the author mentions different methods to overcome this shortcoming when dealing with continuous-time state-space modelling, namely the use of linear filtering methods, the use of integral methods and the use of modulating function methods.

In this thesis we will use the Poisson moment functional (PMF) approach, a filtering method which is equivalent to the state variable filtering method to help overcome the problem associated with the input-output data time-derivatives in continuous-time system identification. The motivation behind using the PMF approach is that the PMF-based filter technique is most effective and has the advantage of having fewer design parameters than the more general state variable filter approach. Secondly, the PMF approach has appropriate theoretical frequency properties for estimating the time-derivatives or the integrals of the input-output signals [62].

For our work the channel will be assumed to be a quasi-static or slowly-varying Rayleigh fading channel and the pilot-symbol based channel estimation method is used. In this, training symbols known to both the transmitter and receiver are transmitted and received for training purposes. These training symbols provide the input-output data stream that is used for system identification. The performance of the state-space model is studied for different system orders.

## **2.2 Massive MIMO System Model**

We consider a massive MIMO wireless system as shown in Figure 2.1, with a base station equipped with  $m$  transmitting antenna elements, and a terminal station equipped with  $p$  receiving antenna elements having length- $L$  ISI channel paths, where for massive MIMO,  $m \gg p$ .

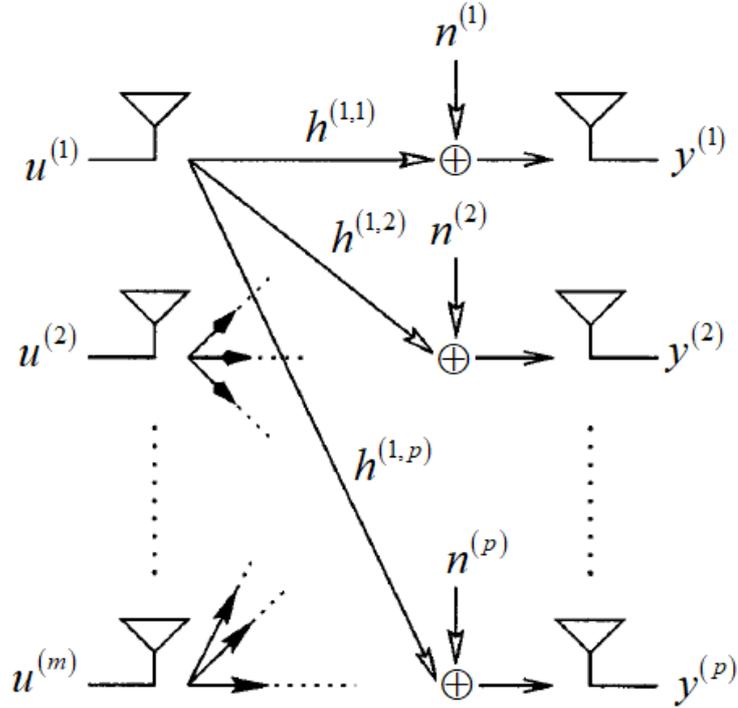


Figure 2. 1: Block diagram of a massive MIMO system

where  $u^{(m)}$  is transmitting antenna  $m$ ,  $y^{(p)}$  is receiving antenna  $p$ ,  $h^{(m,p)}$  is the channel path between transmit antenna  $m$  and receive antenna  $p$ .

In the modelling process, we assume that the channel is quasi-static, i.e. it is time-invariant within each frame and changes independently from frame to frame. We also assume that the antennas are sufficiently spaced such that there is no interference between antennas. Training symbols known to both the transmitter and receiver are inserted at the start of each frame to assist with channel estimation. The received signal of the linear-time invariant length- $L$  ISI channel massive MIMO system is expressed as:

$$\mathbf{y}(t) = \sum_{l=0}^{L-1} \mathbf{H}_l \mathbf{u}(t-l) + \mathbf{n}(t) \quad (2.1)$$

where  $\mathbf{y}(t)$  is the  $p \times 1$  received signal vector,  $\mathbf{u}(t-l)$  is the  $m \times 1$  transmitted symbols vector at time  $(t-l)$ ,  $\mathbf{n}(t) \sim CN(0, N_0 \mathbf{I}_p)$  is the  $p \times 1$  additive white Gaussian noise vector at the receiver side at time  $t$ , with  $N_0$  being the noise power and  $\mathbf{I}_p$  is the  $p \times p$

identity matrix and  $\mathbf{H}_l$  is the  $p \times m$   $l^{\text{th}}$  path massive MIMO channel matrix coefficients.

The transfer function of the channel matrix,  $\mathbf{H}_l$  in (2.1) is expressed as:

$$\mathbf{H}(f) = \sum_{l=0}^{L-1} \mathbf{a}_l e^{-j2\pi f \tau_l} \quad (2.2)$$

where  $\mathbf{a}_l$  is the  $p \times m$  set of complex path gains of the multipath fading channel,  $\tau_l$  is the set of path delays,  $f$  is the sampling frequency of the input signal.

In the next section the state-space model of the massive MIMO system is formulated.

## 2.3 State-Space Model

The state-space model is a mathematical representation or modelling of a physical system as a set of input and output data and state variables related by a first-order differential equation.

The dynamics of the massive multiple-input multiple-output linear-time invariant (MIMO LTI) system can be modelled using the continuous-time deterministic (i.e. ignoring the effects of the process noise and additive noise) state-space model as [64]:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad (2.3a)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \quad (2.3b)$$

where  $\mathbf{x}(t)$  is the  $n \times 1$  state vector,  $\dot{\mathbf{x}}(t)$  is the time derivative of  $\mathbf{x}(t)$ ,  $\mathbf{u}(t)$  is the  $m \times 1$  input vector,  $\mathbf{y}(t)$  is the  $p \times 1$  output vector,  $\mathbf{A}$  is the  $n \times n$  system matrix and it describes the dynamics of the system, i.e. the eigenvalues of the system,  $\mathbf{B}$  is the  $n \times m$  input matrix and it describes the linear transformation by which the inputs influence the next state,  $\mathbf{C}$  is the  $p \times n$  output matrix, and it describes how the state is transferred to

the output,  $D$  is the  $p \times m$  feed-forward matrix, and in continuous-time systems it is usually 0,  $\dot{\mathbf{x}}(t)$  is the derivative of  $\mathbf{x}(t)$ . An integer-order state-space system is stable if the eigenvalues are negative or have negative real parts if they are complex conjugate, i.e. they are located on the left half of the complex plane as shown in Figure 2.2.

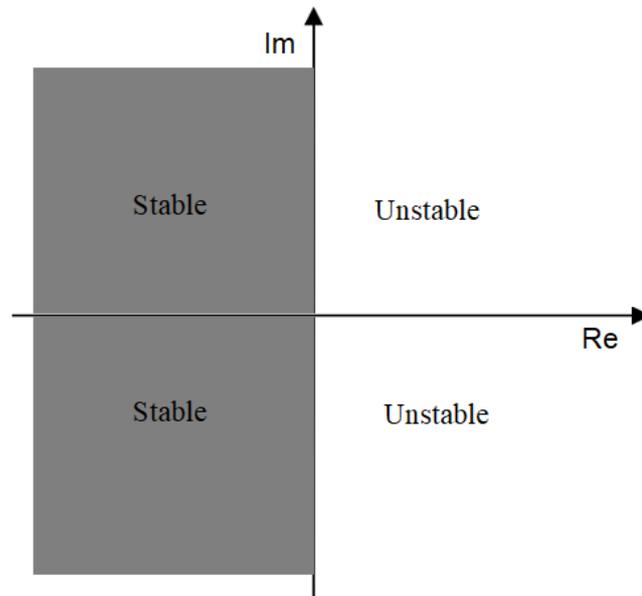


Figure 2. 2: Stability region for integer-order systems

Taking the Laplace transform, (2.3) becomes:

$$s\mathbf{X}(s) = \mathbf{A}\mathbf{X}(s) + \mathbf{B}\mathbf{U}(s) \quad (2.4a)$$

$$\mathbf{Y}(s) = \mathbf{C}\mathbf{X}(s) + \mathbf{U}(s) \quad (2.4b)$$

where  $\mathbf{X}(s) = \int_0^{\infty} \mathbf{x}(t)e^{-st} dt$ ,  $\mathbf{Y}(s) = \int_0^{\infty} \mathbf{y}(t)e^{-st} dt$  and  $\mathbf{U}(s) = \int_0^{\infty} \mathbf{u}(t)e^{-st} dt$ .

Rearranging (2.4) and after some manipulations the transfer function of the state-space model is expressed as:

$$\mathbf{G}(s) = \frac{\mathbf{Y}(s)}{\mathbf{U}(s)} = \mathbf{C}(s\mathbf{I}_n - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} \quad (2.5)$$

where  $I_n$  is an  $n \times n$  identity matrix.

The block diagram of the state-space model is as shown in Figure 2.3.

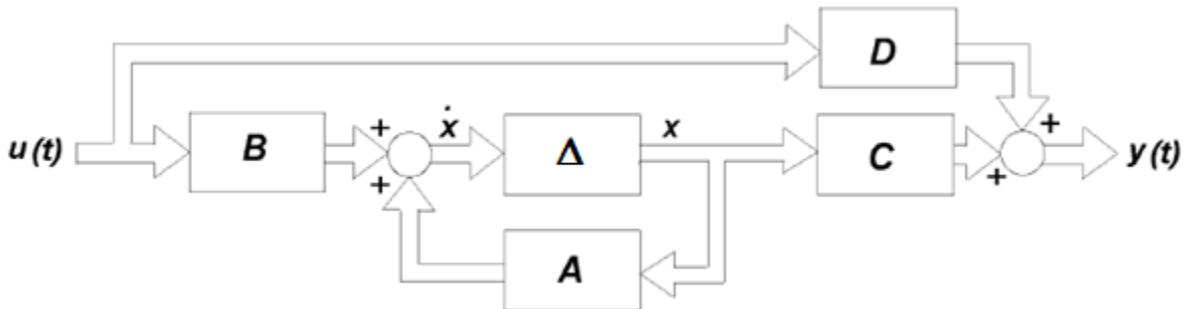


Figure 2. 3: Block diagram of a state-space model

State-space models have given rise to subspace system identification (SSI) algorithms, namely multiple-input multiple-output output-error state space (MOESP), numerical algorithm for subspace state-space system identification (N4SID) and canonical variate analysis (CVA), and these will be discussed in the following section.

## 2.4 Subspace Identification Methods

Subspace system algorithms are algorithms that can estimate a linear time-invariant state-space model, directly from the input-output data [65]. The advantage of SSI algorithms is that they can identify a system in a straightforward way using numerically robust computation tools such as linear quadratic (LQ) decomposition (representation of a matrix in a simpler form via orthogonal transformations) and singular value decomposition (SVD). SVD helps in decomposing a large matrix into a smaller matrix which still contains vital information of the large matrix thus allowing for simpler matrix calculations. SVD gives critical information about the matrix such as the eigenvalues from which the system order can be determined and the stability of the system can be determined. Another advantage of SSI algorithms is their computational efficiency. With SSI algorithms there is no need for an explicit model parameterisation, which for multi dimensional linear systems can be complex. Furthermore, SSI algorithms

can provide a direct way to control the complexity of the estimated channel model since the order of the channel model can be selected by the user by choosing the number of the largest singular values of the estimated extended observability matrix to include [2].

In subspace system identification the input and output data matrices are arranged in input and output Hankel matrices as:

$$\begin{bmatrix} \mathbf{U}_p \\ \mathbf{U}_f \end{bmatrix} = \begin{bmatrix} \mathbf{u}(0) & \mathbf{u}(1) & \cdots & \mathbf{u}(N-1) \\ \mathbf{u}(1) & \mathbf{u}(2) & \cdots & \mathbf{u}(N) \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{u}(i-1) & \mathbf{u}(i) & \cdots & \mathbf{u}(i+N-2) \\ \mathbf{u}(i) & \mathbf{u}(i+1) & \cdots & \mathbf{u}(i+N-1) \\ \mathbf{u}(i+1) & \mathbf{u}(i+2) & \cdots & \mathbf{u}(i+N) \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{u}(2i-1) & \mathbf{u}(2i) & \cdots & \mathbf{u}(2i+N-2) \end{bmatrix} \quad (2.6)$$

$$\begin{bmatrix} \mathbf{Y}_p \\ \mathbf{Y}_f \end{bmatrix} = \begin{bmatrix} \mathbf{y}(0) & \mathbf{y}(1) & \cdots & \mathbf{y}(N-1) \\ \mathbf{y}(1) & \mathbf{y}(2) & \cdots & \mathbf{y}(N) \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{y}(i-1) & \mathbf{y}(i) & \cdots & \mathbf{y}(i+N-2) \\ \mathbf{y}(i) & \mathbf{y}(i+1) & \cdots & \mathbf{y}(i+N-1) \\ \mathbf{y}(i+1) & \mathbf{y}(i+2) & \cdots & \mathbf{y}(i+N) \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{y}(2i-1) & \mathbf{y}(2i) & \cdots & \mathbf{y}(2i+N-2) \end{bmatrix} \quad (2.7)$$

respectively, where  $N$  is the number of data points,  $i$  is a user defined index which is larger than the system order but less than the number of data points, i.e.  $n < i < N$ ,  $\mathbf{u}(t)$  and  $\mathbf{y}(t)$  are known input and output data samples at time instant  $t$ .  $\mathbf{U}_p$  and  $\mathbf{Y}_p$  are the past input and past output Hankel matrices respectively and  $\mathbf{U}_f$  and  $\mathbf{Y}_f$  are the future input and future output Hankel matrices respectively. A Hankel matrix is a matrix with constant ascending skew-diagonal entries from left to right as show in Figure 2.4:

$$\left[ \mathbf{Y}_p \right] = \begin{bmatrix}
\mathbf{y}(0) & \mathbf{y}(1) & \mathbf{y}(2) & \mathbf{y}(3) & \cdots & \mathbf{y}(N-1) \\
\mathbf{y}(1) & \mathbf{y}(2) & \mathbf{y}(3) & \mathbf{y}(4) & \cdots & \mathbf{y}(N) \\
\mathbf{y}(2) & \mathbf{y}(3) & \mathbf{y}(4) & \mathbf{y}(5) & \cdots & \mathbf{y}(N+1) \\
\mathbf{y}(3) & \mathbf{y}(4) & \mathbf{y}(5) & \mathbf{y}(6) & \cdots & \mathbf{y}(N+2) \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\mathbf{y}(i-1) & \mathbf{y}(i) & \mathbf{y}(i+1) & \mathbf{y}(i+2) & \cdots & \mathbf{y}(i+N-2)
\end{bmatrix}$$

Figure 2. 4: Illustrating skew-diagonal entries

The input and output data matrices are divided into two halves, which represent the past input,  $\mathbf{U}_p$  and output,  $\mathbf{Y}_p$  and the future input,  $\mathbf{U}_f$  and output,  $\mathbf{Y}_f$  Hankel matrices with respect to a reference point  $t = i$ . It is important to stress that this division is artificially imposed on the data set, because all data involved in the identification process have already been acquired. The purpose of writing the matrix in this manner is to write the relations between the input, output and state sequences in a matrix form, which allows the identification problem to be placed in a projection geometry framework.

## 2.4.1 The Algebraic Geometric Framework

In this subsection we present definitions for the orthogonal and the oblique projections, and in our analysis we considered the following general matrices,  $A \in \mathbb{R}^{p \times j}$ ,  $B \in \mathbb{R}^{q \times j}$  and  $C \in \mathbb{R}^{r \times j}$ .

### 2.4.1.1 Orthogonal Projection

The orthogonal projection of the row space  $A$  onto the row space of  $B$  is defined as [65]:

$$A / B = AB^T (BB^T)^\dagger B = A\Pi_B \quad (2.8)$$

where  $(\cdot)^\dagger$  is Moore-Penrose pseudo-inverse,  $\Pi_B$  defines the operator of orthogonal projection which projects the row space of a matrix onto the row space of the matrix  $B$ , where  $\Pi_B = AB^T (BB^T)^\dagger B$ .

The row space of matrix  $A \in \mathbb{R}^{p \times j}$ , denoted as  $row(A)$  is the set of all linear combinations of the row vectors of  $A$ . The column space of matrix  $A \in \mathbb{R}^{p \times j}$ , denoted as  $col(A)$  is the set of all linear combinations of the column vectors of  $A$ . The projection of the row space  $A$  on the orthogonal space to the row space of  $B$  is defined as [65]:

$$A / B^\perp = A(I - \Pi_B) = A(I - B^T (BB^T)^\dagger B) = A\Pi_{B^\perp} \quad (2.9)$$

where  $B^\perp$  denotes a base of the orthogonal space to the row space of  $B$ ,  $\Pi_{B^\perp}$  defines the geometric operator that projects the row space of a matrix onto the orthogonal complement to the row space of the matrix  $B$ .

The projections  $A / B$  and  $A / B^\perp$  decompose the matrix  $A$  into two matrices, whose row spaces are mutually orthogonal and this is as shown in Figure 2.5:

$$A = A / B + A / B^\perp \quad (2.10)$$

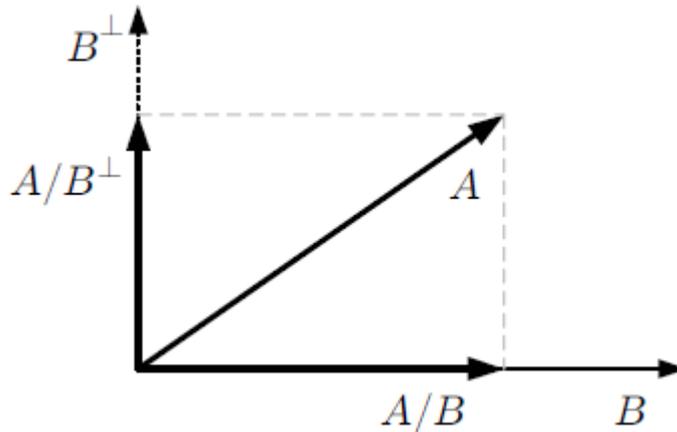


Figure 2. 5: Orthogonal projection

A numerically efficient and robust computation of the orthogonal projection can be done by LQ decomposition as follows [66]:

$$\begin{bmatrix} B \\ A \end{bmatrix} = LQ = \begin{bmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} \quad (2.11)$$

The LQ decomposition of a matrix  $A$  is the same as the QR decomposition of its transpose  $A^T$ , i.e.  $A = (A^T)^T = (QR)^T = LQ^T$ .

Using (2.11), the orthogonal projections can then be written as:

$$A / B = L_{21} Q_1 \quad (2.12)$$

$$A / B^\perp = L_{22} Q_2 \quad (2.13)$$

For noise free data, the subspace identification reduces to the orthogonal projection. The orthogonal projection is used to minimise the influence of the disturbances from the joint space.

#### 2.4.1.2 Oblique Projection

The oblique projection of the row space  $A$  onto the row space of  $C$  along the row space of  $B$  is defined as [65]:

$$A /_B C = [A / B^\perp] [C / B^\perp]^\dagger C = A \Pi_{B^\perp} + (C \Pi_{B^\perp})^\dagger C \quad (2.14)$$

This result can be derived from an orthogonal projection of the row space  $A$  to the row space of  $\begin{bmatrix} B \\ C \end{bmatrix}$ .

Using LQ decomposition we can write:

$$\begin{bmatrix} B \\ C \\ A \end{bmatrix} = \begin{bmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \end{bmatrix} \quad (2.15)$$

Using (2.15), the orthogonal projections can be written as:

$$A /_B C = L_{32} L_{22}^{-1} C = L_{32} L_{22}^{-1} [L_{21} \quad L_{22}] \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} \quad (2.16)$$

The oblique projection decomposes the matrix  $A$  into three matrices and this is as shown in Figure 2.6:

$$A = A /_B C + A /_C B + A / \begin{bmatrix} B \\ C \end{bmatrix} \quad (2.17)$$

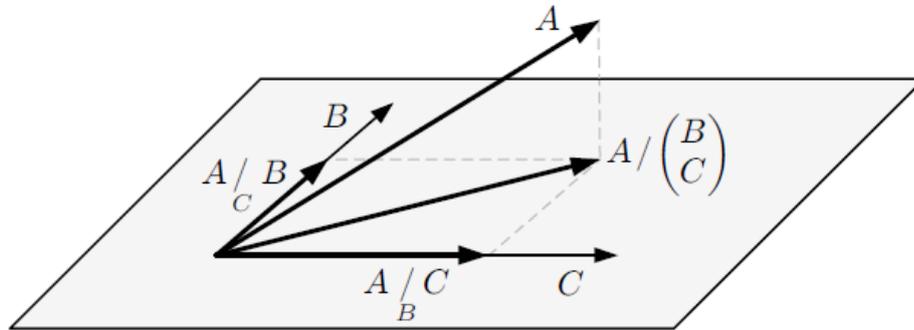


Figure 2. 6: Oblique projection

The oblique projection is used to simultaneously minimise the influence of the disturbances and to minimise the influence of the deterministic measurable signal (the input). The oblique projection is mainly used in the N4SID algorithm to obtain the subspace generated only by the state sequence, which is later used to obtain the model parameters.

The subspace system identification procedure assumes that:

- i) The system is observable, in the sense that the time evolution of all modes or states of interest is reflected at the measured output/ actual output. A system is observable if any state  $\mathbf{x}(t_0)$  can be determined from the observation of  $\mathbf{y}(t)$  and the knowledge of the input  $\mathbf{u}(t)$  within the time limit  $t_0 \leq t \leq t_f$ .
- ii) The system should also be controllable, in the sense that the input excitation affects all modes of interest. In this manner, all the modes or states to be included in the identification process are related to the input-output behavior of the system and can thus be modeled from experimental input-output data. A system is controllable if it is possible to establish a nonrestricted control vector which can lead the system from an initial state,  $\mathbf{x}(t_0)$  to another final state  $\mathbf{x}(t_f)$ , within the time limit  $t_0 \leq t \leq t_f$ .
- iii) The input signal is rich enough to excite all the modes that are to be included in the model (persistent excitation).

In the next subsection an overview of the different SSI algorithms is presented. In the discussion on the different SSI algorithms the discrete-time deterministic system of order  $n$  described by the following state-space equations is considered:

$$\mathbf{x}(t+1) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad (2.18a)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \quad (2.18b)$$

where  $\mathbf{x}(t)$  is the  $n \times 1$  state vector,  $\mathbf{x}(t+1)$  is the next state vector,  $\mathbf{u}(t)$  is the  $m \times 1$  input vector,  $\mathbf{y}(t)$  is the  $p \times 1$  output vector,  $\mathbf{A}$  is the  $n \times n$  system matrix and it describes the dynamics of the system, i.e. the eigenvalues of the system,  $\mathbf{B}$  is the  $n \times m$  input matrix and it describes the linear transformation by which the inputs influence the next state,  $\mathbf{C}$  is the  $p \times n$  output matrix, and it describes how the state is transferred to the output,  $\mathbf{D}$  is the  $p \times m$  feed-forward matrix.

## 2.4.2 The Multiple-Input Multiple-Output Output-Error State Space Algorithm

The MOESP algorithm relies on the knowledge of the Hankel matrices and is based on the orthogonal projection. The MOESP algorithm uses the LQ decomposition to decompose matrix  $W_p$  to another matrix with orthogonal rows as a matrix with a row space that is equal to the column space.

where

$$W_p = \begin{bmatrix} U_p \\ Y_p \end{bmatrix} \quad (2.19)$$

LQ decomposition represents an orthogonal projection of the row space of  $Y_p$  to the row space of  $U_p$ . The singular value decomposition (SVD) is then used to obtain the order of the system, and estimate of the observability matrix from which the system matrices, i.e.  $A$ ,  $B$ ,  $C$  and  $D$  are then calculated. The MOESP algorithm can be summarised as follows:

1. First arrange the input-output data into Hankel matrices,  $U_p$  and  $Y_p$  and then use these matrices to construct matrix  $W_p$ .
2. Apply LQ decomposition to matrix  $W_p$  to obtain:

$$\begin{bmatrix} U_p \\ Y_p \end{bmatrix} = \begin{bmatrix} L_{11} & \mathbf{0} \\ L_{21} & L_{22} \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} \quad (2.20)$$

where  $L_{11} \in \mathbb{R}^{im \times im}$  and  $L_{22} \in \mathbb{R}^{ip \times ip}$  are lower triangular,  $L_{21} \in \mathbb{R}^{ip \times im}$  and  $Q_1 \in \mathbb{R}^{N \times im}$  and  $Q_2 \in \mathbb{R}^{N \times ip}$  are orthogonal.

3. Compute the SVD of  $L_{22}$  as:

$$L_{22} = [U_1 \quad U_2] \begin{bmatrix} S_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} V_1^H \\ V_2^H \end{bmatrix} \quad (2.21)$$

where  $S_1 \in \mathbb{R}^{n \times n}$  is a diagonal matrix containing the singular values different from zero,  $U_1 \in \mathbb{R}^{ip \times n}$ ,  $V_1$  are unitary matrices and  $U_2 \in \mathbb{R}^{ip \times (ip-n)}$ . The order of the system,  $n$  can be estimated from  $S_1$  as the number of singular values different from zero.

4. Estimate the extended observability matrix,  $\Gamma_i$  as:

$$\hat{\Gamma}_i = U_1 S_1^{\frac{1}{2}} \quad (2.22)$$

5. Estimate matrix  $C$  which can be obtained directly from the first  $p$  rows of  $\hat{\Gamma}_i$  as:

$$\hat{C} = \hat{\Gamma}_i(1:p,:) \quad (2.23)$$

6. Estimate matrix  $A$  by using the shift property as:

$$\underline{\Gamma}_i \hat{A} = \bar{\Gamma}_i \quad (2.24a)$$

$$\hat{A} = \underline{\Gamma}_i^\dagger \bar{\Gamma}_i \quad (2.24b)$$

where  $\underline{\Gamma}_i$  is  $\hat{\Gamma}_i$  with the last  $p$  rows removed and  $\bar{\Gamma}_i$  is  $\hat{\Gamma}_i$  with the first  $p$  rows removed.

The estimate of  $A$  can also be expressed as:

$$\hat{\Gamma}_i(1:end-p,:) \hat{A} = \hat{\Gamma}_i(1+p:end,:) \quad (2.25)$$

7. Estimate matrices  $\mathbf{B}$  and  $\mathbf{D}$ . Once the estimates of  $\mathbf{A}$  and  $\mathbf{C}$  are known, the estimates of  $\mathbf{B}$  and  $\mathbf{D}$  are obtained using the following input-output equation:

$$\mathbf{Y}_p = \hat{\Gamma}_i \mathbf{X}_p + \mathbf{\Omega}_i \mathbf{U}_p \quad (2.26)$$

where

$$\mathbf{X}_p = [\mathbf{x}(i) \quad \mathbf{x}(i+1) \quad \cdots \quad \mathbf{x}(i+N-1)] \quad (2.27)$$

$\hat{\Gamma}_i$  is the estimated extended observability matrix written as:

$$\hat{\Gamma}_i = [\hat{\mathbf{C}} \quad \hat{\mathbf{C}}\hat{\mathbf{A}} \quad \cdots \quad \hat{\mathbf{C}}\hat{\mathbf{A}}^{i-2} \quad \hat{\mathbf{C}}\hat{\mathbf{A}}^{i-1}]^T \in \mathbb{R}^{ip \times n} \quad (2.28)$$

and

$\mathbf{\Omega}_i$  is the block Toeplitz matrix, written as:

$$\mathbf{\Omega}_i = \begin{bmatrix} \hat{\mathbf{D}} & \mathbf{0} & \cdots & \mathbf{0} \\ \hat{\mathbf{C}}\hat{\mathbf{B}} & \hat{\mathbf{D}} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{\mathbf{C}}\hat{\mathbf{A}}^{i-2}\mathbf{B} & \hat{\mathbf{C}}\hat{\mathbf{A}}^{i-3}\mathbf{B} & \cdots & \hat{\mathbf{D}} \end{bmatrix} \in \mathbb{R}^{ip \times im} \quad (2.29)$$

The system given by (2.18) is observable if and only if matrix defined by (2.28) is full-rank (i.e. all the rows and columns are linearly independent).

Premultiplying (2.26) by  $\hat{\Gamma}_i^\perp$  to eliminate  $\hat{\Gamma}_i \mathbf{X}_p$  and postmultiplying it by  $\mathbf{U}_p^\dagger = \mathbf{U}_p^H (\mathbf{U}_p \mathbf{U}_p^H)^{-1}$  yields:

$$\hat{\Gamma}_i^\perp \mathbf{Y}_p \mathbf{U}_p^\dagger = \hat{\Gamma}_i^\perp \hat{\Gamma}_i \mathbf{X}_p \mathbf{U}_p^\dagger + \hat{\Gamma}_i^\perp \mathbf{\Omega}_i \mathbf{U}_p \mathbf{U}_p^\dagger \quad (2.30)$$

where  $\hat{\Gamma}_i^\perp$  is a full row rank matrix (i.e. all the rows and columns are linearly independent) satisfying  $\hat{\Gamma}_i^\perp \hat{\Gamma}_i = \mathbf{0}$ . The equation then simplifies to:

$$\hat{\Gamma}_i^\perp \mathbf{Y}_p \mathbf{U}_p^\dagger = \hat{\Gamma}_i^\perp \mathbf{\Omega}_i \quad (2.31)$$

Denoting  $\mathbf{L} = \hat{\Gamma}_i^\perp$  and  $\mathbf{M} = \hat{\Gamma}_i^\perp \mathbf{Y}_p \mathbf{U}_p^\dagger$ , (2.31) can be rewritten as:

$$\mathbf{M} = \mathbf{L}\mathbf{\Omega}_i \quad (2.32)$$

where  $\mathbf{L} = (\mathbf{L}_1 \ \mathbf{L}_2 \ \cdots \ \mathbf{L}_i)$  where  $\mathbf{L}_i \in \mathbb{R}^{(ip-n) \times p}$  and  $\mathbf{M} = (\mathbf{M}_1 \ \mathbf{M}_2 \ \cdots \ \mathbf{M}_i)$  where  $\mathbf{M}_i \in \mathbb{R}^{(ip-n) \times m}$ . Equation (2.32) can be written as:

$$\begin{pmatrix} \mathbf{M}_1 \\ \mathbf{M}_2 \\ \vdots \\ \mathbf{M}_i \end{pmatrix} = \begin{pmatrix} \mathbf{L}_1 & \mathbf{L}_2 & \cdots & \mathbf{L}_{i-1} & \mathbf{L}_i \\ \mathbf{L}_2 & \mathbf{L}_3 & \cdots & \mathbf{L}_i & \mathbf{0} \\ \mathbf{L}_3 & \mathbf{L}_4 & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{L}_i & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{I}_i & \mathbf{0} \\ \mathbf{0} & \hat{\Gamma}_i \end{pmatrix} \begin{pmatrix} \hat{\mathbf{D}} \\ \hat{\mathbf{B}} \end{pmatrix} \quad (2.33)$$

which can be solved using least squares.

Having discussed on the MOESP algorithm, the next SSI algorithm to discuss on is the N4SID algorithm.

### 2.4.3 The Numerical Algorithm for Subspace State-Space System Identification

Compared to the MOESP algorithm, the N4SID algorithm is based on the oblique projection of subspaces generated by the input and output Hankel matrices of the system. The N4SID algorithm projects  $\mathbf{Y}_f$  along  $\mathbf{U}_f$  onto  $\mathbf{W}_p$  i.e.  $\mathbf{Y}_f /_{\mathbf{U}_f} \mathbf{W}_p$ , where

$$\mathbf{W}_p = \begin{bmatrix} \mathbf{U}_p \\ \mathbf{Y}_p \end{bmatrix} \quad (2.34)$$

Geometric and other mathematics tools of linear algebra such as the singular value decomposition (SVD) can be used to obtain the order of the system, an estimate of the observability matrix from which the system matrices, i.e.  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$  and  $\mathbf{D}$  are then calculated. With reference to (2.18) and using Hankel matrices we can write the following input-output equations:

$$\mathbf{X}_f = \mathbf{A}^i \mathbf{X}_p + \Delta_i \mathbf{U}_p \quad (2.35)$$

$$\mathbf{Y}_p = \mathbf{\Gamma}_i \mathbf{X}_p + \mathbf{\Omega}_i \mathbf{U}_p \quad (2.36)$$

$$\mathbf{Y}_f = \mathbf{\Gamma}_i \mathbf{X}_f + \mathbf{\Omega}_i \mathbf{U}_f \quad (2.37)$$

where

$$\mathbf{X}_p = [\mathbf{x}(0) \quad \mathbf{x}(1) \quad \cdots \quad \mathbf{x}(i-1)] \quad (2.38)$$

$$\mathbf{X}_f = [\mathbf{x}(i) \quad \mathbf{x}(i+1) \quad \cdots \quad \mathbf{x}(i+N-1)] \quad (2.39)$$

$\Delta_i$  is the extended controllability matrix written as:

$$\Delta_i = [\mathbf{A}^{i-1} \mathbf{B} \quad \mathbf{A}^{i-2} \mathbf{B} \quad \cdots \quad \mathbf{B}] \quad (2.40)$$

The system given by (2.18) is controllable if and only if the matrix defined by (2.40) is full-rank. Before defining the N4SID algorithm the following assumptions are stated:

1. The inputs excitation is persistent of order  $2i$ .

2. The user defined weighting matrices  $\mathbf{W}_1 \in \mathbb{R}^{p_i \times p_i}$  and  $\mathbf{W}_2 \in \mathbb{R}^{j \times j}$  are such that  $\mathbf{W}_1$  is full-rank and  $\mathbf{W}_2$  satisfies:

$$\text{rank}(\mathbf{W}_p) = \text{rank}(\mathbf{W}_p \mathbf{W}_2) \quad (2.41)$$

where  $i$  and  $j$  are arbitrary constants such that  $i + j \leq N$ .

The N4SID algorithm can then be summarised as follows:

1. First arrange the input-output data into Hankel matrices,  $\mathbf{U}_p$ ,  $\mathbf{U}_f$ ,  $\mathbf{Y}_p$  and  $\mathbf{Y}_f$  and then use these matrices to construct matrix  $\mathbf{W}_p$  as:

$$\mathbf{W}_p = \begin{bmatrix} \mathbf{U}_p \\ \mathbf{Y}_p \end{bmatrix} \quad (2.34)$$

2. Calculate the oblique projection,  $\mathbf{O}_i$  as:

$$\mathbf{O}_i = \mathbf{Y}_f /_{\mathbf{U}_f} \mathbf{W}_p \quad (2.42)$$

Under the assumption that the input is rich enough to excite all the modes that are to be included in the model (persistent excitation), the estimate of the state sequence can be obtained from:

$$\mathbf{O}_i = \Gamma_i \mathbf{X}_f \quad (2.43)$$

3. Compute the SVD of the weighted oblique projection. Start by multiplying  $\mathbf{O}_i$  by  $\mathbf{W}_1$  on the left and  $\mathbf{W}_2$  on the right to improve the estimation of (2.43):

$$\mathbf{W}_1 \mathbf{O}_i \mathbf{W}_2 = [\mathbf{U}_1 \quad \mathbf{U}_2] \begin{bmatrix} \mathbf{S}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^H \\ \mathbf{V}_2^H \end{bmatrix} = \mathbf{U}_1 \mathbf{S}_1 \mathbf{V}_1^H \quad (2.44)$$

where  $S_1$  is a diagonal matrix containing singular values different from zero and  $U_1$ ,  $V_1$  are unitary matrices. The order of the system,  $n$  can be estimated from  $S_1$  as the number of singular values different from zero.

4. Estimate the extended observability matrix,  $\Gamma_i$  as:

$$\hat{\Gamma}_i = W_1^{-1} U_1 S_1^{\frac{1}{2}} T \quad (2.45)$$

where  $T \in \mathbb{R}^{n \times n}$  is a non-singular similarity transformation matrix.

5. Estimate the state,  $X_f$  using (2.43) as:

$$\hat{X}_f = \Gamma_i^\dagger O_i \quad (2.46)$$

6. Estimate matrix  $C$  which can be obtained directly from the first  $p$  rows of  $\hat{\Gamma}_i$  as:

$$\hat{C} = \hat{\Gamma}_i(1:p,:) \quad (2.23)$$

7. Estimate matrix  $A$  by using the shift property as:

$$\underline{\Gamma}_i \hat{A} = \bar{\Gamma}_i \quad (2.24a)$$

$$\hat{A} = \underline{\Gamma}_i^\dagger \bar{\Gamma}_i \quad (2.24b)$$

where  $\underline{\Gamma}_i$  is  $\hat{\Gamma}_i$  with the last  $p$  rows removed and  $\bar{\Gamma}_i$  is  $\hat{\Gamma}_i$  with the first  $p$  rows removed.

The estimate of  $A$  can also be expressed as:

$$\hat{\Gamma}_i(1:end-p,:) \hat{A} = \hat{\Gamma}_i(1+p:end,:) \quad (2.25)$$

8. Estimate matrices  $B$  and  $D$ . Once the estimates of  $A$  and  $C$  are known, the estimates of  $B$  and  $D$  are obtained using the following input-output equation:

$$Y_f = \hat{\Gamma}_i X_f + \Omega_i U_f \quad (2.26)$$

Premultiplying (2.26) by  $\hat{\Gamma}_i^\perp$  to eliminate  $\hat{\Gamma}_i X_f$  and postmultiplying it by  $U_f^\dagger = U_f^H (U_f U_f^H)^{-1}$  yields:

$$\hat{\Gamma}_i^\perp Y_f U_f^\dagger = \hat{\Gamma}_i^\perp \hat{\Gamma}_i X_f U_f^\dagger + \hat{\Gamma}_i^\perp \Omega_i U_f U_f^\dagger \quad (2.30)$$

where  $\hat{\Gamma}_i^\perp$  is a full row rank matrix (i.e. all the rows and columns are linearly independent) satisfying  $\hat{\Gamma}_i^\perp \hat{\Gamma}_i = \mathbf{0}$ . The equation then simplifies to:

$$\hat{\Gamma}_i^\perp Y_f U_f^\dagger = \hat{\Gamma}_i^\perp \Omega_i \quad (2.31)$$

Denoting  $L = \hat{\Gamma}_i^\perp$  and  $M = \hat{\Gamma}_i^\perp Y_f U_f^\dagger$ , (2.31) can be rewritten as:

$$M = L \Omega_i \quad (2.32)$$

where  $L = (L_1 \ L_2 \ \dots \ L_i)$  and  $M = (M_1 \ M_2 \ \dots \ M_i)$ . Equation (2.32) can be written as:

$$\begin{pmatrix} \mathbf{M}_1 \\ \mathbf{M}_2 \\ \vdots \\ \mathbf{M}_i \end{pmatrix} = \begin{pmatrix} \mathbf{L}_1 & \mathbf{L}_2 & \cdots & \mathbf{L}_{i-1} & \mathbf{L}_i \\ \mathbf{L}_2 & \mathbf{L}_3 & \cdots & \mathbf{L}_i & \mathbf{0} \\ \mathbf{L}_3 & \mathbf{L}_4 & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{L}_i & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{I}_i & \mathbf{0} \\ \mathbf{0} & \hat{\Gamma}_i \end{pmatrix} \begin{pmatrix} \hat{\mathbf{D}} \\ \hat{\mathbf{B}} \end{pmatrix} \quad (2.33)$$

which can be solved using least squares.

The discussion on the SSI algorithms is concluded by just making mention of the CVA algorithm.

#### 2.4.4 The Canonical Variate Analysis Algorithm

In the canonical variate analysis (CVA) algorithm, canonical correlation analysis is performed on the past Hankel matrix,  $\mathbf{W}_p$  and the future output Hankel matrix,  $\mathbf{Y}_f$ , where  $\mathbf{W}_p = [\mathbf{U}_p \quad \mathbf{Y}_p]^T$  is a combination of past input,  $\mathbf{U}_p$  and past output,  $\mathbf{Y}_p$  Hankel matrices [66]. Generally speaking CVA is a dimensionality reduction algorithm that maximises the correlation between two data sets [67]. The dominant canonical variate of these two data sets obtained based on the Markov process and the maximum likelihood function gives an estimate of the state variables. The Markov process is a random process whose future is independent of the past values given the present values. The system matrices, i.e.  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$  and  $\mathbf{D}$  are then estimated using the least squares method [68].

The key idea of SSI methods is to estimate the extended observability matrix through the projection of future input-output data onto past input-output data based on the relationship between Hankel matrices of the input and output data.

## 2.4.5 Constructing Data Matrices

In constructing the data matrices for the continuous-time deterministic system, (2.3) and the MOESP algorithm are considered. By computing the successive derivatives of (2.3b) the data equations are constructed as follows [69]:

First-order derivative yields:

$$\frac{dy(t)}{dt} = \mathbf{C} \frac{dx(t)}{dt} + \mathbf{D} \frac{du(t)}{dt} \quad (2.47)$$

Then substituting (2.3a) into (2.47) yields:

$$\begin{aligned} \frac{dy(t)}{dt} &= \mathbf{C} (\mathbf{A}x(t) + \mathbf{B}u(t)) + \mathbf{D} \frac{du(t)}{dt} \\ \frac{dy(t)}{dt} &= \mathbf{C}\mathbf{A}x(t) + \mathbf{C}\mathbf{B}u(t) + \mathbf{D} \frac{du(t)}{dt} \end{aligned} \quad (2.48)$$

Second-order derivative yields:

$$\frac{d^2y(t)}{dt^2} = \mathbf{C}\mathbf{A} \frac{dx(t)}{dt} + \mathbf{C}\mathbf{B} \frac{du(t)}{dt} + \mathbf{D} \frac{d^2u(t)}{dt^2} \quad (2.49)$$

Then substituting (2.3a) into (2.49) to yields:

$$\begin{aligned} \frac{d^2y(t)}{dt^2} &= \mathbf{C}\mathbf{A} (\mathbf{A}x(t) + \mathbf{B}u(t)) + \mathbf{C}\mathbf{B} \frac{du(t)}{dt} + \mathbf{D} \frac{d^2u(t)}{dt^2} \\ \frac{d^2y(t)}{dt^2} &= \mathbf{C}\mathbf{A}^2x(t) + \mathbf{C}\mathbf{A}\mathbf{B}u(t) + \mathbf{C}\mathbf{B} \frac{du(t)}{dt} + \mathbf{D} \frac{d^2u(t)}{dt^2} \end{aligned} \quad (2.50)$$

Third-order derivative yields:

$$\frac{d^3y(t)}{dt^3} = \mathbf{C}\mathbf{A}^2 \frac{dx(t)}{dt} + \mathbf{C}\mathbf{A}\mathbf{B} \frac{du(t)}{dt} + \mathbf{C}\mathbf{B} \frac{d^2u(t)}{dt^2} + \mathbf{D} \frac{d^3u(t)}{dt^3} \quad (2.51)$$

Then substituting (2.3a) into (2.51) yields:

$$\begin{aligned}\frac{d^3 \mathbf{y}(t)}{dt^3} &= \mathbf{CA}^2 (\mathbf{Ax}(t) + \mathbf{Bu}(t)) + \mathbf{CAB} \frac{d\mathbf{u}(t)}{dt} + \mathbf{CB} \frac{d^2 \mathbf{u}(t)}{dt^2} + \mathbf{D} \frac{d^3 \mathbf{u}(t)}{dt^3} \\ \frac{d^3 \mathbf{y}(t)}{dt^3} &= \mathbf{CA}^3 \mathbf{x}(t) + \mathbf{CA}^2 \mathbf{Bu}(t) + \mathbf{CAB} \frac{d\mathbf{u}(t)}{dt} + \mathbf{CB} \frac{d^2 \mathbf{u}(t)}{dt^2} + \mathbf{D} \frac{d^3 \mathbf{u}(t)}{dt^3}\end{aligned}\quad (2.52)$$

and so on till the  $(i-1)$ -order derivative.

Stacking these derivatives into a column vector, the input-output relationship is written as:

$$\underbrace{\begin{bmatrix} \mathbf{y}(t) \\ \frac{d\mathbf{y}(t)}{dt} \\ \vdots \\ \frac{d^{(i-1)}\mathbf{y}(t)}{dt^{(i-1)}} \end{bmatrix}}_{\mathbf{Y}_{0i-1}(t)} = \underbrace{\begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{(i-1)} \end{bmatrix}}_{\mathbf{\Gamma}_i} \mathbf{x}(t) + \underbrace{\begin{bmatrix} \mathbf{D} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CB} & \mathbf{D} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{(i-2)}\mathbf{B} & \cdots & \mathbf{CB} & \mathbf{D} \end{bmatrix}}_{\mathbf{\Omega}_i} \underbrace{\begin{bmatrix} \mathbf{u}(t) \\ \frac{d\mathbf{u}(t)}{dt} \\ \vdots \\ \frac{d^{(i-1)}\mathbf{u}(t)}{dt^{(i-1)}} \end{bmatrix}}_{\mathbf{U}_{0i-1}(t)} \quad (2.53)$$

In compact form (2.53) can be written as:

$$\mathbf{Y}_{0i-1}(t) = \mathbf{\Gamma}_i \mathbf{x}(t) + \mathbf{\Omega}_i \mathbf{U}_{0i-1}(t) \quad (2.54)$$

where the notation  $(\bullet)_{0i-1}$  denotes the zero<sup>th</sup> order derivative to the  $(i-1)$ <sup>th</sup> order derivative,  $\mathbf{\Gamma}_i$  and  $\mathbf{\Omega}_i$  are the observability and Toeplitz matrices respectively.

Since the N4SID algorithm can lead in some cases to ambiguous estimation due to the use of oblique projection [65] the focus of this thesis will be on system identification using the MOESP algorithm which is based on orthogonal projection.

Sampling (2.54) at sampling instances  $t_1, t_2, \dots, t_N$ , the input-output matrices and the state matrix can be rewritten as:

$$\mathbf{U}_{0|i-1, N} = \begin{bmatrix} \mathbf{u}(t_1) & \mathbf{u}(t_2) & \cdots & \mathbf{u}(t_N) \\ \frac{d\mathbf{u}(t_1)}{dt} & \frac{d\mathbf{u}(t_2)}{dt} & \cdots & \frac{d\mathbf{u}(t_N)}{dt} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{d^{(i-1)}\mathbf{u}(t_1)}{dt^{(i-1)}} & \frac{d^{(i-1)}\mathbf{u}(t_2)}{dt^{(i-1)}} & \cdots & \frac{d^{(i-1)}\mathbf{u}(t_N)}{dt^{(i-1)}} \end{bmatrix} \quad (2.55)$$

,

$$\mathbf{Y}_{0|i-1, N} = \begin{bmatrix} \mathbf{y}(t_1) & \mathbf{y}(t_2) & \cdots & \mathbf{y}(t_N) \\ \frac{d\mathbf{y}(t_1)}{dt} & \frac{d\mathbf{y}(t_2)}{dt} & \cdots & \frac{d\mathbf{y}(t_N)}{dt} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{d^{(i-1)}\mathbf{y}(t_1)}{dt^{(i-1)}} & \frac{d^{(i-1)}\mathbf{y}(t_2)}{dt^{(i-1)}} & \cdots & \frac{d^{(i-1)}\mathbf{y}(t_N)}{dt^{(i-1)}} \end{bmatrix} \quad (2.56)$$

and

$$\mathbf{X}_N = [\mathbf{x}(t_1) \quad \mathbf{x}(t_2) \quad \cdots \quad \mathbf{x}(t_N)] \quad (2.57)$$

where the notation  $(\cdot)_{0|i-1, N}$  denotes the zero<sup>th</sup> order derivative to the  $(i-1)$ <sup>th</sup> order derivative for  $N$  samples.

Using the above equations, the sampled data equation is rewritten as:

$$\mathbf{Y}_{0|i-1, N} = \mathbf{\Gamma}_i \mathbf{X}_N + \mathbf{\Omega}_i \mathbf{U}_{0|i-1, N} \quad (2.58)$$

and the data matrix,  $\mathbf{W}_{0|i-1, N}$  can be constructed as:

$$\mathbf{W}_{0|i-1,N} = \begin{bmatrix} \mathbf{U}_{0|i-1,N} \\ \mathbf{Y}_{0|i-1,N} \end{bmatrix} \quad (2.59)$$

Unlike in discrete-time system identification, the time-derivatives of the input-output data are generally not measured which means that the input-output matrices  $\mathbf{U}_{0|i-1,N}$  and  $\mathbf{Y}_{0|i-1,N}$  are not known. As a result, the classical subspace methods originally developed for the identification of discrete-time models cannot be directly adapted for the identification of continuous-time models. To address this problem we propose the use of the Poisson moment functional (PMF) approach as presented in [70] where they used the Poisson pulse function to estimate  $\mathbf{U}_{0|i-1,N}$  and  $\mathbf{Y}_{0|i-1,N}$ . The Poisson moment functional approach for the continuous-time integer-order state-space model is outlined in Appendix A2.

Applying PMF to the data matrix,  $\mathbf{W}_{0|i-1,N}$  in (2.59) the PMF filtered data can be expressed as:

$$\mathbf{P}_t^r [\mathbf{W}_{0|i-1,N}] = \begin{bmatrix} \mathbf{P}_t^r [\mathbf{U}_{0|i-1,N}] \\ \mathbf{P}_t^r [\mathbf{Y}_{0|i-1,N}] \end{bmatrix} \quad (2.60)$$

Having filtered the input-output data, the MOESP identification algorithm outlined for discrete-time system identification can then be applied to the PMF filtered input-output data. Applying LQ decomposition to matrix (2.60) yields:

$$\begin{bmatrix} \mathbf{P}_t^r [\mathbf{U}_{0|i-1,N}] \\ \mathbf{P}_t^r [\mathbf{Y}_{0|i-1,N}] \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{11} & \mathbf{0} \\ \mathbf{L}_{21} & \mathbf{L}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \end{bmatrix} \quad (2.61)$$

Using (2.61) compute the SVD of  $\mathbf{L}_{22}$  as:

$$\mathbf{L}_{22} = [\mathbf{U}_1 \quad \mathbf{U}_2] \begin{bmatrix} \mathbf{S}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^H \\ \mathbf{V}_2^H \end{bmatrix} \quad (2.21)$$

where  $S_1$  is a diagonal matrix containing the singular values different from zero and the order of the system,  $n$  can be estimated from  $S_1$  as the number of singular values different from zero.

The extended observability matrix,  $\Gamma_i$  can then be estimated from (2.21) as:

$$\hat{\Gamma}_i = U_1 S_1^{\frac{1}{2}} \quad (2.22)$$

The procedure for estimating matrices  $A$ ,  $B$ ,  $C$  and  $D$  is as discussed in the discrete-time MOESP algorithm.

In our discussion so far we ignored the effects of the noise when identifying the ISI channel for the massive MIMO system, but in real life system identification the effects of this noise have to be considered. The next section discusses the influence of the additive white Gaussian noise in the continuous-time integer-order massive MIMO system identification.

## 2.5 Integer-Order System Identification with Measurement Noise Considered

In this case, integer-order system identification is performed having considered the measurement noise. Using continuous-time integer-order state-space modelling, the dynamics of the linear time invariant length- $L$  ISI channel massive MIMO system can be expressed as:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad (2.62a)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) + \mathbf{n}(t) \quad (2.62b)$$

where  $\mathbf{x}(t)$  is the  $n \times 1$  state vector,  $\dot{\mathbf{x}}(t)$  is the time derivative of  $\mathbf{x}(t)$ ,  $\mathbf{u}(t)$  is the  $m \times 1$  input vector,  $\mathbf{y}(t)$  is the  $p \times 1$  output vector,  $\mathbf{n}(t)$  is the  $p \times 1$  additive white

Gaussian noise vector,  $\mathbf{A}$  is the  $n \times n$  system matrix and it describes the dynamics of the system, i.e. the eigenvalues of the system,  $\mathbf{B}$  is the  $n \times m$  input matrix and it describes the linear transformation by which the inputs influence the next state,  $\mathbf{C}$  is the  $p \times n$  output matrix, and it describes how the state is transferred to the output,  $\mathbf{D}$  is the  $p \times m$  feed-forward matrix, and in continuous-time systems it is usually 0,  $\dot{\mathbf{x}}(t)$  is the derivative of  $\mathbf{x}(t)$

Applying the Laplace transform to (2.62) results in:

$$\begin{aligned} s\mathbf{X}(s) &= \mathbf{A}\mathbf{X}(s) + \mathbf{B}\mathbf{U}(s) \\ \mathbf{Y}(s) &= \mathbf{C}\mathbf{X}(s) + \mathbf{D}\mathbf{U}(s) + \mathbf{N}(s) \end{aligned} \quad (2.63)$$

First-order derivative yields:

$$\frac{d\mathbf{y}(t)}{dt} = \mathbf{C} \frac{d\mathbf{x}(t)}{dt} + \mathbf{D} \frac{d\mathbf{u}(t)}{dt} + \mathbf{D} \frac{d\mathbf{n}(t)}{dt} \quad (2.64)$$

Then substituting (2.62a) into (2.64) yields:

$$\begin{aligned} \frac{d\mathbf{y}(t)}{dt} &= \mathbf{C}(\mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)) + \mathbf{D} \frac{d\mathbf{u}(t)}{dt} + \mathbf{D} \frac{d\mathbf{n}(t)}{dt} \\ \frac{d\mathbf{y}(t)}{dt} &= \mathbf{C}\mathbf{A}\mathbf{x}(t) + \mathbf{C}\mathbf{B}\mathbf{u}(t) + \mathbf{D} \frac{d\mathbf{u}(t)}{dt} + \mathbf{D} \frac{d\mathbf{n}(t)}{dt} \end{aligned} \quad (2.65)$$

Second-order derivative yields:

$$\frac{d^2\mathbf{y}(t)}{dt^2} = \mathbf{C}\mathbf{A} \frac{d\mathbf{x}(t)}{dt} + \mathbf{C}\mathbf{B} \frac{d\mathbf{u}(t)}{dt} + \mathbf{D} \frac{d^2\mathbf{u}(t)}{dt^2} + \mathbf{D} \frac{d^2\mathbf{n}(t)}{dt^2} \quad (2.66)$$

Then substituting (2.62a) into (2.66) to yields:

$$\begin{aligned}\frac{d^2\mathbf{y}(t)}{dt^2} &= \mathbf{CA}(\mathbf{Ax}(t) + \mathbf{Bu}(t)) + \mathbf{CB}\frac{d\mathbf{u}(t)}{dt} + \mathbf{D}\frac{d^2\mathbf{u}(t)}{dt^2} + \mathbf{D}\frac{d^2\mathbf{n}(t)}{dt^2} \\ \frac{d^2\mathbf{y}(t)}{dt^2} &= \mathbf{CA}^2\mathbf{x}(t) + \mathbf{CABu}(t) + \mathbf{CB}\frac{d\mathbf{u}(t)}{dt} + \mathbf{D}\frac{d^2\mathbf{u}(t)}{dt^2} + \mathbf{D}\frac{d^2\mathbf{n}(t)}{dt^2}\end{aligned}\quad (2.67)$$

Third-order derivative yields:

$$\frac{d^3\mathbf{y}(t)}{dt^3} = \mathbf{CA}^2\frac{d\mathbf{x}(t)}{dt} + \mathbf{CAB}\frac{d\mathbf{u}(t)}{dt} + \mathbf{CB}\frac{d^2\mathbf{u}(t)}{dt^2} + \mathbf{D}\frac{d^3\mathbf{u}(t)}{dt^3} + \mathbf{D}\frac{d^3\mathbf{n}(t)}{dt^3}\quad (2.68)$$

Then substituting (2.62a) into (2.68) yields:

$$\begin{aligned}\frac{d^3\mathbf{y}(t)}{dt^3} &= \mathbf{CA}^2(\mathbf{Ax}(t) + \mathbf{Bu}(t)) + \mathbf{CAB}\frac{d\mathbf{u}(t)}{dt} + \mathbf{CB}\frac{d^2\mathbf{u}(t)}{dt^2} + \mathbf{D}\frac{d^3\mathbf{u}(t)}{dt^3} + \mathbf{D}\frac{d^3\mathbf{n}(t)}{dt^3} \\ \frac{d^3\mathbf{y}(t)}{dt^3} &= \mathbf{CA}^3\mathbf{x}(t) + \mathbf{CA}^2\mathbf{Bu}(t) + \mathbf{CAB}\frac{d\mathbf{u}(t)}{dt} + \mathbf{CB}\frac{d^2\mathbf{u}(t)}{dt^2} + \mathbf{D}\frac{d^3\mathbf{u}(t)}{dt^3} + \mathbf{D}\frac{d^3\mathbf{n}(t)}{dt^3}\end{aligned}\quad (2.69)$$

and so on till the  $(i-1)$ -order derivative.

Stacking (2.62b) and all these derivatives into a column vector, the input-output relationship can be written as:

$$\underbrace{\begin{bmatrix} \mathbf{y}(t) \\ \frac{d\mathbf{y}(t)}{dt} \\ \vdots \\ \frac{d^{(i-1)}\mathbf{y}(t)}{dt^{(i-1)}} \end{bmatrix}}_{\mathbf{Y}_{0i-1}(t)} = \underbrace{\begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{(i-1)} \end{bmatrix}}_{\mathbf{\Gamma}_i} \mathbf{x}(t) + \underbrace{\begin{bmatrix} \mathbf{D} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CB} & \mathbf{D} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{(i-2)}\mathbf{B} & \cdots & \mathbf{CB} & \mathbf{D} \end{bmatrix}}_{\mathbf{\Omega}} \underbrace{\begin{bmatrix} \mathbf{u}(t) \\ \frac{d\mathbf{u}(t)}{dt} \\ \vdots \\ \frac{d^{(i-1)}\mathbf{u}(t)}{dt^{(i-1)}} \end{bmatrix}}_{\mathbf{U}_{0i-1}(t)} + \underbrace{\begin{bmatrix} \mathbf{n}(t) \\ \frac{d\mathbf{n}(t)}{dt} \\ \vdots \\ \frac{d^{(i-1)}\mathbf{n}(t)}{dt^{(i-1)}} \end{bmatrix}}_{\mathbf{N}_{0i-1}(t)}\quad (2.70)$$

where  $i$  is a user defined index .

In compact form (2.70) can be written as:

$$\mathbf{Y}_{0i-1}(t) = \mathbf{\Gamma}_i \mathbf{x}(t) + \mathbf{\Omega}_i \mathbf{U}_{0i-1}(t) + \mathbf{N}_{0i-1}(t) \quad (2.71)$$

where  $\mathbf{\Gamma}_i$  and  $\mathbf{\Omega}_i$  are the observability and Toeplitz matrices respectively.

Sampling (2.71) at sampling instances  $t_1, t_2, \dots, t_N$  results in the following input,  $\mathbf{U}_{0i-1,N}$  output,  $\mathbf{Y}_{0i-1,N}$ , noise,  $\mathbf{N}_{0i-1,N}$  and state,  $\mathbf{X}_N$  matrices:

$$\mathbf{U}_{0i-1,N} = \begin{bmatrix} \mathbf{u}(t_1) & \mathbf{u}(t_2) & \cdots & \mathbf{u}(t_N) \\ \frac{d\mathbf{u}(t_1)}{dt} & \frac{d\mathbf{u}(t_2)}{dt} & \cdots & \frac{d\mathbf{u}(t_N)}{dt} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{d^{(i-1)}\mathbf{u}(t_1)}{dt^{(i-1)}} & \frac{d^{(i-1)}\mathbf{u}(t_2)}{dt^{(i-1)}} & \cdots & \frac{d^{(i-1)}\mathbf{u}(t_N)}{dt^{(i-1)}} \end{bmatrix} \quad (2.72)$$

,

$$\mathbf{Y}_{0i-1,N} = \begin{bmatrix} \mathbf{y}(t_1) & \mathbf{y}(t_2) & \cdots & \mathbf{y}(t_N) \\ \frac{d\mathbf{y}(t_1)}{dt} & \frac{d\mathbf{y}(t_2)}{dt} & \cdots & \frac{d\mathbf{y}(t_N)}{dt} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{d^{(i-1)}\mathbf{y}(t_1)}{dt^{(i-1)}} & \frac{d^{(i-1)}\mathbf{y}(t_2)}{dt^{(i-1)}} & \cdots & \frac{d^{(i-1)}\mathbf{y}(t_N)}{dt^{(i-1)}} \end{bmatrix} \quad (2.73)$$

,

$$\mathbf{N}_{0i-1,N} = \begin{bmatrix} \mathbf{n}(t_1) & \mathbf{n}(t_2) & \cdots & \mathbf{n}(t_N) \\ \frac{d\mathbf{n}(t_1)}{dt} & \frac{d\mathbf{n}(t_2)}{dt} & \cdots & \frac{d\mathbf{n}(t_N)}{dt} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{d^{(i-1)}\mathbf{n}(t_1)}{dt^{(i-1)}} & \frac{d^{(i-1)}\mathbf{n}(t_2)}{dt^{(i-1)}} & \cdots & \frac{d^{(i-1)}\mathbf{n}(t_N)}{dt^{(i-1)}} \end{bmatrix} \quad (2.74)$$

and

$$\mathbf{X}_N = [\mathbf{x}(t_1) \quad \mathbf{x}(t_2) \quad \cdots \quad \mathbf{x}(t_N)] \quad (2.75)$$

respectively, where  $N$  is the number of samples.

Using (2.72), (2.73), (2.74) and (2.75), equation (2.71) can then be transformed into the following equation [71]:

$$\mathbf{Y}_{0|i-1,N} = \mathbf{\Gamma}_i \mathbf{X}_N + \mathbf{\Omega}_i \mathbf{U}_{0|i-1,N}(t) + \mathbf{N}_{0|i-1,N} \quad (2.76)$$

Equation (2.76) contains time-derivatives of the input data, output data and noise which in most practical cases are not measured. To address this problem the Poisson moment functional (PMF) approach is applied, details of which are presented in Appendix A2.

Applying PMF to (2.76), the PMF filtered input-output data and noise can be expressed as:

$$\mathbf{P}_t^r [\mathbf{Y}_{0|i-1,N}] = \mathbf{\Gamma}_i \mathbf{P}_t^r [\mathbf{X}_N] + \mathbf{\Omega}_i \mathbf{P}_t^r [\mathbf{U}_{0|i-1,N}] + \mathbf{P}_t^r [\mathbf{N}_{0|i-1,N}] \quad (2.77)$$

The idea is to estimate the observability matrix given the input-output data only, but equation (2.77) also has a noise term,  $\mathbf{P}_t^r [\mathbf{N}_{0|i-1,N}]$  which affects the geometrical properties of the MOESP algorithm [15]], for that reason there is need to reduce the effects of this noise term on the overall filtered data. To address this problem, the authors in [72] proposed correlating the PMF filtered output data,  $\mathbf{P}_t^r [\mathbf{Y}_{0|i-1,N}]$  with instrumental variables (IV). For this to work, the noise variables in the output must be uncorrelated with those of the instrumental variables and on the other hand,  $\mathbf{P}_t^r [\mathbf{Y}_{0|i-1,N}]$  must remain undistorted. This can be achieved by partitioning the input-output data and noise into two parts, namely the past and the future parts. The future output part can then be used as an observation vector, whilst the past input and output parts can be used as instrumental variables. The partitions can be defined as [72]:

$$\mathbf{P}_t^r [\mathbf{U}_{0|i-1,N}] = \begin{bmatrix} \mathbf{P}_t^r [\mathbf{U}_p] \\ \mathbf{P}_t^r [\mathbf{U}_f] \end{bmatrix} \quad (2.78)$$

$$\mathbf{P}_t^r [\mathbf{Y}_{0|i-1,N}] = \begin{bmatrix} \mathbf{P}_t^r [\mathbf{Y}_p] \\ \mathbf{P}_t^r [\mathbf{Y}_f] \end{bmatrix} \quad (2.79)$$

$$\mathbf{P}_t^r [\mathbf{N}_{0|i-1,N}] = \begin{bmatrix} \mathbf{P}_t^r [\mathbf{N}_p] \\ \mathbf{P}_t^r [\mathbf{N}_f] \end{bmatrix} \quad (2.80)$$

where  $\mathbf{P}_t^r [\mathbf{U}_p]$ ,  $\mathbf{P}_t^r [\mathbf{Y}_p]$  and  $\mathbf{P}_t^r [\mathbf{N}_p]$  denote the past input, output and noise respectively, and  $\mathbf{P}_t^r [\mathbf{U}_f]$ ,  $\mathbf{P}_t^r [\mathbf{Y}_f]$  and  $\mathbf{P}_t^r [\mathbf{N}_f]$  denote the future input, output and noise respectively. The past data is defined for index 0 to  $(i-1)$ , whilst the future data is defined for index  $i$  to  $(2i-1)$ .

An instrumental variables matrix,  $\mathbf{V}$  can be constructed using the past input and output parts as:

$$\mathbf{V} = \begin{bmatrix} \mathbf{P}_t^r [\mathbf{U}_p] \\ \mathbf{P}_t^r [\mathbf{Y}_p] \end{bmatrix} \quad (2.81)$$

Considering only the future parts, (2.77) can be rewritten as:

$$\mathbf{P}_t^r [\mathbf{Y}_f] = \mathbf{\Gamma}_i \mathbf{P}_t^r [\mathbf{X}_N] + \mathbf{\Omega}_i \mathbf{P}_t^r [\mathbf{U}_f] + \mathbf{P}_t^r [\mathbf{N}_f] \quad (2.82)$$

Equation (2.82) is then multiplied by (2.81) to help reduce the effects of the noise term on  $\mathbf{P}_t^r [\mathbf{Y}_f]$ , yielding:

$$\mathbf{P}_t^r [\mathbf{Y}_f] \mathbf{V} = \mathbf{\Gamma}_i \mathbf{P}_t^r [\mathbf{X}_N] \mathbf{V} + \mathbf{\Omega}_i \mathbf{P}_t^r [\mathbf{U}_f] \mathbf{V} + \mathbf{P}_t^r [\mathbf{N}_f] \mathbf{V} \quad (2.83)$$

Since the instrumental variables matrix,  $\mathbf{V}$  is independent of the noise term, the noise term in (2.82) disappears as a result of  $\mathbf{P}_t^r [\mathbf{N}_f] \mathbf{V} = \mathbf{0}$ , resulting in:

$$\mathbf{P}_t^r [\mathbf{Y}_f] \mathbf{V} = \Gamma_i \mathbf{P}_t^r [\mathbf{X}_N] \mathbf{V} + \mathbf{\Omega}_t \mathbf{P}_t^r [\mathbf{U}_f] \mathbf{V} \quad (2.84)$$

Having addressed the issue with the noise term, the observability matrix can now be obtained using LQ decomposition and SVD. In (2.62b), given that additive white Gaussian noise is considered, the instrumental variables are chosen to follow the past output multiple-input multiple-output output-error state space (PO-MOESP) algorithm when computing the LQ decomposition and SVD of the filtered input-output data as follows [73]:

Applying the LQ decomposition to matrix (2.84) yields [74]:

$$\begin{bmatrix} \mathbf{P}_t^r [\mathbf{U}_f] \\ \mathbf{P}_t^r [\mathbf{U}_p] \\ \mathbf{P}_t^r [\mathbf{Y}_p] \\ \mathbf{P}_t^r [\mathbf{Y}_f] \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{11} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{L}_{21} & \mathbf{L}_{22} & \mathbf{0} & \mathbf{0} \\ \mathbf{L}_{31} & \mathbf{L}_{32} & \mathbf{L}_{33} & \mathbf{0} \\ \mathbf{L}_{41} & \mathbf{L}_{42} & \mathbf{L}_{43} & \mathbf{L}_{44} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \\ \mathbf{Q}_3 \\ \mathbf{Q}_4 \end{bmatrix} \quad (2.85)$$

where  $\mathbf{L}_{11}, \mathbf{L}_{22} \in \mathbb{R}^{im \times im}$ ,  $\mathbf{L}_{33}, \mathbf{L}_{44} \in \mathbb{R}^{ip \times ip}$  are lower triangular matrices, and  $\mathbf{Q}_1, \mathbf{Q}_2 \in \mathbb{R}^{N \times im}$ ,  $\mathbf{Q}_3, \mathbf{Q}_4 \in \mathbb{R}^{N \times ip}$  are orthogonal matrices.

Using (2.85) the SVD of  $[\mathbf{L}_{42} \quad \mathbf{L}_{43}]$  is computed as:

$$[\mathbf{L}_{42} \quad \mathbf{L}_{43}] = [\mathbf{U}_1 \quad \mathbf{U}_2] \begin{bmatrix} \mathbf{S}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^H \\ \mathbf{V}_2^H \end{bmatrix} \quad (2.86)$$

where  $S_1$  is a diagonal matrix containing the singular values different from zero and the order of the system,  $n$  can be estimated from  $S_1$  as the number of singular values different from zero.

The extended observability matrix,  $\Gamma_i$  can then be estimated from (2.86) as:

$$\hat{\Gamma}_i = U_1 S_1^{\frac{1}{2}} \quad (2.22)$$

The procedure for estimating matrices  $A$ ,  $B$ ,  $C$  and  $D$  is as discussed for the MOESP algorithm.

Having presented a discussion on the integer-order MOESP and integer-order PO-MOESP algorithms for the massive MIMO system, the next step is to study the performance of these algorithms. The following section presents the MATLAB simulation results and analysis of these algorithms having been applied to the massive MIMO system.

## 2.6 Simulation Results

The simulations were run in MATLAB having considered the input signal,  $\mathbf{u}(t)$  to be a chirp signal with frequency ranging from 20Hz to 20 kHz with a sampling frequency twice the highest frequency range, which is a persistently excitation signal. This frequency range was chosen to cover the range of audible frequencies for humans. The chirp excitation signal is shown in Figure 2.7.

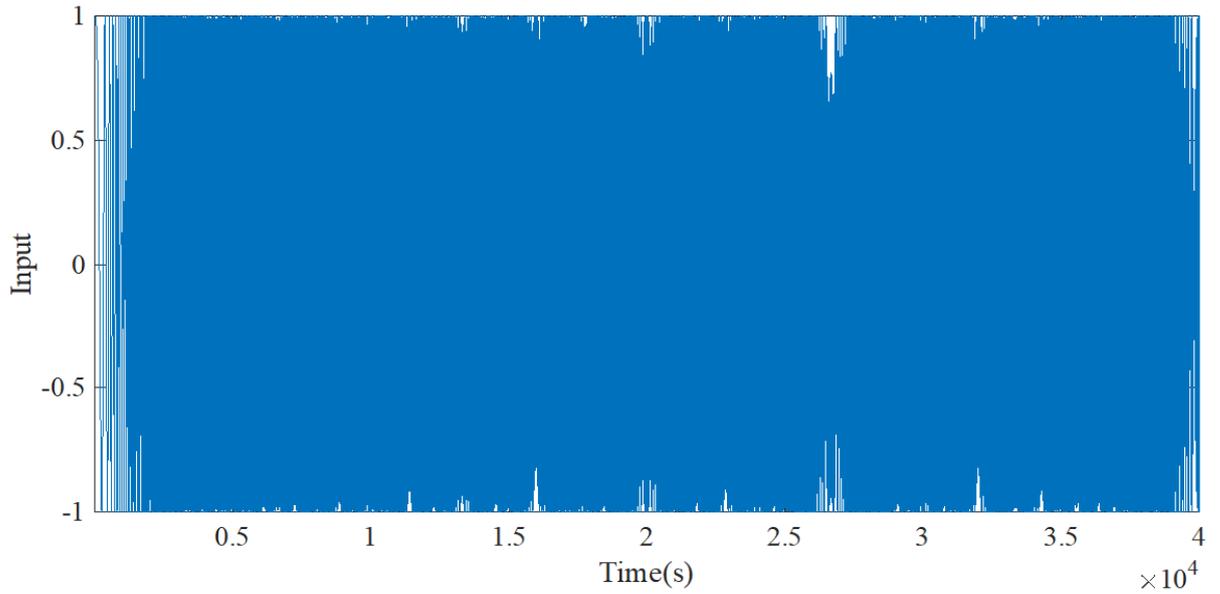
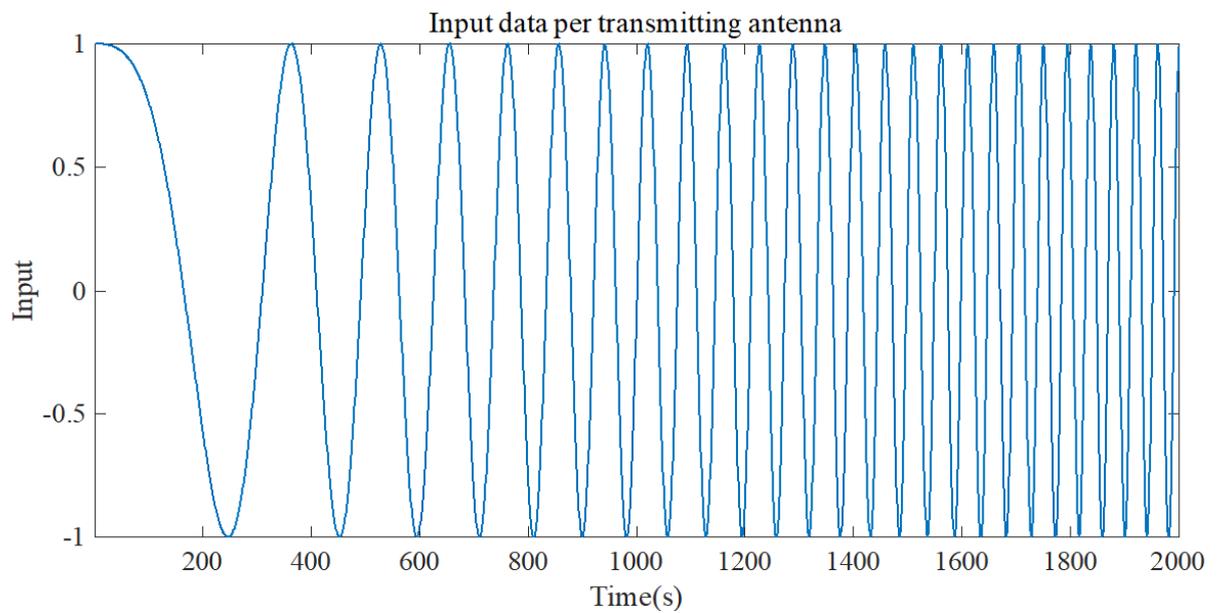


Figure 2. 7: Chirp signal for the input

Through our simulations we noted that not all these data points are needed for system identification. As a result, we reduced the number of data points needed to identify the massive MIMO system, and Figure 2.8 shows the reduced length input signal on each transmitting antenna and the output signal at the receiving antenna we used for system identification.



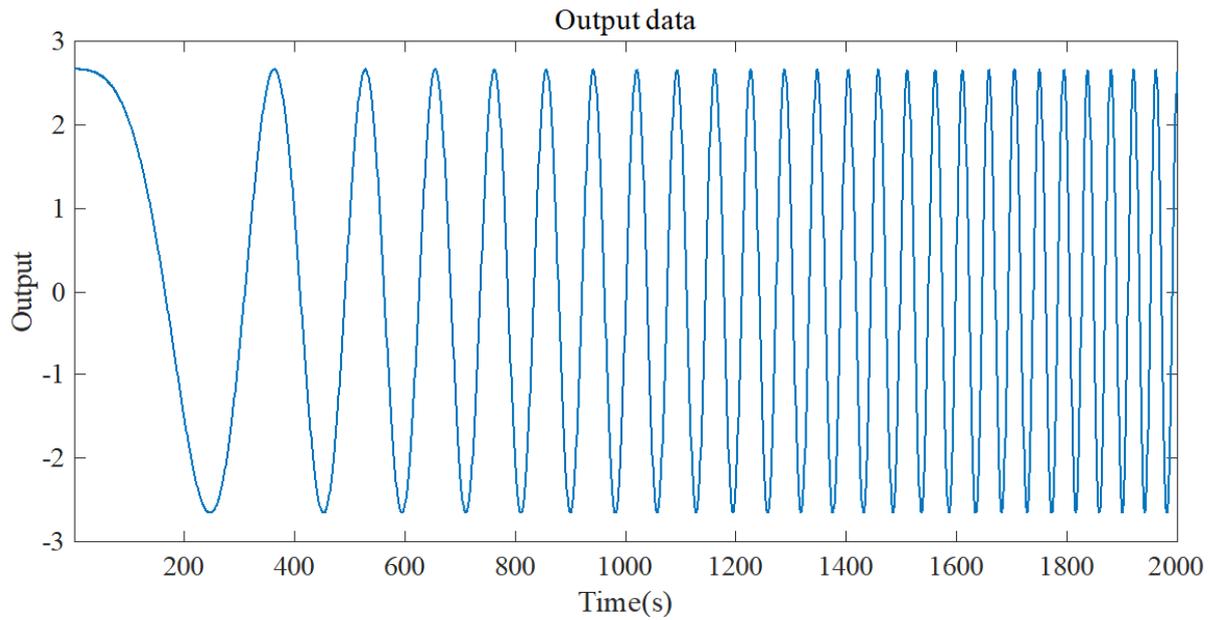


Figure 2. 8: Plot of input and output data

The continuous-time integer-order system identification for the massive MIMO system was first implemented using the in-built MATLAB commands found in MATLAB's System Identification Toolbox to help determine the system order that is able to reliably capture the dynamics of the massive MIMO system. The results of which are shown in Figure 2.9.

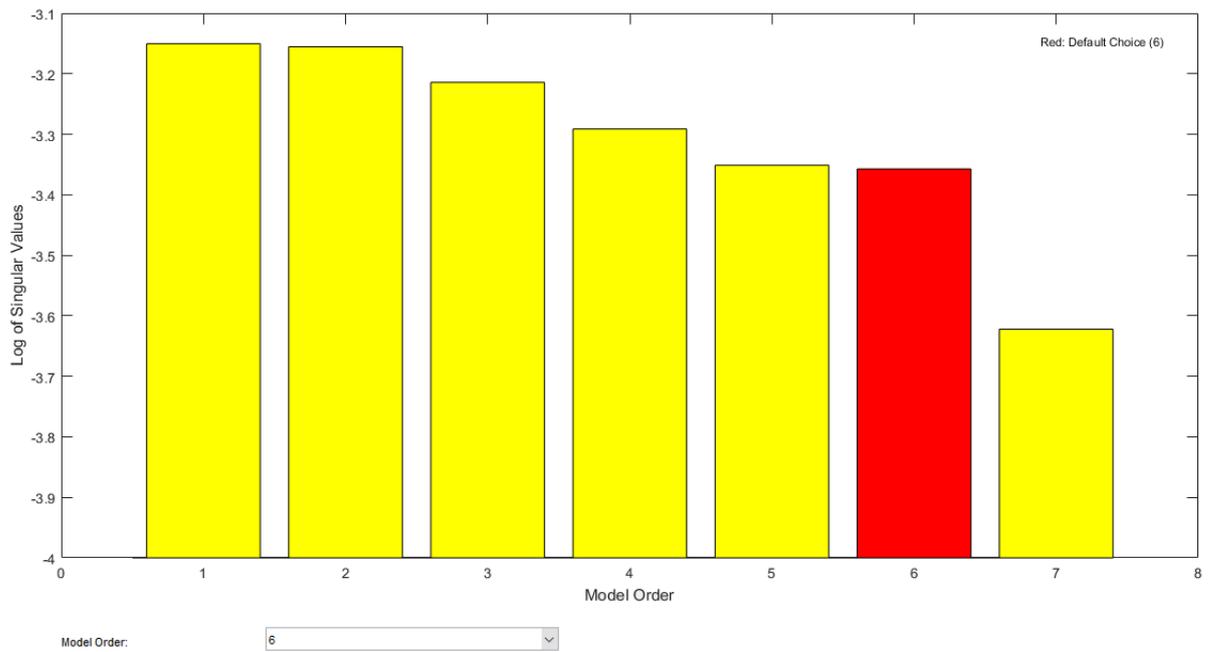


Figure 2. 9: System order obtained using the MATLAB’s System Identification Toolbox

A system of order,  $n=6$  was identified as one that was able to capture the system dynamics reliably. Having obtained the order of the system using MATLAB’s in-built System Identification Toolbox, the simulations were then extended to analytical modelling where simulations based on the integer-order MOESP system identification algorithm discussed in this chapter were run in MATLAB. The data set was divided into estimation data and validation data, where the estimation data was the first half of the data set and the second half was the validation data. The estimation data was used to estimate the system model and the estimated model was verified using the validation data. The integer-order MOESP system identification algorithm was based on the parameters in Table 2.1.

Table 2. 1: System model and system identification parameters for integer-order model

Symbol	Description	Value
$T_x$	Number of transmitting antennas	100
$R_x$	Number of receiving antennas	1
$n$	Model order	1 to 7
$\Delta t$	Sampling time	0.1s
$i$	User defined index	8
$N$	Number of samples	2001
$N_{est}$	Estimation data	1000
$N_{val}$	Validation data	1000
$SNR_{dB}$	Signal-to-noise ratio	20
$L$	Number of channel paths	3
$r$	Poisson filter order	8
$\beta$	Poisson filter gain	1
$\lambda$	Poisson filter constant	1

### 2.6.1 Selection of Identification Parameters

In system identification the performance of the model depends on how much data is available to describe the system and whether the sampling time is good enough to capture the dynamics of the system. In our simulations, the sampling time was varied between different sampling times and sampling time,  $\Delta t = 0.1s$  was identified as the sampling time that could reliably capture the system dynamics. The user defined index,  $i$  was selected to be eight based on the fact that  $i > n$  and with reference to system identification using the in-built MATLAB's System Identification Toolbox,  $n = 6$  was the system order that was able to capture the system dynamics, so a user defined index,  $i$  greater than six had to be selected. The selection of the PMF filter order,  $r$  was based on the criteria defined in (A2.11) and the selection of the Poisson filter gain,  $\beta$  and Poisson filter constant,  $\lambda$  was based on the criteria defined in (A2.12), where for simplicity we assumed that,  $\lambda = \beta = 1$ .

With analytical modelling, the system of order  $n=7$  was observed as the one that reliably captures the dynamics of the massive MIMO system. The comparison of the estimation and validation data to the estimated output which was obtained using the integer-order MOESP algorithm with order  $n=7$  is shown in Figure 2.10.

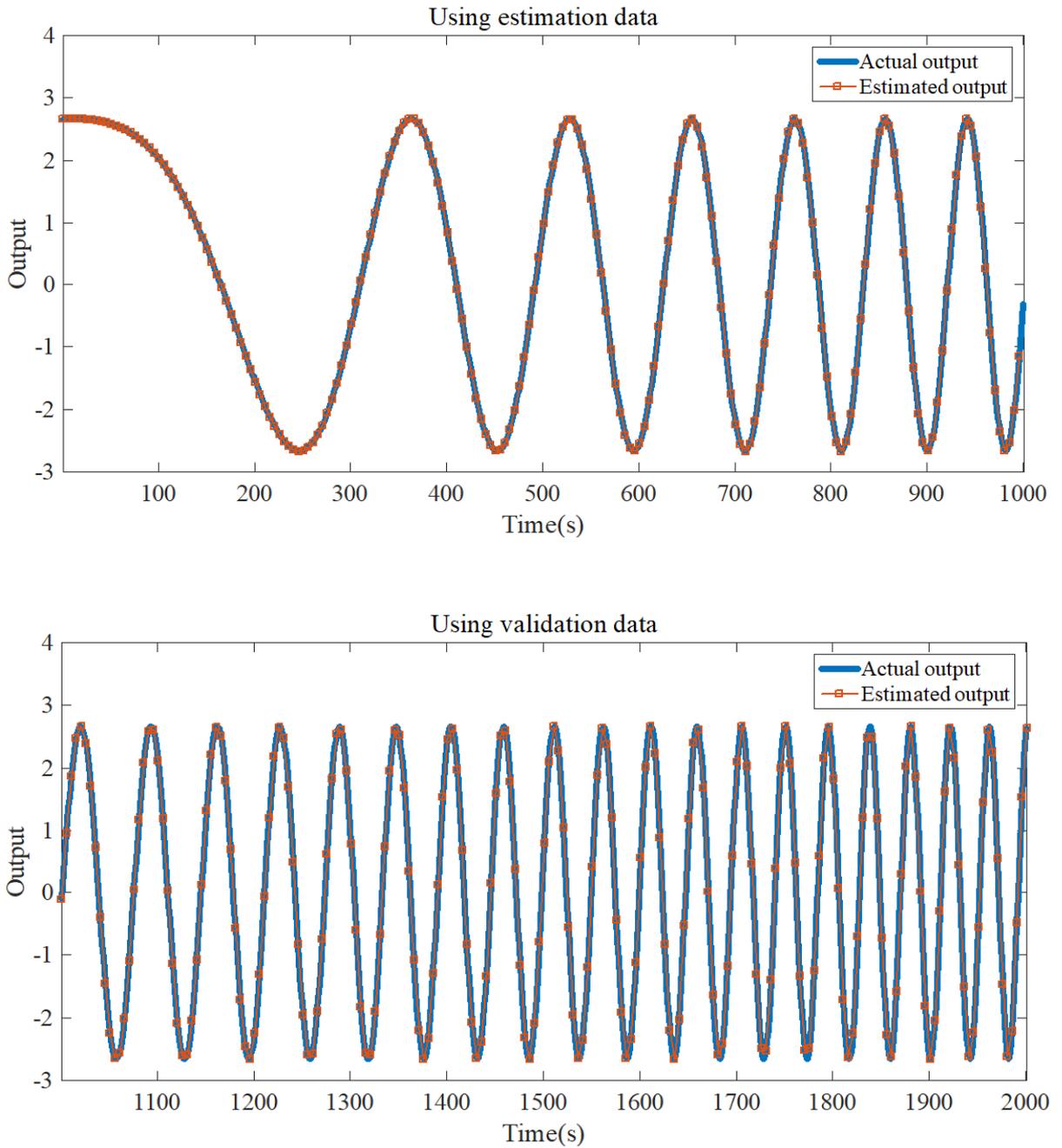


Figure 2. 10: Comparison of estimation and validation data sets for (system order,  $n=7$ ) with the actual system output

We then zoomed in to show a clear picture of our results and Figure 2.11 shows the zoomed in results comparing the integer-order MOESP algorithm estimated output for system order  $n = 7$  and the actual system output.

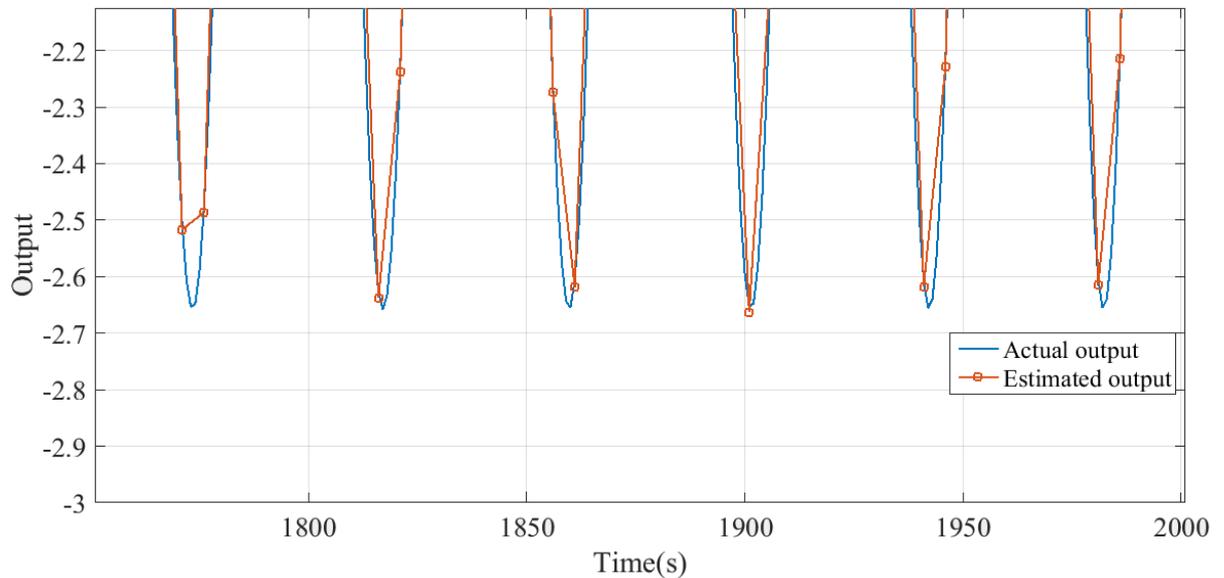


Figure 2. 11: Comparing the performance of the system output obtained using the integer-order MOESP algorithm for  $n = 7$  and the actual system output (zoomed in results)

For completeness, Figure 2.12. shows the performance results of the integer-order MOESP system identification algorithm for different system orders, i.e.  $n = 3$ . Figures for the other system orders, i.e.  $n = 1$ ,  $n = 2$ ,  $n = 4$ ,  $n = 5$  and  $n = 6$  are not shown because for these system orders the system was unstable,. This is why in the analytical modelling we had to extend the system order to  $n = 7$  instead of just  $n = 6$  which was the reliable system order when the in-built MATLAB's System Identification Toolbox commands were used.

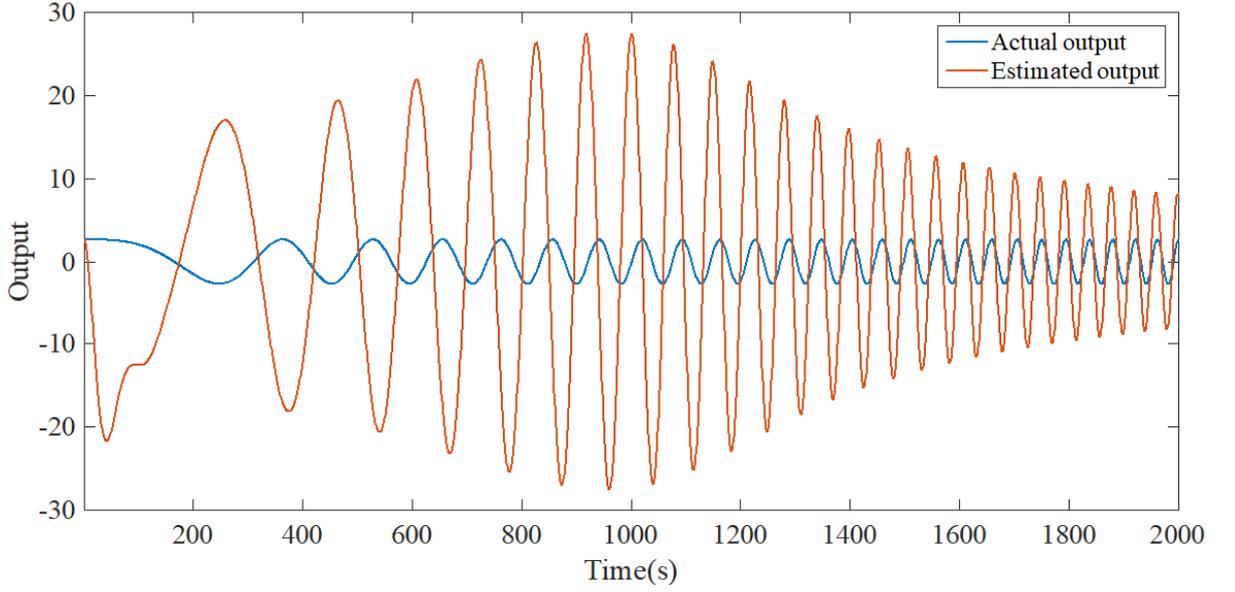


Figure 2. 12: Comparing the performance of the system output obtained using the integer-order MOESP algorithm for  $n = 3$  and the actual system output

All the presented results show that the performance of the integer-order MOESP system identification algorithm improved as the system order was increased from three to seven, order seven being the best performing system order. This shows that with the right selection of the modelling parameters, the proposed continuous-time integer-order MOESP algorithm can actually be used to identify the dynamics of the massive MIMO system.

The estimated outputs for the different system orders were then compared with the measured output/ actual output, and the measure of accuracy of the proposed model was expressed using the mean-square error (MSE) as:

$$MSE = \frac{1}{N} \sum_{t=1}^N |y(t) - \hat{y}(t)|^2 \quad (2.74)$$

where  $N$  is the number of samples and  $\hat{y}(t)$  is the MOESP estimated output.

The MSE test for system orders  $n = 3$  and  $n = 7$  are shown in Table 2.2. Only results for system orders  $n = 3$  and  $n = 7$  are shown because for system orders  $n = 1$ ,  $n = 2$ ,  $n = 4$ ,  $n = 5$  and  $n = 6$  the system was unstable due to the fact that the eigenvalues did

not meet the stability criteria for integer-order systems which states that for a system to be stable the eigenvalues should be negative or should have negative real parts if they are complex conjugate.

Table 2. 2: MSE for system orders  $n = 3$  and  $n = 7$

Model order, $n$	MSE
3	$1.9715 \times 10^6$
7	0.4237

In Table 2.2 it can be seen that system order  $n = 7$  has a low value of MSE compared to system order  $n = 3$  thus verifying our choice of model order  $n = 7$  as the one that gives better performance results.

The simulation results thus far have ignored the effects of the noise when identifying the ISI channel for the massive MIMO system. The next set of results take into consideration the effects of the measurement noise in system identification and to handle this we used the integer-order PO-MOESP identification algorithm. Figure 2.13 shows the performance results of the integer-order PO-MOESP system identification algorithm for system order,  $n = 7$ .

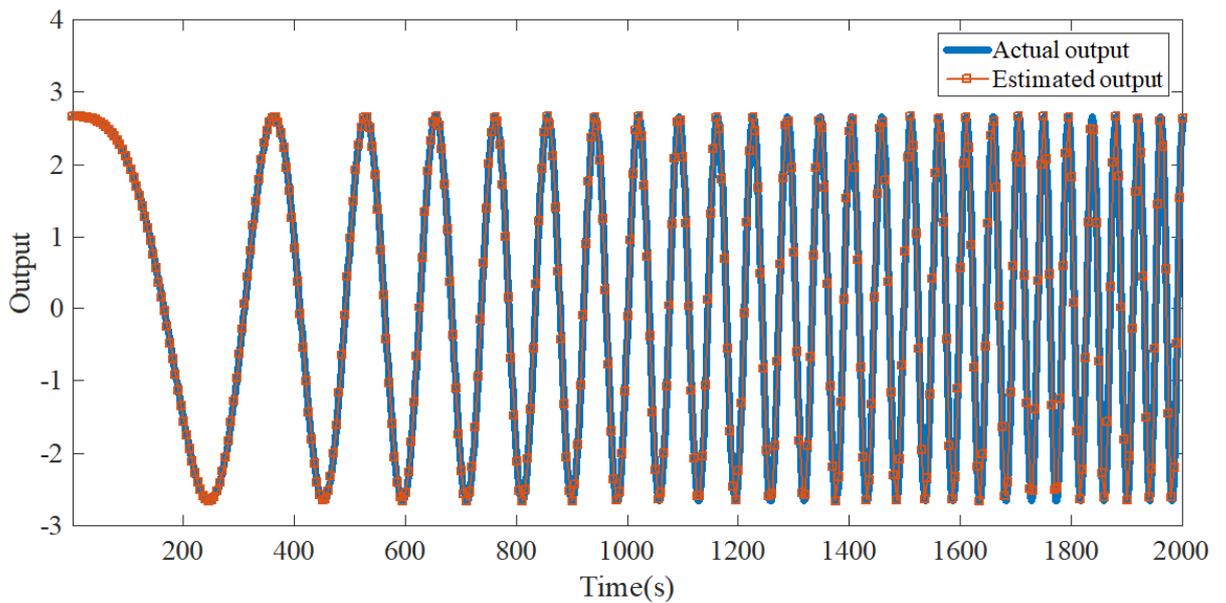


Figure 2. 13: Comparing the performance of the system output obtained using the integer-order PO-MOESP algorithm for  $n = 7$  and the actual system output

We then zoomed in to show a clear picture of our results and Figure 2.14 shows the zoomed in results comparing the integer-order PO-MOESP algorithm estimated output for system order  $n = 7$  and the actual system output.

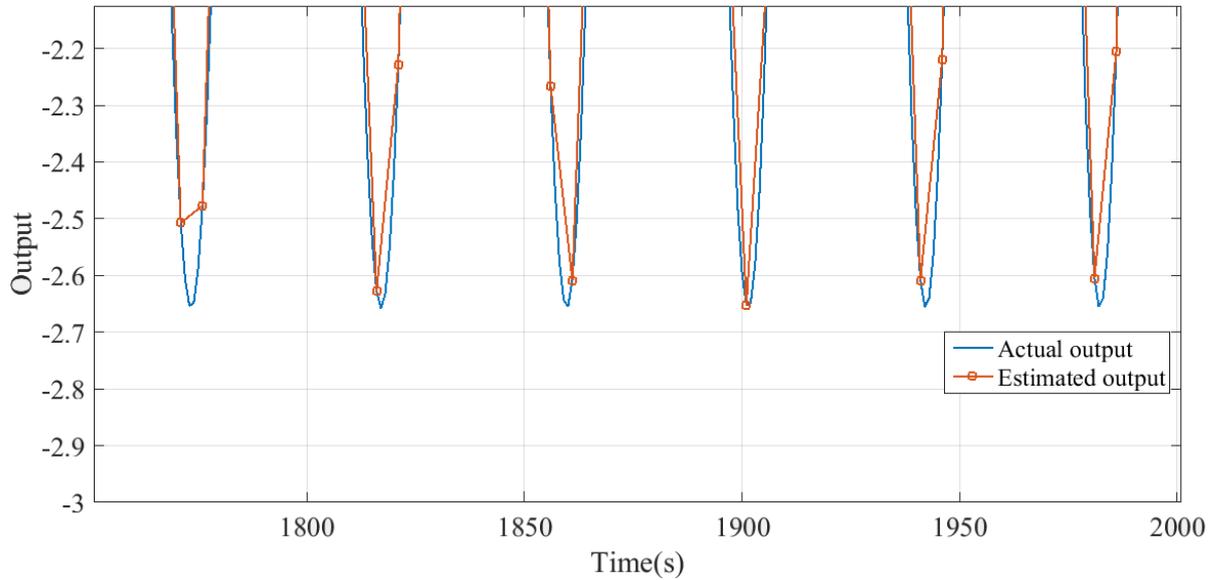


Figure 2. 14: Comparing the performance of the system output obtained using the integer-order PO-MOESP algorithm for  $n = 7$  and the actual system output (zoomed in results)

Figure 2.15 shows the zoomed in performance results of the integer-order MOESP system identification algorithm, integer-order PO-MOESP system identification algorithm for system order,  $n = 7$  and the actual system output.

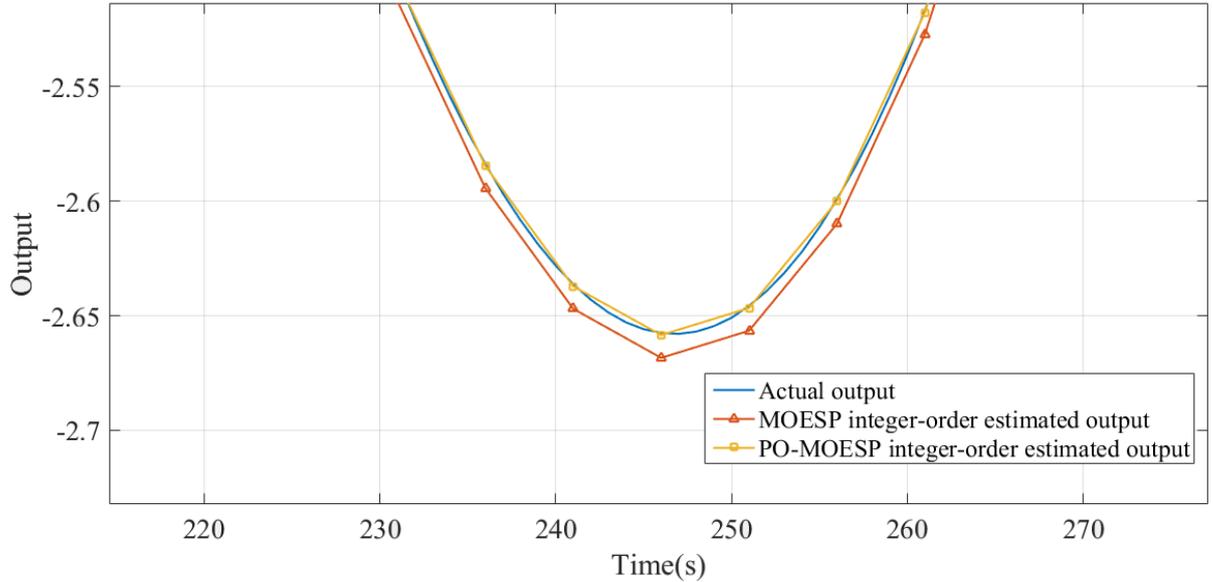


Figure 2. 15: Comparing the performance of the system output obtained using the integer-order MOESP algorithm, integer-order PO-MOESP algorithm for  $n = 7$  and the actual system output (zoomed in results)

The output of the integer-order PO-MOESP identified system was then compared with the measured output/ actual output in terms of the MSE, giving a MSE of 0.0046. These results show an improvement in system identification with the use of the PO-MOESP algorithm. This is because of the introduction of the instrumental variables which help reduce the effects of the noise in the system.

## 2.7 Summary

This chapter presented a continuous-time integer-order state-space identification method, namely the MOESP algorithm to identify the massive MIMO frequency-selective wireless channels. Unlike in discrete-time system identification, the time-derivatives of the input-output data is generally not measured meaning that the input-output matrices are not known. As a result, the classical subspace methods originally developed for the identification of discrete-time models cannot be directly adapted for the identification of continuous-time models. To address this problem the Poisson moment functional (PMF) approach was used when dealing with continuous-time system identification. The input-output data was filtered using the PMF filter after which the MOESP identification algorithm outlined for discrete-time system

identification was then applied to the PMF filtered input-output data. Simulations were run in MATLAB to identify the massive MIMO system using the proposed algorithm. The results showed an improvement in the performance of the continuous-time integer-order MOESP system identification algorithm as the system order was increased from three to seven, order seven being a better performing system order. This shows that the integer-order MOESP algorithm can be used to identify the dynamics of the massive MIMO system. To try to reduce the effects of noise in the identification process, the integer-order PO-MOESP algorithm was considered. This introduced instrumental variables which play a great role in reducing the effects of noise that occur in the system. In the following chapter we present the fractional-order MOESP algorithm. The application of fractional-order system identification in wireless communications is a novelty and is one of the main contributions of this thesis.

## Appendix A2

### A2.1 The Poisson Moment Functional Filtering for Continuous-Time Integer-Order System Identification

To handle the estimation of  $U_{0|i-1,N}$  and  $Y_{0|i-1,N}$  the PMF transform of order  $r$  is applied to (2.58), resulting in:

$$\mathbf{P}_t^r [Y_{0|i-1,N}] = \Gamma_i \mathbf{P}_t^r [X_N] + \Omega_i \mathbf{P}_t^r [U_{0|i-1,N}] \quad (\text{A2.1})$$

where  $\mathbf{P}_t^r [g(t)]$  is the  $r^{\text{th}}$  order PMF transform of signal  $g(t)$  at time instant  $t$  and is given by the following convolution product [70]:

$$\mathbf{P}_t^r [g(t)] = g(t) * p_r(t) = \int_0^t g(\tau) p_r(t-\tau) d\tau \quad (\text{A2.2})$$

where  $p_r(t)$  is the  $r^{\text{th}}$  order Poisson pulse function expressed as:

$$p_r(t) = \frac{\beta^{r+1} t^r e^{-\lambda t}}{r!} \quad (\text{A2.3})$$

where  $r \in \mathbb{N}$ ,  $r \geq i$ , whereby  $i$  is the user-defined index and  $\lambda \in \mathbb{R}^+$  and  $\beta \in \mathbb{R}^+$  are the Poisson filter constant and gain respectively.

The  $r^{\text{th}}$  order PMF of  $g(t)$  can be measured as the output at time instant  $t$  of a cascaded low-pass filter chain of  $(r+1)$  identical stages, each with a transfer function expressed as [70]:

$$P_r(s) = \left( \frac{\beta}{s + \lambda} \right) \quad (\text{A2.4})$$

According to [62] the PMF transform applied to the successive time derivatives of  $g(t)$  is a linear combination of the Poisson filter chain outputs of the signal  $g(t)$ . Considering signal  $g(t)$  with highest order derivative of order  $i$ , where  $i$  is a user defined index and the PMF transform of order  $r$ , the state-space model based transfer function of the  $r^{\text{th}}$  order PMF based filter,  $P_r$  for signal  $g(t)$  can be expressed as [62]:

$$P_r(s) = \begin{bmatrix} \frac{\beta^{(r+1)}}{(s + \lambda)^{(r+1)}} \\ \frac{\beta^{(r+1)}s^1}{(s + \lambda)^{(r+1)}} \\ \vdots \\ \frac{\beta^{(r+1)}s^i}{(s + \lambda)^{(r+1)}} \end{bmatrix} \quad (\text{A2.5})$$

The transfer function in (A2.5) play a great role in transforming the time derivatives of  $g(t)$  using the PMF based approach. For example, given the  $i^{\text{th}}$  derivative of  $g(t)$  as:

$$\frac{d^i g(t)}{dt^i} \quad (\text{A2.6})$$

Applying the PMF based transfer function, (A2.5) to the Laplace transform of (A2.6) yields:

$$\mathbf{P}_s^r [g(s)] = P_r(s) g(s)$$

$$\mathbf{P}_s^r [g(s)] = \begin{bmatrix} \frac{\beta^{(r+1)}}{(s+\lambda)^{(r+1)}} \\ \frac{\beta^{(r+1)} s^1}{(s+\lambda)^{(r+1)}} \\ \vdots \\ \frac{\beta^{(r+1)} s^i}{(s+\lambda)^{(r+1)}} \end{bmatrix} g(s) \quad (\text{A2.7})$$

It can be observed that (A2.7) is free of any time-derivatives.

Applying the property of (A2.7) to the  $(i-1)$  successive time derivatives of the input data, the following PMF based filtered input data is obtained:

$$\mathbf{P}_s^r [\mathbf{U}_{0|i-1,N}] = \begin{bmatrix} \frac{\beta^{(r+1)}}{(s+\lambda)^{(r+1)}} \\ \frac{\beta^{(r+1)} s^1}{(s+\lambda)^{(r+1)}} \\ \vdots \\ \frac{\beta^{(r+1)} s^{(i-1)}}{(s+\lambda)^{(r+1)}} \end{bmatrix} \mathbf{u}(s) \quad (\text{A2.8})$$

where  $g(s)$  in (A2.7) is replaced by  $\mathbf{u}(s)$ .

Applying the property of (A2.7) to the  $(i-1)$  successive time derivatives of the output data and the noise, the following PMF based filtered output data and noise is obtained:

$$\mathbf{P}_s^r [\mathbf{Y}_{0i-1,N}] = \begin{bmatrix} \frac{\beta^{(r+1)}}{(s+\lambda)^{(r+1)}} \\ \frac{\beta^{(r+1)}s^1}{(s+\lambda)^{(r+1)}} \\ \vdots \\ \frac{\beta^{(r+1)}s^{(i-1)}}{(s+\lambda)^{(r+1)}} \end{bmatrix} \mathbf{y}(s) \quad (\text{A2.9})$$

and

$$\mathbf{P}_s^r [\mathbf{N}_{0i-1,N}] = \begin{bmatrix} \frac{\beta^{(r+1)}}{(s+\lambda)^{(r+1)}} \\ \frac{\beta^{(r+1)}s^1}{(s+\lambda)^{(r+1)}} \\ \vdots \\ \frac{\beta^{(r+1)}s^{(i-1)}}{(s+\lambda)^{(r+1)}} \end{bmatrix} \mathbf{n}(s) \quad (\text{A2.10})$$

respectively.

According to [62] the PMF filter order and the PMF filter parameters are chosen as follows:

The PMF filter order,  $r$  has to respect the following condition:

$$r \geq i \quad (\text{A2.11})$$

and for the PMF filter parameters, it is generally assumed that:

$$\lambda = \beta \quad (\text{A2.12})$$

This assumption helps to reduce the number of design parameters.

## Chapter 3

# Massive MIMO Continuous-Time System Identification Using Fractional-Order Algorithm

### 3.1 Introduction

The history of fractional-order calculus dates back to the 17<sup>th</sup> century. It came about through the communication between L'Hospital and Leibniz, where L'Hospital asked Leibniz, what if the order of a derivative was a not an integer? This then led to the founding of the field of fractional-order differentiation and integration, i.e. differ-integration which they refer to as fractional-order calculus in [75] and [76]. The main contribution of this chapter is the application of fractional-order system identification in wireless communications which is a novelty. It discusses on the identification of the massive MIMO frequency-selective wireless channels using the continuous-time fractional-order algorithm. Fractional-order calculus can be thought of as a direct extension to integer-order calculus. Fractional-order calculus is very important, especially in explaining many events which traditional mathematics cannot explain [75]. It has also been observed that many real-world physical systems are well characterized by fractional-order differential equations rather than using classical integer-order models [8]. In signal processing, fractional-order operators are used in the design of fractional-order integrators and fractional-order differentiators and for modelling speech signals [77]. The major advantages of fractional-order calculus are that they make it possible to obtain mathematical models that describe the results closer to the experimental measurements, their ability to predict more accurately the dynamics of the system that is being modelled and they make it possible to obtain simplified system models with just a few physically motivated parameters [78] and [79]. A linear time-invariant fractional-order system can be expressed as [76]:

$$a_n D^{\alpha_n} y(t) + a_{n-1} D^{\alpha_{n-1}} y(t) + \cdots + a_0 D^{\alpha_0} y(t) = b_m D^{\beta_m} u(t) + b_{m-1} D^{\beta_{m-1}} u(t) + \cdots + b_0 D^{\beta_0} u(t) \quad (3.1)$$

where  $a_n, b_m \in \mathbb{R}$  and  $\alpha_n, \beta_m \in \mathbb{R}^+$ ,  $D^{\alpha_n}$  and  $D^{\beta_m}$  are the fractional-order derivative operators.

In (3.1) if the orders of the differentiations are integer multiple of a single base, i.e.  $\alpha_k, \beta_k = k\alpha$ , the system is referred to as commensurate fractional-order system. The advantage of working with commensurate order models is that all fractional powers in the model are integer multiple of a single fractional-order and therefore there is need to estimate only one term, i.e.  $\alpha$ . The commensurate fractional-order system can then be expressed as:

$$\sum_{k=0}^n a_k D^{k\alpha} y(t) = \sum_{k=0}^m b_k D^{k\alpha} u(t) \quad (3.2)$$

Using the fact that operator  $D^\alpha$  acting in time-domain can be written as  $s^\alpha$  in s-domain, the transfer function of (3.2) can be written as:

$$G(s) = \frac{\sum_{k=0}^m b_k (s^\alpha)^k}{\sum_{k=0}^n a_k (s^\alpha)^k} \quad (3.3)$$

The fractional-order time-derivatives of the input-output data are generally not measured meaning that the input-output matrices are not known, as a result the classical subspace methods originally developed for the identification of discrete-time models cannot be directly adapted for the identification of continuous-time fractional-order models. To address this problem the Poisson moment functional (PMF) approach is used when dealing with continuous-time fractional-order system identification, where the input-output data is first filtered using the PMF filter after which the MOESP identification algorithm outlined for discrete-time system identification is then applied

to the PMF filtered input-output data. Details of applying the PMF to fractional-order system identification are discussed at a later stage in this chapter.

Given the complexity involved in dealing with fractional-order transfer functions as discussed in [80], [81] and [82] the use of different continuous-time approximation methods namely Oustaloup, Carlson and Charefs' methods that can be used to present the fractional-order transfer functions as approximated rational transfer functions has been proposed. These methods will also be applied in approximating the fractional-order transfer function of the massive MIMO frequency-selective wireless channels. The mathematical background of fractional-order calculus, i.e. some important functions, special properties and definitions of fractional-order calculus is presented in Appendix A3

## 3.2 Fractional-Order System Identification

Fractional-order state-space models in controllable, observable and diagonal canonical forms are similar to integer-order state-space models [83]. A fractional-order linear time invariant (LTI) system is mathematically equivalent to an infinite dimensional LTI filter. Thus, a fractional-order system can be approximated using higher order polynomials having integer-order differ-integration operators.

The following assumptions are necessary when dealing with system identification:

- i) The system is persistently excited by the training symbols.
- ii) The system is stable, observable and controllable.
- iii) The dimensions of matrix  $A$  are known, and  $rank(A) = n$ , where  $n$  is the order of the system. Rank is the number of linearly independent rows.
- iv) The random white noise is uncorrelated to the input signal.

The dynamics of the massive multiple-input multiple-output linear time invariant (MIMO LTI) system can be modelled using fractional-order state-space modelling. Ignoring the effects of the process noise and additive noise and using

continuous-time fractional-order state-space modelling, the linear time invariant length- $L$  ISI channel massive MIMO system, (2.1) can be expressed as [84]:

$$D^\alpha \mathbf{x}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad (3.4a)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \quad (3.4b)$$

where  $\mathbf{x}(t)$  is the  $n \times 1$  state vector,  $\mathbf{u}(t)$  is the  $m \times 1$  input vector,  $\mathbf{y}(t)$  is the  $p \times 1$  output vector,  $\mathbf{A}$  is the  $n \times n$  system matrix and it describes the dynamics of the system, i.e. the eigenvalues of the system,  $\mathbf{B}$  is the  $n \times m$  input matrix and it describes the linear transformation by which the inputs influence the next state,  $\mathbf{C}$  is the  $p \times n$  output matrix, and it describes how the state is transferred to the output,  $\mathbf{D}$  is the  $p \times m$  feed-forward matrix,  $\alpha$  is the commensurate fractional-order and  $D^\alpha$  is the fractional derivative of order  $\alpha$ . Equation (3.4a) is called the fractional-order state equation and (3.4b) called the output equation.

Applying the Laplace transform to (3.4) results in:

$$\begin{aligned} s^\alpha \mathbf{X}(s) &= \mathbf{A}\mathbf{X}(s) + \mathbf{B}\mathbf{U}(s) \\ \mathbf{Y}(s) &= \mathbf{C}\mathbf{X}(s) + \mathbf{D}\mathbf{U}(s) \end{aligned} \quad (3.5)$$

Rearranging (3.5) and after some manipulations the transfer function of the system is expressed as:

$$\mathbf{G}(s) = \frac{\mathbf{Y}(s)}{\mathbf{U}(s)} = \mathbf{C} \left( s^\alpha \mathbf{I}_n - \mathbf{A} \right)^{-1} \mathbf{B} + \mathbf{D} \quad (3.6)$$

where  $\mathbf{I}_n$  is an  $n \times n$  identity matrix.

The general characteristics of the transfer function, (3.6) are [85]:

- i) The magnitude curve has a constant slope of  $-20\alpha$  dB/dec.

- ii) The phase plot is a horizontal line of value  $-\alpha \frac{\pi}{2}$ .
- iii) The gain crossover frequency depends on  $A$ .
- iv) The Nyquist plot is a straight line which starts from the origin with argument  $-\alpha \frac{\pi}{2}$ .

It is important to know if a fractional-order system is stable or not and in the fractional-order LTI case, the stability is different from that of the integer-order one. In the fractional-order case a stable fractional-order system may have roots on the right half of the complex plane as shown in Figure 3.7. A fractional-order system is stable if [86] and [87]:

$$|\arg(\text{eig}(A))| > \alpha \frac{\pi}{2} \quad (3.7)$$

$$|\arg(\lambda_k)| > \alpha \frac{\pi}{2}, \forall k = 1, \dots, n \quad (3.8)$$

where  $0 < \alpha < 2$ ,  $\text{eig}(A)$  gives the eigenvalues of matrix  $A$  and  $\lambda_k$  is the  $k$ th eigenvalue of  $A$  and  $-\pi < \arg(\lambda_k) \leq \pi$ .

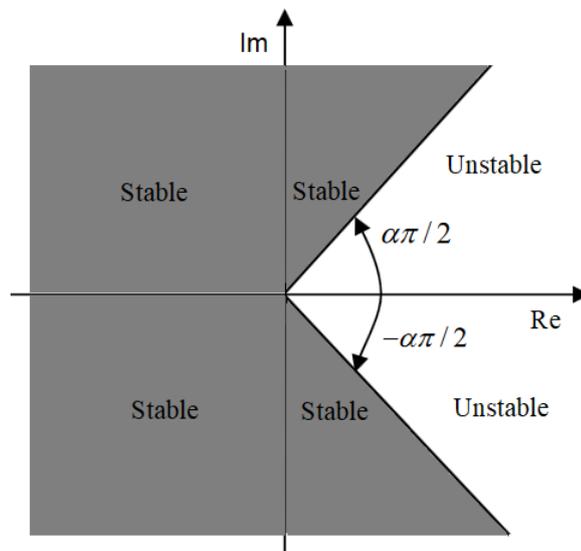


Figure 3. 1: Stability region for fractional-order commensurate systems

Since the continuous-time state-space representation of commensurate fractional-order systems is similar to that of integer-order systems [88], the analysis of the MOESP fractional-order model follows the one outlined for the classical MOESP model as proposed in [65] and [84].

### 3.2.1 Constructing Data Matrices

In constructing the data matrices for the continuous-time fractional-order system, (3.4) and the MOESP algorithm are considered. By computing the successive  $\alpha$ -order fractional derivatives of (3.4) the data equations are constructed as follows [69]:

Taking the  $\alpha$ -order derivative of (3.4b) yields:

$$D^\alpha \mathbf{y}(t) = \mathbf{C}D^\alpha \mathbf{x}(t) + \mathbf{D}D^\alpha \mathbf{u}(t) \quad (3.9)$$

Substituting (3.4a) into (3.9) yields:

$$\begin{aligned} D^\alpha \mathbf{y}(t) &= \mathbf{C}(\mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)) + \mathbf{D}D^\alpha \mathbf{u}(t) \\ D^\alpha \mathbf{y}(t) &= \mathbf{C}\mathbf{A}\mathbf{x}(t) + \mathbf{C}\mathbf{B}\mathbf{u}(t) + \mathbf{D}D^\alpha \mathbf{u}(t) \end{aligned} \quad (3.10)$$

Taking the  $2\alpha$ -order derivative of (3.4b) and making the necessary simplifications yields:

$$D^{2\alpha} \mathbf{y}(t) = \mathbf{C}\mathbf{A}D^\alpha \mathbf{x}(t) + \mathbf{C}\mathbf{B}D^\alpha \mathbf{u}(t) + \mathbf{D}D^{2\alpha} \mathbf{u}(t) \quad (3.11)$$

Substituting (3.4a) into (3.11) yields:

$$\begin{aligned} D^{2\alpha} \mathbf{y}(t) &= \mathbf{C}\mathbf{A}(\mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)) + \mathbf{C}\mathbf{B}D^\alpha \mathbf{u}(t) + \mathbf{D}D^{2\alpha} \mathbf{u}(t) \\ D^{2\alpha} \mathbf{y}(t) &= \mathbf{C}\mathbf{A}^2 \mathbf{x}(t) + \mathbf{C}\mathbf{A}\mathbf{B}\mathbf{u}(t) + \mathbf{C}\mathbf{B}D^\alpha \mathbf{u}(t) + \mathbf{D}D^{2\alpha} \mathbf{u}(t) \end{aligned} \quad (3.12)$$

Taking the  $3\alpha$ -order derivative of (3.4b) and making the necessary simplifications yields:

$$D^{3\alpha} \mathbf{y}(t) = \mathbf{CA}^2 D^\alpha \mathbf{x}(t) + \mathbf{CABD}^\alpha \mathbf{u}(t) + \mathbf{CBD}^{2\alpha} \mathbf{u}(t) + \mathbf{DD}^{3\alpha} \mathbf{u}(t) \quad (3.13)$$

Substituting (3.4a) into (3.13) yields:

$$\begin{aligned} D^{3\alpha} \mathbf{y}(t) &= \mathbf{CA}^2 (\mathbf{Ax}(t) + \mathbf{Bu}(t)) + \mathbf{CABD}^\alpha \mathbf{u}(t) + \mathbf{CBD}^{2\alpha} \mathbf{u}(t) + \mathbf{DD}^{3\alpha} \mathbf{u}(t) \\ D^{3\alpha} \mathbf{y}(t) &= \mathbf{CA}^3 \mathbf{x}(t) + \mathbf{CA}^2 \mathbf{Bu}(t) + \mathbf{CABD}^\alpha \mathbf{u}(t) + \mathbf{CBD}^{2\alpha} \mathbf{u}(t) + \mathbf{DD}^{3\alpha} \mathbf{u}(t) \end{aligned} \quad (3.14)$$

and so on till the  $(i-1)\alpha$ -order derivative.

Stacking (3.4b) and all its  $\alpha$ -order derivatives obtained above up to order  $(i-1)\alpha$  into a column vector, the input-output relationship can be written as:

$$\underbrace{\begin{bmatrix} \mathbf{y}(t) \\ D^\alpha \mathbf{y}(t) \\ \vdots \\ D^{(i-1)\alpha} \mathbf{y}(t) \end{bmatrix}}_{\mathbf{Y}_{0(i-1)\alpha}(t)} = \underbrace{\begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{(i-1)} \end{bmatrix}}_{\mathbf{\Gamma}_i} \mathbf{x}(t) + \underbrace{\begin{bmatrix} \mathbf{D} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CB} & \mathbf{D} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{(i-2)} \mathbf{B} & \cdots & \mathbf{CB} & \mathbf{D} \end{bmatrix}}_{\mathbf{\Omega}} \underbrace{\begin{bmatrix} \mathbf{u}(t) \\ D^\alpha \mathbf{u}(t) \\ \vdots \\ D^{(i-1)\alpha} \mathbf{u}(t) \end{bmatrix}}_{\mathbf{U}_{0(i-1)\alpha}(t)} \quad (3.15)$$

where  $i$  is a user defined index .

In compact form (3.15) can be written as:

$$\mathbf{Y}_{0(i-1)\alpha}(t) = \mathbf{\Gamma}_i \mathbf{x}(t) + \mathbf{\Omega}_i \mathbf{U}_{0(i-1)\alpha}(t) \quad (3.16)$$

$$\text{where } \mathbf{U}_{0(i-1)\alpha}(t) = \begin{bmatrix} \mathbf{u}(t) \\ D^\alpha \mathbf{u}(t) \\ \vdots \\ D^{(i-1)\alpha} \mathbf{u}(t) \end{bmatrix} \in \mathbb{R}^{im}, \mathbf{Y}_{0(i-1)\alpha}(t) = \begin{bmatrix} \mathbf{y}(t) \\ D^\alpha \mathbf{y}(t) \\ \vdots \\ D^{(i-1)\alpha} \mathbf{y}(t) \end{bmatrix} \in \mathbb{R}^{ip},$$

$$\mathbf{\Gamma}_i = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{i-1} \end{bmatrix} \in \mathbb{R}^{ip \times n} \text{ and } \mathbf{\Omega}_i = \begin{bmatrix} \mathbf{D} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CB} & \mathbf{D} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{i-2} \mathbf{B} & \mathbf{CA}^{i-3} \mathbf{B} & \cdots & \mathbf{D} \end{bmatrix} \in \mathbb{R}^{ip \times im}$$

where the notation  $(\bullet)_{0(i-1)\alpha}$  denotes the zero<sup>th</sup> order derivative to the  $(i-1)\alpha$ <sup>th</sup> order derivative,  $\mathbf{\Gamma}_i$  and  $\mathbf{\Omega}_i$  are the observability and Toeplitz matrices respectively.

Sampling (3.16) at sampling instances  $t_1, t_2, \dots, t_N$  results in the following input,

$\mathbf{U}_{0(i-1)\alpha, N}$ , output,  $\mathbf{Y}_{0(i-1)\alpha, N}$  and state  $\mathbf{X}_N$  matrices:

$$\mathbf{U}_{0(i-1)\alpha, N} = \begin{bmatrix} \mathbf{u}(t_1) & \mathbf{u}(t_2) & \cdots & \mathbf{u}(t_N) \\ D^\alpha \mathbf{u}(t_1) & D^\alpha \mathbf{u}(t_2) & \cdots & D^\alpha \mathbf{u}(t_N) \\ \vdots & \vdots & \cdots & \vdots \\ D^{(i-1)\alpha} \mathbf{u}(t_1) & D^{(i-1)\alpha} \mathbf{u}(t_2) & \cdots & D^{(i-1)\alpha} \mathbf{u}(t_N) \end{bmatrix} \in \mathbb{R}^{im \times N} \quad (3.17)$$

,

$$\mathbf{Y}_{0(i-1)\alpha, N} = \begin{bmatrix} \mathbf{y}(t_1) & \mathbf{y}(t_2) & \cdots & \mathbf{y}(t_N) \\ D^\alpha \mathbf{y}(t_1) & D^\alpha \mathbf{y}(t_2) & \cdots & D^\alpha \mathbf{y}(t_N) \\ \vdots & \vdots & \cdots & \vdots \\ D^{(i-1)\alpha} \mathbf{y}(t_1) & D^{(i-1)\alpha} \mathbf{y}(t_2) & \cdots & D^{(i-1)\alpha} \mathbf{y}(t_N) \end{bmatrix} \in \mathbb{R}^{ip \times N} \quad (3.18)$$

and

$$\mathbf{X}_N = [\mathbf{x}(t_1) \quad \mathbf{x}(t_2) \quad \cdots \quad \mathbf{x}(t_N)] \quad (3.19)$$

respectively, where the notation  $(\bullet)_{0(i-1)\alpha}$  denotes the zero<sup>th</sup> order derivative to the  $i\alpha$ <sup>th</sup> order derivative for  $N$  samples.

Using (3.17), (3.18) and (3.19), equation (3.16) can then be transformed into the following equation [71]:

$$\mathbf{Y}_{0(i-1)\alpha,N} = \mathbf{\Gamma}_i \mathbf{X}_N + \mathbf{\Omega}_i \mathbf{U}_{0(i-1)\alpha,N} \quad (3.20)$$

### 3.2.2 Estimating the Extended Observability Matrix

With reference to (3.17) and (3.18) the data matrix,  $\mathbf{W}_{0(i-1)\alpha,N}$  can be constructed as:

$$\mathbf{W}_{0(i-1)\alpha,N} = \begin{bmatrix} \mathbf{U}_{0(i-1)\alpha,N} \\ \mathbf{Y}_{0(i-1)\alpha,N} \end{bmatrix} \quad (3.21)$$

Equation (3.21) contains successive  $\alpha$ -order fractional derivatives of the input-output data which in most practical cases are not measured. To address this problem we propose the use of the Poisson moment functional (PMF) approach, details of which are presented in Appendix A3.

Applying PMF to the data matrix,  $\mathbf{W}_{0(i-1)\alpha,N}$ , the PMF filtered data can be expressed as:

$$\mathbf{P}_t^r \left[ \mathbf{W}_{0(i-1)\alpha,N} \right] = \begin{bmatrix} \mathbf{P}_t^r \left[ \mathbf{U}_{0(i-1)\alpha,N} \right] \\ \mathbf{P}_t^r \left[ \mathbf{Y}_{0(i-1)\alpha,N} \right] \end{bmatrix} \quad (3.22)$$

Having filtered the input-output data the MOESP identification algorithm can then be applied.

Applying LQ decomposition to matrix (3.22) yields:

$$\begin{bmatrix} \mathbf{P}_t^r \left[ \mathbf{U}_{0(i-1)\alpha,N} \right] \\ \mathbf{P}_t^r \left[ \mathbf{Y}_{0(i-1)\alpha,N} \right] \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{11} & \mathbf{0} \\ \mathbf{L}_{21} & \mathbf{L}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \end{bmatrix} \quad (3.23)$$

Using (3.23) compute the SVD of  $L_{22}$  as:

$$L_{22} = [U_1 \quad U_2] \begin{bmatrix} S_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} V_1^H \\ V_2^H \end{bmatrix} \quad (3.24)$$

where  $S_1$  is a diagonal matrix containing the singular values different from zero and the order of the system,  $n$  can be estimated from  $S_1$  as the number of singular values different from zero.

The extended observability matrix,  $\Gamma_i$  can then be estimated from (3.24) as:

$$\hat{\Gamma}_i = U_1 S_1^{\frac{1}{2}} \quad (2.22)$$

The procedure for estimating matrices  $A$ ,  $B$ ,  $C$  and  $D$  is as discussed for the MOESP algorithm. If the fractional-order,  $\alpha \in (0, 2)$  is unknown then it can be estimated by minimizing [89]:

$$\hat{\alpha} = \arg \min_{\alpha \in (0, 2)} \frac{1}{2} \|\mathbf{y}(t) - \hat{\mathbf{y}}_{\alpha}(t)\|_2^2 \quad (3.25)$$

where  $\hat{\mathbf{y}}_{\alpha}(t)$  is the fractional-order estimated output.

### 3.2.3 Analytical Solution of the Fractional-Order Massive MIMO System

Considering (3.4a), and applying the fractional-order integration of order  $\alpha$  on both sides results in:

$$\mathbf{x}(t) = A I^{\alpha} \mathbf{x}(t) + B I^{\alpha} \mathbf{u}(t) + \mathbf{x}(0) \quad (3.26)$$

where  $I^\alpha$  is the fractional-order integral operator and  $\mathbf{x}(0)$  is the initial condition.

Assuming that the general solution of (3.26) can be written as [90]:

$$\mathbf{x}(t) = \sum_{k=0}^{\infty} \mathbf{x}_k(t) \quad (3.27)$$

where

$$\mathbf{x}_0(t) = \mathbf{x}(0) + \mathbf{B}I^\alpha \mathbf{u}(t) \quad (3.28)$$

and

$$\mathbf{x}_k(t) = \mathbf{A}I^\alpha \mathbf{x}_{k-1}(t), \quad k \geq 1 \quad (3.29)$$

where  $\mathbf{x}_{k-1}(t)$  is the previous state vector.

Using (3.28) and (3.29) the state vector can be recursively written as follows:

At  $k=1$ :

$$\mathbf{x}_1(t) = \mathbf{A}I^\alpha \mathbf{x}_0(t) \quad (3.30)$$

Substituting (3.28) into (3.30) yields:

$$\mathbf{x}_1(t) = \mathbf{A}I^\alpha [\mathbf{x}(0) + \mathbf{B}I^\alpha \mathbf{u}(t)] = I^\alpha [\mathbf{A}\mathbf{x}(0) + \mathbf{A}\mathbf{B}I^\alpha \mathbf{u}(t)] \quad (3.31)$$

At  $k=2$ :

$$\mathbf{x}_2(t) = \mathbf{A}I^\alpha \mathbf{x}_1(t) \quad (3.32)$$

Substituting (3.31) into (3.32) yields:

$$\mathbf{x}_2(t) = \mathbf{A}I^\alpha \left[ I^\alpha \left[ \mathbf{A}\mathbf{x}(0) + \mathbf{A}\mathbf{B}I^\alpha \mathbf{u}(t) \right] \right] = I^{2\alpha} \left[ \mathbf{A}^2 \mathbf{x}(0) + \mathbf{A}^2 \mathbf{B}I^\alpha \mathbf{u}(t) \right] \quad (3.33)$$

At  $k=3$ :

$$\mathbf{x}_3(t) = \mathbf{A}I^\alpha \mathbf{x}_2(t) \quad (3.34)$$

Substituting (3.33) into (3.34) yields:

$$\mathbf{x}_3(t) = \mathbf{A}I^\alpha \left[ I^{2\alpha} \left[ \mathbf{A}^2 \mathbf{x}(0) + \mathbf{A}^2 \mathbf{B}I^\alpha \mathbf{u}(t) \right] \right] = I^{3\alpha} \left[ \mathbf{A}^3 \mathbf{x}(0) + \mathbf{A}^3 \mathbf{B}I^\alpha \mathbf{u}(t) \right] \quad (3.35)$$

and so on we evaluate.

With reference to the above equations, the general expression for  $\mathbf{x}_k(t)$  can be written as:

$$\mathbf{x}_k(t) = I^{k\alpha} \left[ \mathbf{A}^k \mathbf{x}(0) + \mathbf{A}^k \mathbf{B}I^\alpha \mathbf{u}(t) \right] \quad (3.36)$$

Using the property [90]:

$$I^\alpha t^\gamma = \frac{\Gamma(\gamma+1)}{\Gamma(\gamma+\alpha+1)} t^{\gamma+\alpha}, \quad \alpha > 0, \gamma > -1, t > 0 \quad (3.37)$$

equations (3.31), (3.33), (3.35), etc can be written as:

$$\begin{aligned} \mathbf{x}_1(t) &= I^\alpha \left[ \mathbf{A}\mathbf{x}(0) + \mathbf{A}\mathbf{B}I^\alpha \mathbf{u}(t) \right] = \mathbf{A}I^\alpha \mathbf{x}(0) + \mathbf{A}\mathbf{B}I^{2\alpha} \mathbf{u}(t) \\ \mathbf{x}_1(t) &= \mathbf{A}\mathbf{x}(0) \frac{\Gamma(1)}{\Gamma(\alpha+1)} t^\alpha + \mathbf{A}\mathbf{B}I^{2\alpha} \mathbf{u}(t) \end{aligned} \quad (3.38)$$

$$\begin{aligned}\mathbf{x}_2(t) &= I^{2\alpha} \left[ \mathbf{A}^2 \mathbf{x}(0) + \mathbf{A}^2 \mathbf{B} I^\alpha \mathbf{u}(t) \right] = \mathbf{A}^2 I^{2\alpha} \mathbf{x}(0) + \mathbf{A}^2 \mathbf{B} I^{3\alpha} \mathbf{u}(t) \\ \mathbf{x}_2(t) &= \mathbf{A}^2 \mathbf{x}(0) \frac{\Gamma(1)}{\Gamma(2\alpha+1)} t^{2\alpha} + \mathbf{A}^2 \mathbf{B} I^{3\alpha} \mathbf{u}(t)\end{aligned}\quad (3.39)$$

$$\begin{aligned}\mathbf{x}_3(t) &= I^{3\alpha} \left[ \mathbf{A}^3 \mathbf{x}(0) + \mathbf{A}^3 \mathbf{B} I^\alpha \mathbf{u}(t) \right] = \mathbf{A}^3 I^{3\alpha} \mathbf{x}(0) + \mathbf{A}^3 \mathbf{B} I^{4\alpha} \mathbf{u}(t) \\ \mathbf{x}_3(t) &= \mathbf{A}^3 \mathbf{x}(0) \frac{\Gamma(1)}{\Gamma(3\alpha+1)} t^{3\alpha} + \mathbf{A}^3 \mathbf{B} I^{4\alpha} \mathbf{u}(t)\end{aligned}\quad (3.40)$$

respectively, where  $\gamma$  is taken to be  $\gamma = 0$  since we do not have the 't' term.

Using the above property, the general expression for  $\mathbf{x}_k(t)$  can be rewritten as:

$$\begin{aligned}\mathbf{x}_k(t) &= I^{k\alpha} \left[ \mathbf{A}^k \mathbf{x}(0) + \mathbf{A}^k \mathbf{B} I^\alpha \mathbf{u}(t) \right] = \mathbf{A}^k I^{k\alpha} \mathbf{x}(0) + \mathbf{A}^k \mathbf{B} I^{\alpha(k+1)} \mathbf{u}(t) \\ \mathbf{x}_k(t) &= \mathbf{A}^k \mathbf{x}(0) \frac{\Gamma(1)}{\Gamma(k\alpha+1)} t^{k\alpha} + \mathbf{A}^k \mathbf{B} I^{\alpha(k+1)} \mathbf{u}(t)\end{aligned}\quad (3.41)$$

Using (3.41), equation (3.27) can now be rewritten as:

$$\begin{aligned}\mathbf{x}(t) &= \sum_{k=0}^{\infty} \left( \mathbf{A}^k \mathbf{x}(0) \frac{\Gamma(1)}{\Gamma(k\alpha+1)} t^{k\alpha} + \mathbf{A}^k \mathbf{B} I^{\alpha(k+1)} \mathbf{u}(t) \right) \\ \mathbf{x}(t) &= \sum_{k=0}^{\infty} \mathbf{A}^k \mathbf{x}(0) \frac{\Gamma(1)}{\Gamma(k\alpha+1)} t^{k\alpha} + \sum_{k=0}^{\infty} \mathbf{A}^k \mathbf{B} I^{\alpha(k+1)} \mathbf{u}(t)\end{aligned}\quad (3.42a)$$

$$\begin{aligned}\mathbf{x}(t) &= \sum_{k=0}^{\infty} \left( \frac{\mathbf{A}^k t^{k\alpha}}{\Gamma(k\alpha+1)} \mathbf{x}(0) + \mathbf{A}^k \mathbf{B} I^{\alpha(k+1)} \mathbf{u}(t) \right) \\ \mathbf{x}(t) &= \sum_{k=0}^{\infty} \frac{\mathbf{A}^k t^{k\alpha}}{\Gamma(k\alpha+1)} \mathbf{x}(0) + \sum_{k=0}^{\infty} \mathbf{A}^k \mathbf{B} I^{\alpha(k+1)} \mathbf{u}(t)\end{aligned}\quad (3.42b)$$

$$\begin{aligned}
\mathbf{x}(t) &= \sum_{k=0}^{\infty} \frac{\mathbf{A}^k t^{k\alpha}}{\Gamma(k\alpha + 1)} \mathbf{x}(0) + \sum_{k=0}^{\infty} \mathbf{A}^k \mathbf{B} t^{\alpha(k+1)} \mathbf{u}(t) \\
\mathbf{x}(t) &= \sum_{k=0}^{\infty} \frac{(\mathbf{A}t^\alpha)^k}{\Gamma(k\alpha + 1)} \mathbf{x}(0) + \int_0^t \sum_{k=0}^{\infty} \frac{\mathbf{A}^k (t-\tau)^{(k+1)\alpha-1}}{\Gamma((k+1)\alpha)} \mathbf{B} \mathbf{u}(\tau) d\tau
\end{aligned} \tag{3.42c}$$

In terms of the Mittag-Leffler function in Appendix A3, equation (3.42) can be written as:

$$\mathbf{x}(t) = E_{\alpha,1}(\mathbf{A}t^\alpha) \mathbf{x}(0) + \left[ t^{\alpha-1} E_{\alpha,\alpha}(\mathbf{A}t^\alpha) \right] \left[ \mathbf{B} \mathbf{u}(\tau) d\tau \right] \tag{3.43}$$

Substituting (3.43) into (3.4b), the general solution of the massive MIMO system having used fractional-order system identification can be written as:

$$\mathbf{y}(t) = \mathbf{C} \left( E_{\alpha,1}(\mathbf{A}t^\alpha) \mathbf{x}(0) + \left[ t^{\alpha-1} E_{\alpha,\alpha}(\mathbf{A}t^\alpha) \right] \left[ \mathbf{B} \mathbf{u}(t) \right] \right) + \mathbf{D} \mathbf{u}(t) \tag{3.44}$$

In most cases it is not easy to obtain the analytical solution of a linear fractional-order system having been given the system input and system matrices. To address this problem an alternative technique that can be used to obtain the output of the fractional-order massive MIMO system is discussed in the following sections.

### 3.3 Fractional-Order Realisations / Integer-Order Approximation

Using the estimated values of  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$  and  $\mathbf{D}$ , the estimated fractional-order transfer function of the massive MIMO system is written as:

$$\hat{\mathbf{G}}(s) = \hat{\mathbf{C}} \left( s^\alpha \mathbf{I}_n - \hat{\mathbf{A}} \right)^{-1} \hat{\mathbf{B}} + \hat{\mathbf{D}} \tag{3.45}$$

where  $(\hat{\cdot})$  indicates an estimate.

In the practical sense, transfer functions such as (3.45) are not easy to implement, and this has led to the rise in rational transfer functions that can be used to approximate these fractional-order transfer functions. This means that whenever there is a fractional-order transfer function in system identification there is need to replace it with an easier to handle approximate rational transfer function. The following sections deal with these approximate rational transfer functions.

The realisation of fractional-order systems is complicated and in practice fractional-order operators in continuous-time are approximated using different methods, namely Oustaloup's method, Carlson's method and Charef's method [84]. In these methods an equivalent continuous-time rational model obtained from approximating a fractional-order differentiation operator by a rational one is used to get the fractional-order model output. The fractional behaviour of systems is usually limited within a specific frequency range, i.e. lower frequency and upper frequency denoted as  $(\omega_L, \omega_H)$  [91], where  $\omega_L = 2\pi f_L$  and  $\omega_H = 2\pi f_H$ . The lower frequency is limited by the input data spectrum, whilst the upper frequency is limited by the sampling period. Thus fractional-order systems must have the same dynamics as their approximated continuous-time rational counterparts within that specific frequency range.

The Carlson's realisation performance is sometimes very poor and on top of that the frequency intervals of interest cannot be specified by the user which is another drawback of the Carlson's realisation. The Charef's realisation is used for irrational systems which cannot be modelled exactly with the standard form of fractional-order transfer functions. It is a useful tool in finding integer-order fitting of the irrational model. Lastly, the Oustaloup's realisation is used in the simulation of complicated fractional-order systems by imitating the fractional-order derivative and integral actions. In this realisation the frequency intervals of interest can be selected by the user [92].

### 3.3.1 Carlson's Realisation

Let  $H(s)$  be the fractional-order transfer function and  $G(s)$  be a rational transfer function. Carlson's realisation aims to find a rational approximation  $H(s)$  to a model with a fractional power of integer-order  $G(s)$  such that:

$$H(s) \approx G^q(s) \quad (3.46)$$

where  $q = \frac{m}{p}$  is a fractional-order of the transfer function. Using the recursive zeros

and poles approximation,  $H(s)$  can be approximated as [84]:

$$H_i(s) = H_{i-1}(s) \frac{(p-m)H_{i-1}^2(s) + (p+m)G(s)}{(p+m)H_{i-1}^2(s) + (p-m)G(s)} \quad (3.47)$$

with the initial model,  $H_0(s)$  set to  $H_0(s) = 1$

Equation (3.47) can be rewritten as:

$$H_i(s) = H_{i-1}(s) \frac{G(s) + \alpha H_{i-1}^2(s)}{\alpha G(s) + H_{i-1}^2(s)} \quad (3.48)$$

where  $\alpha := \frac{p-m}{p+m} = \frac{1 - \left(\frac{m}{p}\right)}{1 + \left(\frac{m}{p}\right)} = \frac{1-q}{1+q}$  and  $q := \frac{1-\alpha}{1+\alpha}$

### 3.3.2 Charef's Realisation

This is based on finding an integer-order approximation of a fractional-order transfer function of the form:

$$H(s) = \frac{1}{\left(1 + \frac{s}{p_T}\right)^m} \quad (3.49)$$

where  $p_T$  is the pole of the fractional-order transfer function and  $m$  is the fractional-order of the transfer function. Again using the approximation with recursive zeros and poles, the approximation of (3.49) with a maximum allowable deviation of  $\delta$  dB from the original magnitude response in the frequency band  $(\omega_L, \omega_H)$  is given by [93]:

$$H(s) = \frac{\prod_{i=0}^{F-1} \left(1 + \frac{s}{z_i}\right)}{\prod_{i=0}^F \left(1 + \frac{s}{p_i}\right)} = \frac{\left(1 + \frac{s}{z_0}\right)\left(1 + \frac{s}{z_1}\right)\cdots\left(1 + \frac{s}{z_{N-1}}\right)}{\left(1 + \frac{s}{p_0}\right)\left(1 + \frac{s}{p_1}\right)\cdots\left(1 + \frac{s}{p_N}\right)} \quad (3.50)$$

where  $F$  is the number of zeros and poles. The performance of the approximation is strongly dependent on its approximation parameter choice, i.e. the smaller the value of  $F$  the lower the system order meaning simpler approximations which imply less calculations to be performed.

The approximation of the first pole,  $p_0$  is given as:

$$p_0 = p_T 10^{\left(\frac{\delta}{20m}\right)} \quad (3.51)$$

The other poles and zeros can be calculated as:

$$p_{i+1} = p_0 (ab)^{i+1}, \quad i = 0, 1, \dots, F-1 \quad (3.52)$$

$$z_i = ap_i, \quad i = 0, 1, \dots, F-1 \quad (3.53)$$

where  $a = 10^{\left(\frac{\delta}{10(1-m)}\right)}$  and  $b = 10^{\left(\frac{\delta}{10m}\right)}$ .

The value of  $F$  can be obtained as:

$$F = \text{integer} \left[ \frac{\log\left(\frac{\omega_{\max}}{p_0}\right)}{\log(ab)} \right] + 1 \quad (3.54)$$

where  $\omega_{\max} = 100\omega_H = 200\pi f_H$  [94].

### 3.3.3 Oustaloup's Realisation

The Oustaloup's realisation makes use of a recursive distribution of poles and zeros and is based on approximating a function of the form [93] and [95]:

$$G(s) = s^m, \quad m \in \mathbb{R}^+ \quad (3.55)$$

within the frequency band  $(\omega_L, \omega_H)$ , by a rational transfer function of the form:

$$H(s) = K \prod_{i=-F}^F \left( \frac{1 + \frac{s}{\omega_i}}{1 + \frac{s}{\omega_i}} \right) \quad (3.56)$$

where  $K = \omega_H^m$ ,  $\omega_i = \omega_L \left( \frac{\omega_H}{\omega_L} \right)^{\frac{i+F+\frac{1}{2}(1+m)}{2F+1}}$  and  $\omega_i = \omega_L \left( \frac{\omega_H}{\omega_L} \right)^{\frac{i+F+\frac{1}{2}(1-m)}{2F+1}}$  for  $i = 1, 2, \dots, F, F$

is the order of the filter.

All the segments in the asymptotes are generated by integer-order poles and zeros such that the slopes of the magnitude asymptotes are alternating at 0 dB/dec and -20 dB/dec. If an adequate number of segments is selected, the shapes of the exact Bode magnitude may look like a straight line in the range. Therefore, the polylines will approximate the straight line in a very close manner.

Given its advantage over the Carlson's and Charef's realisations and given that our model is a rational model, the Oustaloup's realisation was chosen when approximating the continuous-time fractional-order model.

In our discussion so far, we ignored the effects of the noise when identifying the ISI channel for the massive MIMO system, but in real life system identification the effects of this noise have to be considered. The next section discusses the influence of the additive white Gaussian noise in the fractional-order massive MIMO system identification.

### 3.4 Fractional-Order System Identification with Measurement Noise Considered

In this case, fractional-order system identification is performed having considered the measurement noise. Using continuous-time fractional-order state-space modelling, the dynamics of the linear time invariant length- $L$  ISI channel massive MIMO system can be expressed as:

$$D^\alpha \mathbf{x}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad (3.57a)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) + \mathbf{n}(t) \quad (3.57b)$$

where  $\mathbf{x}(t)$  is the  $n \times 1$  state vector,  $\mathbf{u}(t)$  is the  $m \times 1$  input vector,  $\mathbf{y}(t)$  is the  $p \times 1$  output vector,  $\mathbf{n}(t)$  is the  $p \times 1$  additive white Gaussian noise vector,  $\mathbf{A}$  is the  $n \times n$  system matrix and it describes the dynamics of the system, i.e. the eigenvalues of the system,  $\mathbf{B}$  is the  $n \times m$  input matrix and it describes the linear transformation by which the inputs influence the next state,  $\mathbf{C}$  is the  $p \times n$  output matrix, and it describes how the state is transferred to the output,  $\mathbf{D}$  is the  $p \times m$  feed-forward matrix,  $\alpha$  is the commensurate fractional-order and  $D^\alpha$  is the fractional derivative of order  $\alpha$ .

Applying the Laplace transform to (3.57) results in:

$$\begin{aligned} s^\alpha \mathbf{X}(s) &= \mathbf{A}\mathbf{X}(s) + \mathbf{B}\mathbf{U}(s) \\ \mathbf{Y}(s) &= \mathbf{C}\mathbf{X}(s) + \mathbf{D}\mathbf{U}(s) + \mathbf{N}(s) \end{aligned} \quad (3.58)$$

Taking the  $\alpha$ -order derivative of (3.57b) yields:

$$D^\alpha \mathbf{y}(t) = \mathbf{C}D^\alpha \mathbf{x}(t) + \mathbf{D}D^\alpha \mathbf{u}(t) + D^\alpha \mathbf{n}(t) \quad (3.59)$$

Substituting (3.57a) into (3.59) yields:

$$\begin{aligned} D^\alpha \mathbf{y}(t) &= \mathbf{C}(\mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)) + \mathbf{D}D^\alpha \mathbf{u}(t) + D^\alpha \mathbf{n}(t) \\ D^\alpha \mathbf{y}(t) &= \mathbf{C}\mathbf{A}\mathbf{x}(t) + \mathbf{C}\mathbf{B}\mathbf{u}(t) + \mathbf{D}D^\alpha \mathbf{u}(t) + D^\alpha \mathbf{n}(t) \end{aligned} \quad (3.60)$$

Taking the  $2\alpha$ -order derivative of (3.57b) and making the necessary simplifications yields:

$$D^{2\alpha} \mathbf{y}(t) = \mathbf{C}\mathbf{A}D^\alpha \mathbf{x}(t) + \mathbf{C}\mathbf{B}D^\alpha \mathbf{u}(t) + \mathbf{D}D^{2\alpha} \mathbf{u}(t) + D^{2\alpha} \mathbf{n}(t) \quad (3.61)$$

Substituting (3.57a) into (3.61) yields:

$$\begin{aligned} D^{2\alpha} \mathbf{y}(t) &= \mathbf{C}\mathbf{A}(\mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)) + \mathbf{C}\mathbf{B}D^\alpha \mathbf{u}(t) + \mathbf{D}D^{2\alpha} \mathbf{u}(t) + D^{2\alpha} \mathbf{n}(t) \\ D^{2\alpha} \mathbf{y}(t) &= \mathbf{C}\mathbf{A}^2 \mathbf{x}(t) + \mathbf{C}\mathbf{A}\mathbf{B}\mathbf{u}(t) + \mathbf{C}\mathbf{B}D^\alpha \mathbf{u}(t) + \mathbf{D}D^{2\alpha} \mathbf{u}(t) + D^{2\alpha} \mathbf{n}(t) \end{aligned} \quad (3.62)$$

Taking the  $3\alpha$ -order derivative of (3.57b) and making the necessary simplifications yields:

$$D^{3\alpha} \mathbf{y}(t) = \mathbf{C}\mathbf{A}^2 D^\alpha \mathbf{x}(t) + \mathbf{C}\mathbf{A}\mathbf{B}D^\alpha \mathbf{u}(t) + \mathbf{C}\mathbf{B}D^{2\alpha} \mathbf{u}(t) + \mathbf{D}D^{3\alpha} \mathbf{u}(t) + D^{3\alpha} \mathbf{n}(t) \quad (3.63)$$

Substituting (3.57a) into (3.63) yields:

$$\begin{aligned}
D^{3\alpha} \mathbf{y}(t) &= \mathbf{CA}^2 (\mathbf{Ax}(t) + \mathbf{Bu}(t)) + \mathbf{CABD}^\alpha \mathbf{u}(t) + \mathbf{CBD}^{2\alpha} \mathbf{u}(t) + \mathbf{DD}^{3\alpha} \mathbf{u}(t) + D^{3\alpha} \mathbf{n}(t) \\
D^{3\alpha} \mathbf{y}(t) &= \mathbf{CA}^3 \mathbf{x}(t) + \mathbf{CA}^2 \mathbf{Bu}(t) + \mathbf{CABD}^\alpha \mathbf{u}(t) + \mathbf{CBD}^{2\alpha} \mathbf{u}(t) + \mathbf{DD}^{3\alpha} \mathbf{u}(t) + D^{3\alpha} \mathbf{n}(t)
\end{aligned} \tag{3.64}$$

and so on till the  $(i-1)\alpha$ -order derivative.

Stacking (3.57b) and all its  $\alpha$ -order derivatives obtained above up to order  $(i-1)\alpha$  into a column vector, the input-output relationship can be written as:

$$\underbrace{\begin{bmatrix} \mathbf{y}(t) \\ D^\alpha \mathbf{y}(t) \\ \vdots \\ D^{(i-1)\alpha} \mathbf{y}(t) \end{bmatrix}}_{\mathbf{Y}_{0(i-1)\alpha}(t)} = \underbrace{\begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{(i-1)} \end{bmatrix}}_{\mathbf{\Gamma}_i} \mathbf{x}(t) + \underbrace{\begin{bmatrix} \mathbf{D} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CB} & \mathbf{D} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{(i-2)} \mathbf{B} & \cdots & \mathbf{CB} & \mathbf{D} \end{bmatrix}}_{\mathbf{\Omega}} \underbrace{\begin{bmatrix} \mathbf{u}(t) \\ D^\alpha \mathbf{u}(t) \\ \vdots \\ D^{(i-1)\alpha} \mathbf{u}(t) \end{bmatrix}}_{\mathbf{U}_{0(i-1)\alpha}(t)} + \underbrace{\begin{bmatrix} \mathbf{n}(t) \\ D^\alpha \mathbf{n}(t) \\ \vdots \\ D^{(i-1)\alpha} \mathbf{n}(t) \end{bmatrix}}_{\mathbf{N}_{0(i-1)\alpha}(t)} \tag{3.65}$$

where  $i$  is a user defined index .

In compact form (3.65) can be written as:

$$\mathbf{Y}_{0(i-1)\alpha}(t) = \mathbf{\Gamma}_i \mathbf{x}(t) + \mathbf{\Omega}_i \mathbf{U}_{0(i-1)\alpha}(t) + \mathbf{N}_{0(i-1)\alpha}(t) \tag{3.66}$$

Sampling (3.66) at sampling instances  $t_1, t_2, \dots, t_N$  results in the following input,

$\mathbf{U}_{0(i-1)\alpha, N}$  output,  $\mathbf{Y}_{0(i-1)\alpha, N}$ , noise,  $\mathbf{N}_{0(i-1)\alpha, N}$  and state,  $\mathbf{X}_N$  matrices:

$$\mathbf{U}_{0(i-1)\alpha, N} = \begin{bmatrix} \mathbf{u}(t_1) & \mathbf{u}(t_2) & \cdots & \mathbf{u}(t_N) \\ D^\alpha \mathbf{u}(t_1) & D^\alpha \mathbf{u}(t_2) & \cdots & D^\alpha \mathbf{u}(t_N) \\ \vdots & \vdots & \cdots & \vdots \\ D^{(i-1)\alpha} \mathbf{u}(t_1) & D^{(i-1)\alpha} \mathbf{u}(t_2) & \cdots & D^{(i-1)\alpha} \mathbf{u}(t_N) \end{bmatrix} \in \mathbb{R}^{im \times N} \tag{3.67}$$

$$\mathbf{Y}_{0(i-1)\alpha,N} = \begin{bmatrix} \mathbf{y}(t_1) & \mathbf{y}(t_2) & \cdots & \mathbf{y}(t_N) \\ D^\alpha \mathbf{y}(t_1) & D^\alpha \mathbf{y}(t_2) & \cdots & D^\alpha \mathbf{y}(t_N) \\ \vdots & \vdots & \cdots & \vdots \\ D^{(i-1)\alpha} \mathbf{y}(t_1) & D^{(i-1)\alpha} \mathbf{y}(t_2) & \cdots & D^{(i-1)\alpha} \mathbf{y}(t_N) \end{bmatrix} \in \mathbb{R}^{ip \times N} \quad (3.68)$$

$$\mathbf{N}_{0(i-1)\alpha,N} = \begin{bmatrix} \mathbf{n}(t_1) & \mathbf{n}(t_2) & \cdots & \mathbf{n}(t_N) \\ D^\alpha \mathbf{n}(t_1) & D^\alpha \mathbf{n}(t_2) & \cdots & D^\alpha \mathbf{n}(t_N) \\ \vdots & \vdots & \cdots & \vdots \\ D^{(i-1)\alpha} \mathbf{n}(t_1) & D^{(i-1)\alpha} \mathbf{n}(t_2) & \cdots & D^{(i-1)\alpha} \mathbf{n}(t_N) \end{bmatrix} \in \mathbb{R}^{ip \times N} \quad (3.69)$$

$$\mathbf{X}_N = [\mathbf{x}(t_1) \quad \mathbf{x}(t_2) \quad \cdots \quad \mathbf{x}(t_N)] \quad (3.70)$$

Using (3.67) , (3.68), (3.69) and (3.70), equation (3.66) can then be transformed into the following equation [71]:

$$\mathbf{Y}_{0(i-1)\alpha,N} = \mathbf{\Gamma}_i \mathbf{X}_N + \mathbf{\Omega}_i \mathbf{U}_{0(i-1)\alpha,N} + \mathbf{N}_{0(i-1)\alpha,N} \quad (3.71)$$

Equation (3.71) contains successive  $(i-1)\alpha$ -order fractional derivatives of the input data, output data and noise which in most practical cases are not measured. To address this problem the Poisson moment functional (PMF) approach is applied, details of which are presented in Appendix A3.

Applying PMF to (3.71), the PMF filtered input-output data and noise can be expressed as:

$$\mathbf{P}_t^r [\mathbf{Y}_{0(i-1)\alpha,N}] = \mathbf{\Gamma}_i \mathbf{P}_t^r [\mathbf{X}_N] + \mathbf{\Omega}_i \mathbf{P}_t^r [\mathbf{U}_{0(i-1)\alpha,N}] + \mathbf{P}_t^r [\mathbf{N}_{0(i-1)\alpha,N}] \quad (3.72)$$

The idea is to estimate the observability matrix given the input-output data only, but equation (3.72) also has a noise term,  $\mathbf{P}_t^r [\mathbf{N}_{0(i-1)\alpha,N}]$  which affects the geometrical properties of the MOESP algorithm [15], for that reason there is need to reduce the effects of this noise term on the overall filtered data. To address this problem, the

authors in [72] proposed correlating the PMF filtered output data,  $\mathbf{P}_t^r [\mathbf{Y}_{0(i-1)\alpha, N}]$  with instrumental variables (IV). For this to work, the noise variables in the output must be uncorrelated with those of the instrumental variables and on the other hand,  $\mathbf{P}_t^r [\mathbf{Y}_{0(i-1)\alpha, N}]$  must remain undistorted. This can be achieved by partitioning the input-output data and noise into two parts, namely the past and the future parts. The future output part can then be used as an observation vector, whilst the past input and output parts can be used as instrumental variables. The partitions can be defined as [72]:

$$\mathbf{P}_t^r [\mathbf{U}_{0(i-1)\alpha, N}] = \begin{bmatrix} \mathbf{P}_t^r [\mathbf{U}_p] \\ \mathbf{P}_t^r [\mathbf{U}_f] \end{bmatrix} \quad (3.73)$$

$$\mathbf{P}_t^r [\mathbf{Y}_{0(i-1)\alpha, N}] = \begin{bmatrix} \mathbf{P}_t^r [\mathbf{Y}_p] \\ \mathbf{P}_t^r [\mathbf{Y}_f] \end{bmatrix} \quad (3.74)$$

$$\mathbf{P}_t^r [\mathbf{N}_{0(i-1)\alpha, N}] = \begin{bmatrix} \mathbf{P}_t^r [\mathbf{N}_p] \\ \mathbf{P}_t^r [\mathbf{N}_f] \end{bmatrix} \quad (3.75)$$

where  $\mathbf{P}_t^r [\mathbf{U}_p]$ ,  $\mathbf{P}_t^r [\mathbf{Y}_p]$  and  $\mathbf{P}_t^r [\mathbf{N}_p]$  denote the past input, output and noise respectively, and  $\mathbf{P}_t^r [\mathbf{U}_f]$ ,  $\mathbf{P}_t^r [\mathbf{Y}_f]$  and  $\mathbf{P}_t^r [\mathbf{N}_f]$  denote the future input, output and noise respectively. The past data is defined for index 0 to  $(i-1)\alpha$ , whilst the future data is defined for index  $i\alpha$  to  $(2i-1)\alpha$ .

An instrumental variables matrix,  $\mathbf{V}$  can be constructed using the past input and output parts as:

$$\mathbf{V} = \begin{bmatrix} \mathbf{P}_t^r [\mathbf{U}_p] \\ \mathbf{P}_t^r [\mathbf{Y}_p] \end{bmatrix} \quad (3.76)$$

Considering only the future parts, (3.72) can be rewritten as:

$$\mathbf{P}_t^r [\mathbf{Y}_f] = \mathbf{\Gamma}_i \mathbf{P}_t^r [\mathbf{X}_N] + \mathbf{\Omega}_t \mathbf{P}_t^r [\mathbf{U}_f] + \mathbf{P}_t^r [\mathbf{N}_f] \quad (3.77)$$

Equation (3.77) is then multiplied by (3.76) to help reduce the effects of the noise term on  $\mathbf{P}_t^r [\mathbf{Y}_f]$ , yielding:

$$\mathbf{P}_t^r [\mathbf{Y}_f] \mathbf{V} = \mathbf{\Gamma}_i \mathbf{P}_t^r [\mathbf{X}_N] \mathbf{V} + \mathbf{\Omega}_t \mathbf{P}_t^r [\mathbf{U}_f] \mathbf{V} + \mathbf{P}_t^r [\mathbf{N}_f] \mathbf{V} \quad (3.78)$$

Since the instrumental variables matrix,  $\mathbf{V}$  is independent of the noise term, the noise term in (3.78) disappears as a result of  $\mathbf{P}_t^r [\mathbf{N}_f] \mathbf{V} = \mathbf{0}$ , resulting in:

$$\mathbf{P}_t^r [\mathbf{Y}_f] \mathbf{V} = \mathbf{\Gamma}_i \mathbf{P}_t^r [\mathbf{X}_N] \mathbf{V} + \mathbf{\Omega}_t \mathbf{P}_t^r [\mathbf{U}_f] \mathbf{V} \quad (3.79)$$

Having addressed the issue with the noise term, the observability matrix can now be obtained using LQ decomposition and SVD. In (3.57b), given that additive white Gaussian noise is considered, the instrumental variables are chosen to follow the past output multiple-input multiple-output output-error state space (PO-MOESP) algorithm when computing the LQ decomposition and SVD of the filtered input-output data as follows [73]:

Applying the LQ decomposition to matrix (3.79) yields [74]:

$$\begin{bmatrix} \mathbf{P}_t^r [\mathbf{U}_f] \\ \mathbf{P}_t^r [\mathbf{U}_p] \\ \mathbf{P}_t^r [\mathbf{Y}_p] \\ \mathbf{P}_t^r [\mathbf{Y}_f] \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{11} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{L}_{21} & \mathbf{L}_{22} & \mathbf{0} & \mathbf{0} \\ \mathbf{L}_{31} & \mathbf{L}_{32} & \mathbf{L}_{33} & \mathbf{0} \\ \mathbf{L}_{41} & \mathbf{L}_{42} & \mathbf{L}_{43} & \mathbf{L}_{44} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \\ \mathbf{Q}_3 \\ \mathbf{Q}_4 \end{bmatrix} \quad (3.80)$$

where  $\mathbf{L}_{11}, \mathbf{L}_{22} \in \mathbb{R}^{im \times im}$ ,  $\mathbf{L}_{33}, \mathbf{L}_{44} \in \mathbb{R}^{ip \times ip}$  are lower triangular matrices, and  $\mathbf{Q}_1, \mathbf{Q}_2 \in \mathbb{R}^{N \times im}$ ,  $\mathbf{Q}_3, \mathbf{Q}_4 \in \mathbb{R}^{N \times ip}$  are orthogonal matrices.

Using (3.80) the SVD of  $[\mathbf{L}_{42} \quad \mathbf{L}_{43}]$  is computed as:

$$[\mathbf{L}_{42} \quad \mathbf{L}_{43}] = [\mathbf{U}_1 \quad \mathbf{U}_2] \begin{bmatrix} \mathbf{S}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^H \\ \mathbf{V}_2^H \end{bmatrix} \quad (3.81)$$

where  $\mathbf{S}_1$  is a diagonal matrix containing the singular values different from zero and the order of the system,  $n$  can be estimated from  $\mathbf{S}_1$  as the number of singular values different from zero.

The extended observability matrix,  $\mathbf{\Gamma}_i$  can then be estimated from (3.81) as:

$$\hat{\mathbf{\Gamma}}_i = \mathbf{U}_1 \mathbf{S}_1^{\frac{1}{2}} \quad (3.82)$$

The procedure for estimating matrices  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$  and  $\mathbf{D}$  is as discussed for the MOESP algorithm.

Having presented a discussion on the fractional-order MOESP and fractional-order PO-MOESP algorithms for the massive MIMO system, the next step is to study the performance of these algorithms. The following section presents the MATLAB simulation results and analysis of these algorithms having been applied to the massive MIMO system.

### 3.5 Simulation Results

The simulations were run in MATLAB having considered the input signal,  $\mathbf{u}(t)$  to be a chirp signal with frequency ranging from 20Hz to 20 kHz with a sampling frequency twice the highest frequency range, which is a persistently excitation signal. This frequency range was chosen so as to cover the range of audible frequencies for humans. The chirp excitation signal is shown in Figure 3.2 The fractional-order  $\alpha$  was chosen to be  $\alpha = 0.1$ . This is because for fractional-orders greater than  $\alpha = 0.1$  the system proved

to be unstable, i.e. the system did not meet the stability criteria for fractional-order systems as stated in (3.7) and (3.8). Given that we assumed zero initial conditions for our fractional-order model, the Riemann-Liouville definition was chosen for our simulations.

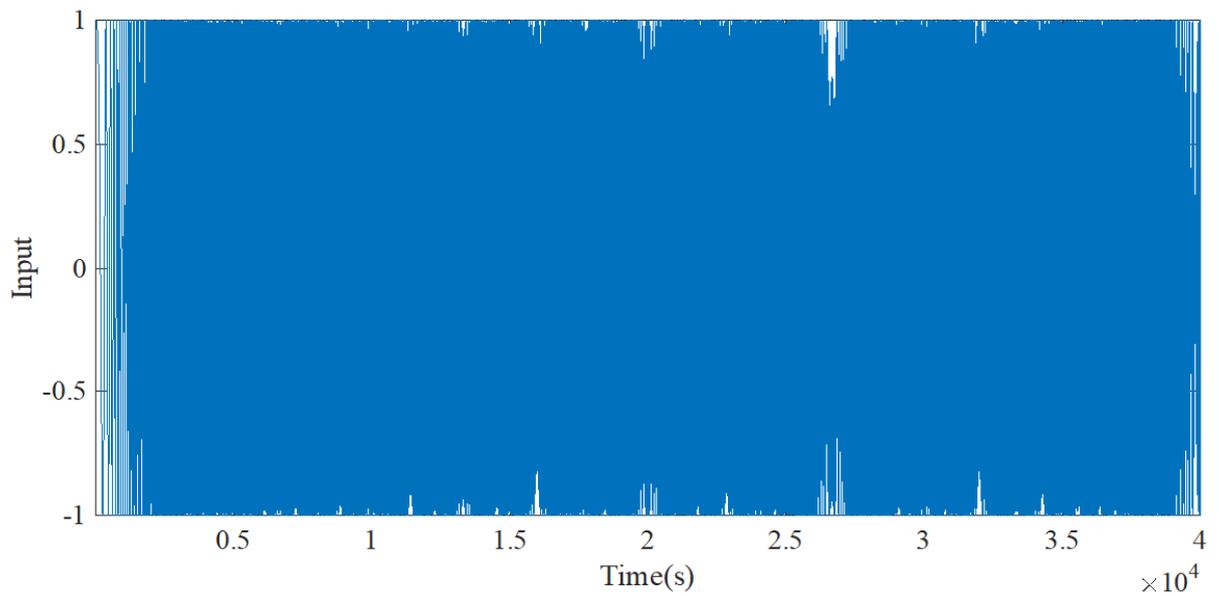


Figure 3. 2: Chirp signal for the input

Through our simulations we noted that not all these data points are needed for system identification. As a result, we reduced the number of data points needed to identify the massive MIMO system, and Figure 3.3 shows the reduced length input signal on each transmitting antenna we used for system identification.

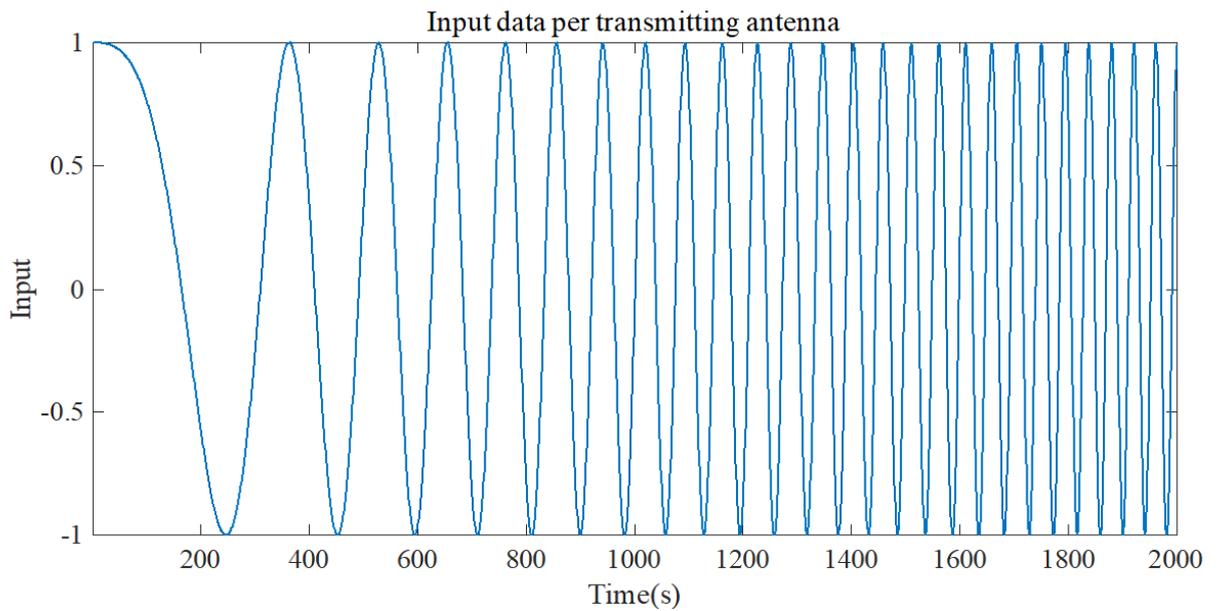


Figure 3. 3: Plot of input data

In the continuous-time fractional-order system identification algorithm the data set was divided into estimation data and validation data, where the estimation data was the first half of the data set and the second half was the validation data. The estimation data was used to estimate the system model and the estimated model was verified using the validation data. The fractional-order MOESP system identification algorithm was based on the parameters in Table 3.1

Table 3. 1: System model and system identification parameters for fractional-order model

Symbol	Description	Value
$T_x$	Number of transmitting antennas	100
$R_x$	Number of receiving antennas	1
$n$	Initialising model order	1
$\Delta t$	Sampling time	0.1s
$i$	User defined index	3
$N$	Number of samples	2001
$\alpha$	Fractional-order	0.1
$N_{est}$	Estimation data	1000
$N_{val}$	Validation data	1000
$SNR_{dB}$	Signal-to-noise ratio	20
$L$	Number of channel paths	3
$r$	Poisson filter order	8
$\beta$	Poisson filter gain	1
$\lambda$	Poisson filter constant	1

### 3.5.1 Selection of Identification Parameters

In system identification the performance of the model depends on how much data is available to describe the system and whether the sampling time is good enough to capture the dynamics of the system. In our simulations, the sampling time was varied between different sampling times and sampling time,  $\Delta t = 0.1s$  was identified as the sampling time that could reliably capture the system dynamics. The user defined index,  $i$  was selected to be seven as for lower values of  $i$  the dynamics of the massive MIMO system could not be well captured. The selection of the PMF filter order,  $r$  was based on the criteria defined in (A3.32) and the selection of the Poisson filter gain,  $\beta$  and Poisson filter constant,  $\lambda$  was based on the criteria defined in (A3.33), where for simplicity we assumed that,  $\lambda = \beta = 1$ .

The fractional-order system identification was implemented in MATLAB and the results are shown in Figure 3.4.

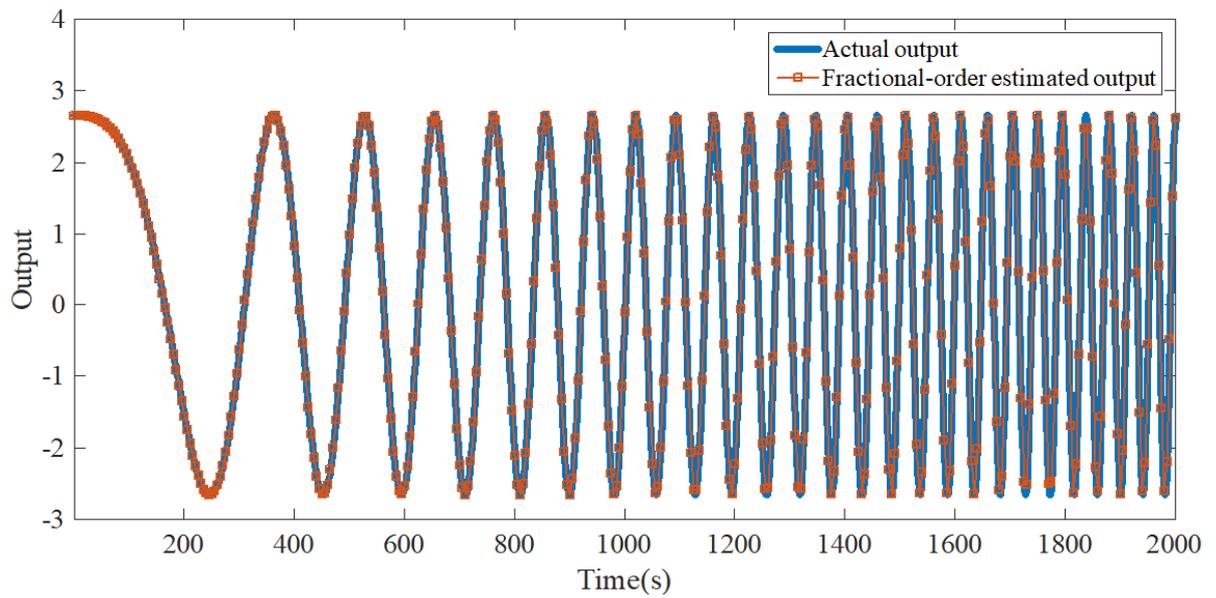


Figure 3. 4: Comparing the performance of the system output obtained using the fractional-order MOESP algorithm for  $\alpha = 0.1$  and the actual system output

We then zoomed in to show a clear picture of our results and Figure 3.5 shows the zoomed in results comparing the fractional-order MOESP algorithm estimated output for  $\alpha = 0.1$  and the actual system output.

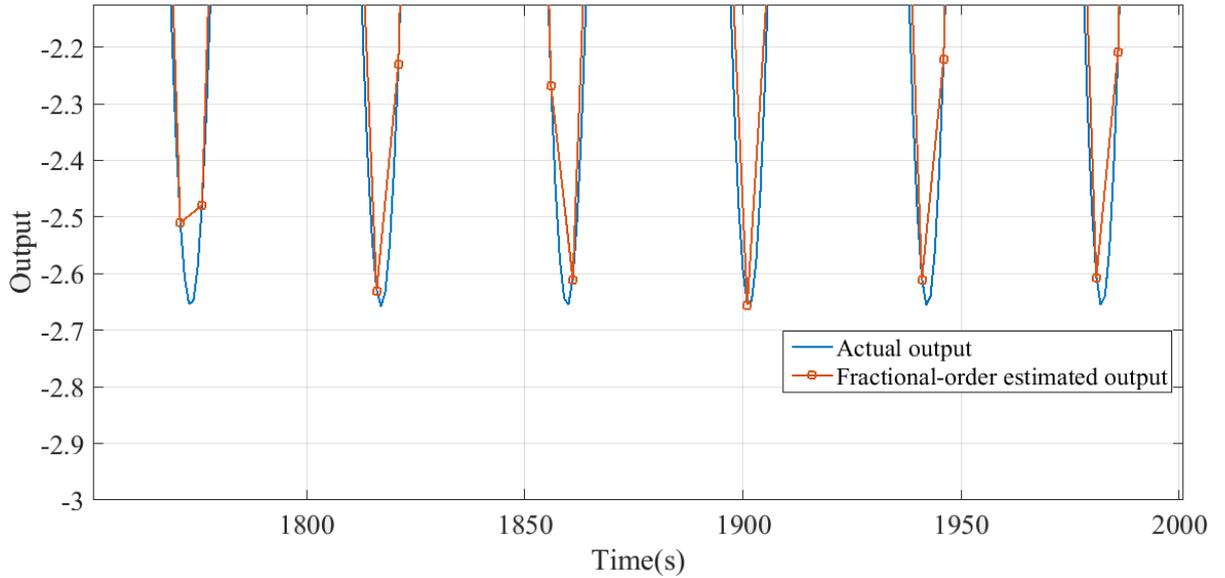


Figure 3. 5: Comparing the performance of the system output obtained using the fractional-order MOESP algorithm for  $\alpha = 0.1$  and the actual system output (zoomed in results)

These results show that the fractional-order MOESP algorithm can actually be used to identify the dynamics of the massive MIMO system when the appropriate fractional-order  $\alpha$  value is selected. The estimated fractional-order output was then compared with the measured output/ actual output, and the measure of accuracy of the proposed model is expressed as the MSE:

$$MSE = \frac{1}{N} \sum_{t=1}^N |y(t) - \hat{y}_\alpha(t)|^2 \quad (3.85)$$

where  $N$  is the number of samples and  $\hat{y}_\alpha(t)$  is the fractional-order estimated output.

Comparing the output of the fractional-order MOESP identified system and the measured output/ actual output, gives a MSE of 0.0455.

The simulation results thus far have ignored the effects of the noise when identifying the dynamics of the massive MIMO system. The next set of results take into consideration the effects of the measurement noise in fractional-order system identification, and to handle this we used the fractional-order PO-MOESP identification

algorithm. Figure 3.6 shows the performance results of the fractional-order PO-MOESP system identification algorithm

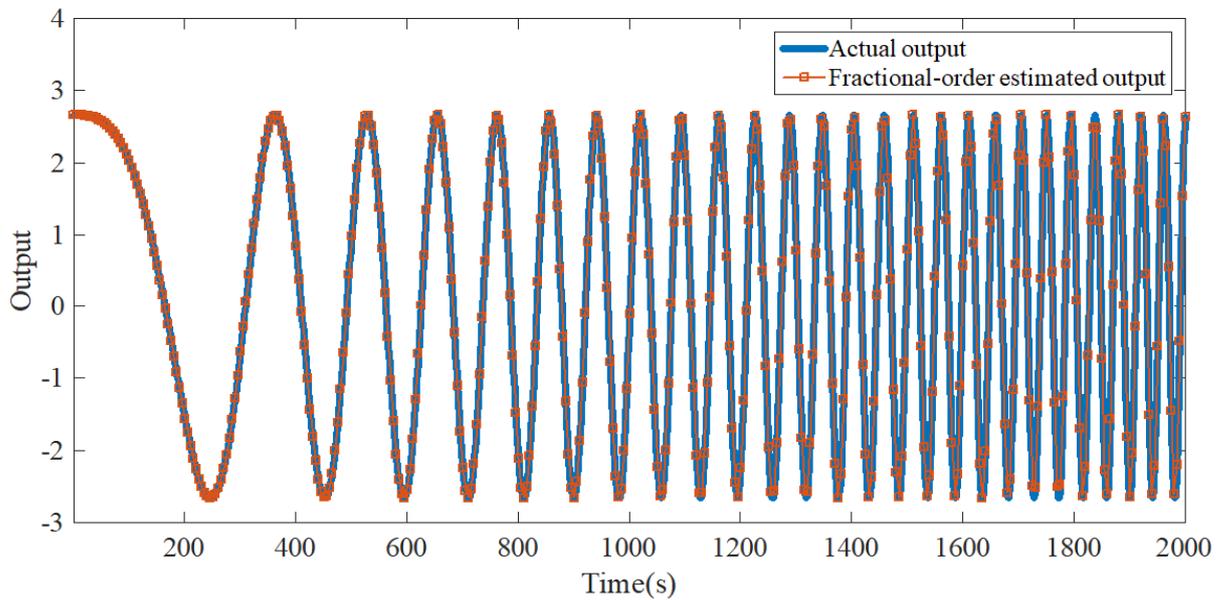


Figure 3. 6: Comparing the performance of the system output obtained using the fractional-order PO-MOESP algorithm for  $\alpha = 0.1$  and the actual system output

We then zoomed in to show a clear picture of our results and Figure 3.7 shows the zoomed in results comparing the fractional-order PO-MOESP algorithm estimated output for  $\alpha = 0.1$  and the actual system output.

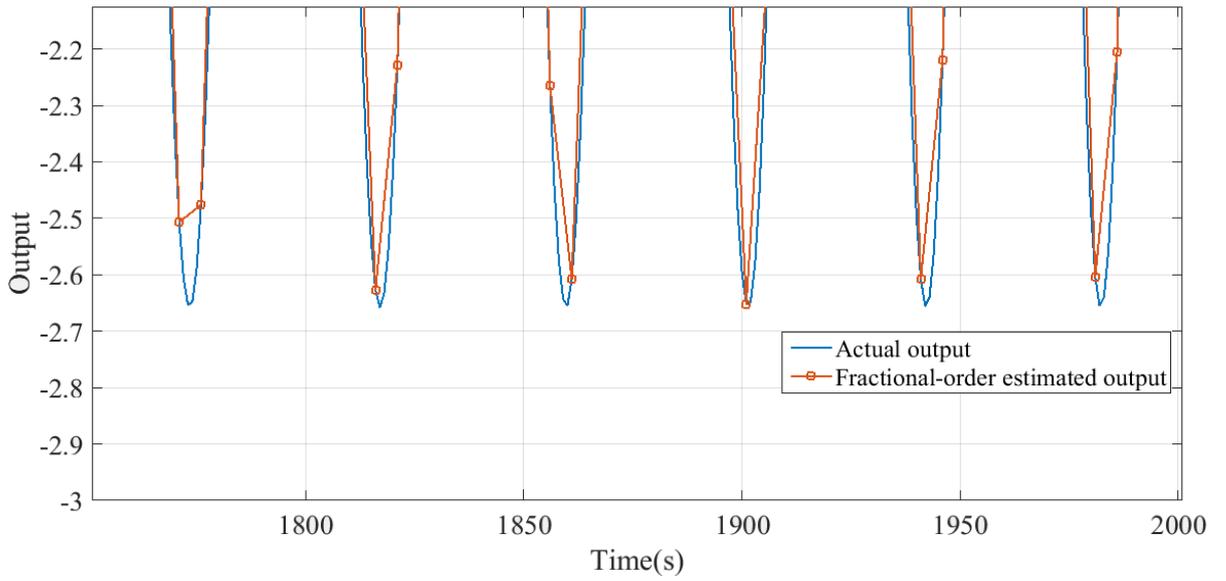


Figure 3. 7: Comparing the performance of the system output obtained using the fractional-order PO-MOESP algorithm for  $\alpha = 0.1$  and the actual system output (zoomed in results)

Figure 3.8 shows the zoomed in performance results of the fractional-order MOESP system identification algorithm, fractional-order PO-MOESP system identification algorithm for  $\alpha = 0.1$  and the actual system output.

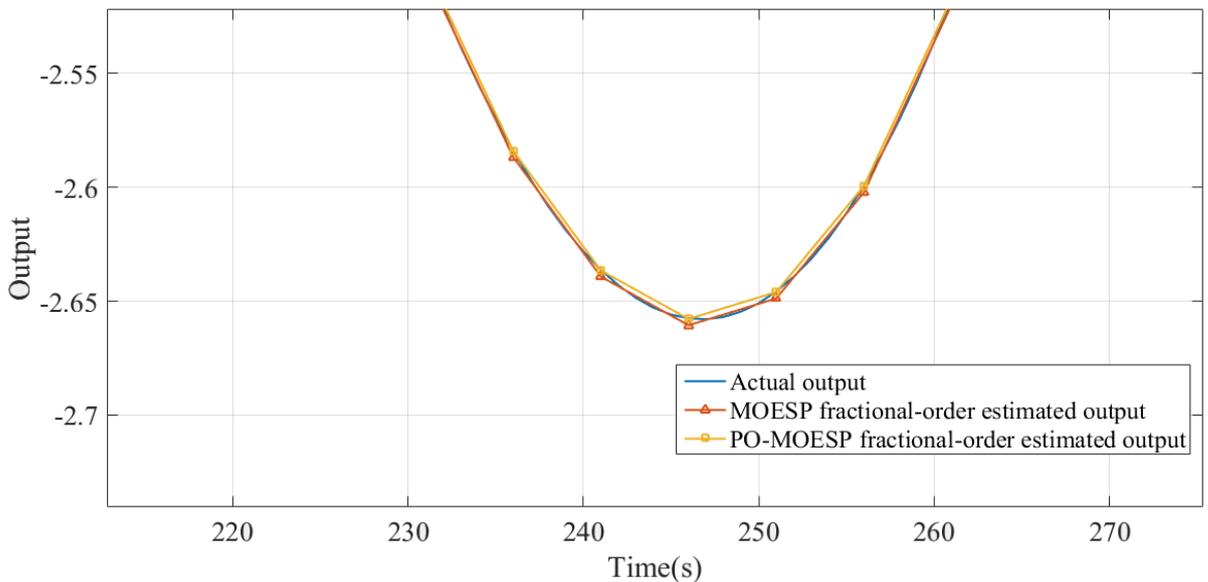


Figure 3. 8: Comparing the performance of the system output obtained using the fractional-order MOESP algorithm, fractional-order PO-MOESP algorithm for  $\alpha = 0.1$  and the actual system output (zoomed in results)

The fractional-order PO-MOESP identified system was then compared with the measured output/ actual output in terms of the MSE, giving a MSE of  $4.3218 \times 10^{-4}$ . These results show an improvement in system identification with the use of the fractional-order PO-MOESP algorithm. This is because of the introduction of the instrumental variables which help reduce the effects of the noise in the system.

Next we present the fractional-order identification results using the fractional-order realisations / integer-order approximation. Given the advantage of the Oustaloup's realisation over the Carlson's and Charef's realisations and given that our model is a rational model, the Oustaloup's realisation was chosen when approximating the continuous-time fractional-order model. The simulation parameters for the Oustaloup's realisation are shown in Table 3.2.

Table 3. 2: Simulation parameters for the Oustaloup's realisation

<b>Symbol</b>	<b>Description</b>	<b>Value</b>
$F$	Order of filter	1
$\alpha$	Fractional-order	0.1
$\omega_L$	Lower frequency limit	17750
$\omega_H$	Upper frequency limit	20000

The fractional-order  $\alpha$  was chosen to be the same value as the one used in the analytical solution of the fractional-order system model. The values of the lower and upper frequencies for the Oustaloup's realisation were varied to find a system model which gives the best prediction. The number of iterations,  $F$  in the Oustaloup's realisation was selected to be the lowest possible value. Figure 3.9 shows the performance of the Oustaloup's realisation compared to the measured output/ actual output giving a MSE of 0.0440.

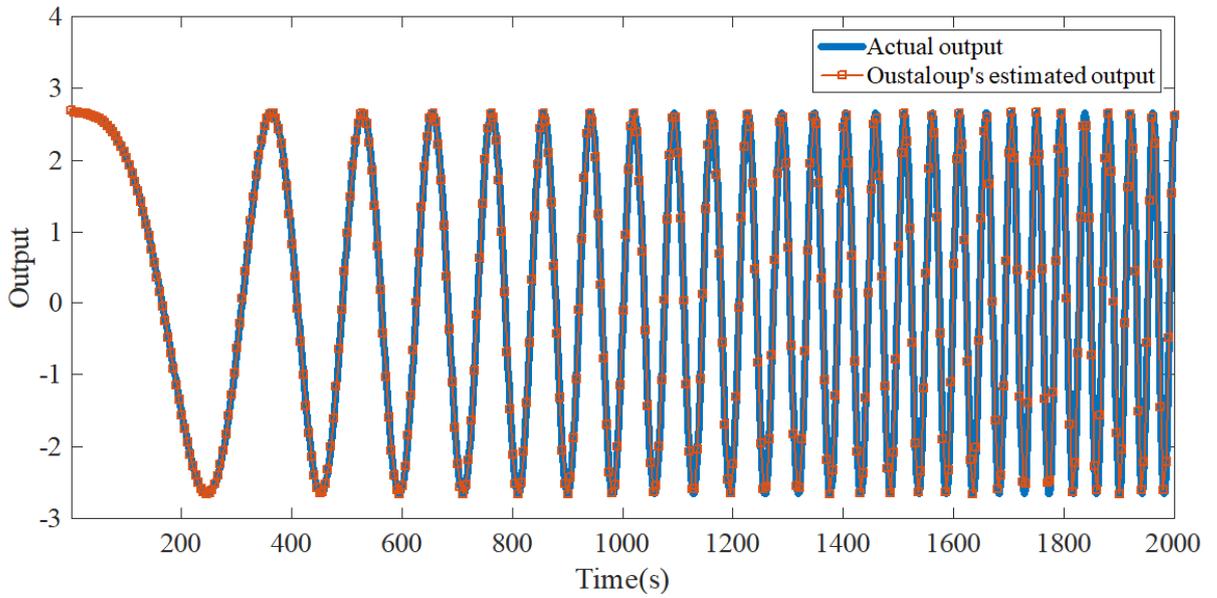


Figure 3. 9: Comparing the performance of the system output obtained using the Oustaloup's realisation for  $\alpha = 0.1$  and the actual system output

We then zoomed in to show a clear picture of our results and Figure 3.10 shows the zoomed in results comparing the Oustaloup's realisation for  $\alpha = 0.1$  and the actual system output.

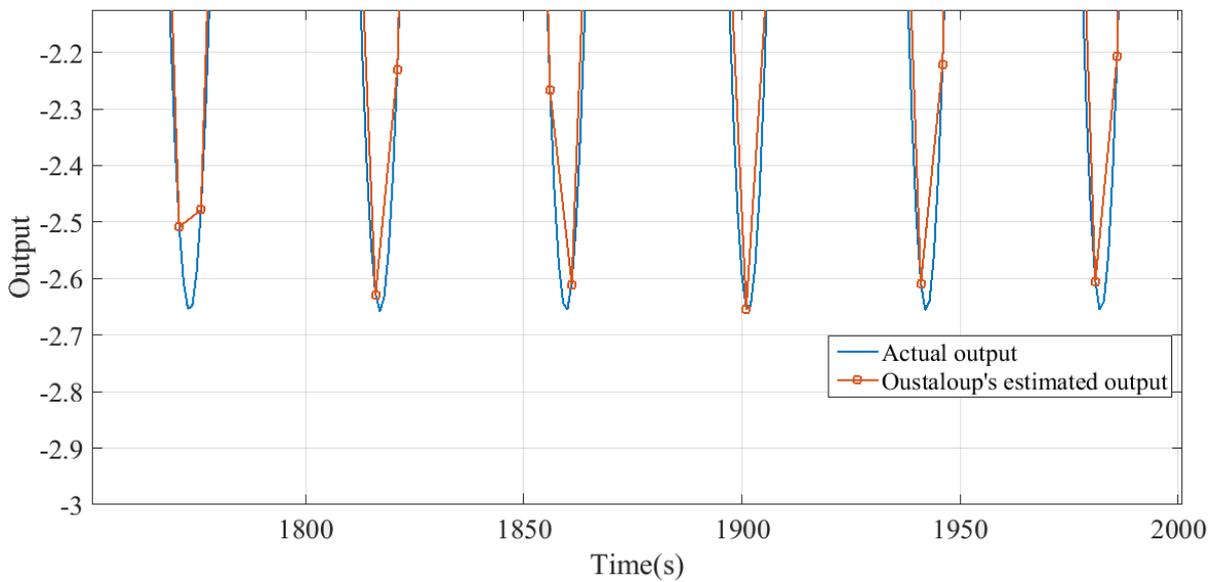


Figure 3. 10: Comparing the performance of the system output obtained using the Oustaloup's realisation for  $\alpha = 0.1$  and the actual system output

Figure 3.10 shows that with the right selection of the Oustaloup parameters the fractional-order algorithm can be used to estimate the continuous-time massive MIMO system.

### 3.6 Summary

This chapter presented the continuous-time MOESP fractional-order system identification algorithm to identify the massive MIMO frequency-selective wireless channels. Here the Riemann-Liouville definition had to be used to help transform the input-output data to fractional-order derivatives of order  $\alpha$ . Given that the fractional-order time-derivatives of the input-output data are generally not measured meaning that the input-output matrices are not known and as a result the classical subspace methods originally developed for the identification of discrete-time models cannot be directly adapted for the identification of continuous-time fractional-order models. To address this problem the Poisson moment functional (PMF) approach was used when dealing with continuous-time fractional-order system identification. The fractional-order input-output data was filtered using the PMF filter after which the MOESP identification algorithm outlined for discrete-time system identification was then applied to the PMF filtered input-output data. Analytical modelling was carried out in MATLAB to identify the massive MIMO system using the proposed algorithm.

The MATLAB simulations were then extended to include the effects of noise when identifying the dynamics of the massive MIMO system. To try to reduce the effects of noise in the identification process, the fractional-order PO-MOESP algorithm was considered. This introduces instrumental variables which play a great role in reducing the effects of noise that occur in the system.

The simulation results showed that the proposed continuous-time fractional-order MOESP system identification algorithm can actually be used to identify the dynamics of the massive MIMO system. Due to the challenge of dealing with transfer functions such as (3.29) we extended our discussion to the use of rational transfer functions that can be used to approximate these fractional-order transfer functions. This means that whenever there is a fractional-order transfer function in system

identification there is need to replace it with an easier to handle approximate rational transfer function. Given its advantage over the Carlson's and Charef's realisations, the Oustaloup's realisation was chosen when approximating the continuous-time fractional-order model. The performance of the Oustaloup realisation was studied with the help of MATLAB simulations and it was found to be a very close match to representing the massive MIMO system. This also showed that the approximate methods mainly the Oustaloup's method can also be used to identify the dynamics of the massive MIMO system.

## Appendix A3

### A3.1 Special Functions

Some important functions that are usually encountered in fractional-order calculus are discussed below but are not limited to these.

#### A3.1.1 Gamma Function

The Gamma function is important because all fractional-order calculus definitions are based on it. It is written as [96]:

$$\Gamma(x) = \int_0^{\infty} e^{-u} u^{x-1} du \quad (\text{A3.1})$$

Some useful properties of the Gamma function are [75]:

$$\Gamma(x+1) = x\Gamma(x) \quad (\text{A3.2})$$

$$\Gamma(x+1) = x! \quad (\text{A3.3})$$

$$\frac{d^{\alpha}}{dx^{\alpha}} x^{\alpha} = \frac{\Gamma(\alpha+1)}{\Gamma(\alpha-\alpha+1)} x^{\alpha-\alpha} = \Gamma(\alpha+1) \quad (\text{A3.4})$$

#### A3.1.2 Mittag-Leffler Function

The Mittag-Leffler function is a direct extension of the exponential function which plays an important role in the solution of fractional differential calculus as the exponential function does in ordinary differential calculus [97]. The simplest form of

the Mittag-Leffler function is the Mittag-Leffler function with one parameter, written as [98]:

$$E_{\alpha}(x) = \sum_{k=0}^{\infty} \frac{x^k}{\Gamma(\alpha k + 1)}, \quad \alpha > 0 \quad (\text{A3.5})$$

The two parameter form of the Mittag-Leffler function are as follows [99]:

$$E_{\alpha,\beta}(x) = \sum_{k=0}^{\infty} \frac{x^k}{\Gamma(\alpha k + \beta)}, \quad \alpha > 0, \beta > 0 \quad (\text{A3.6})$$

If  $\beta = 1$  the Mittag-Leffler function with two parameters, i.e. (A3.6) is similar to the one with one parameter, i.e.:

$$E_{\alpha,1}(x) = E_{\alpha}(x) \quad (\text{A3.7})$$

Some properties of the Mittag-Leffler function can be written as:

$$E_{1,1}(x) = e^x \quad (\text{A3.8})$$

$$E_{1,2}(x) = \frac{e^x - 1}{x} \quad (\text{A3.9})$$

## A3.2 Properties of Fractional-Order Calculus

Just like in integer-order calculus, the Laplace transform can also be applied to fractional-order calculus. The integration property is one useful property when dealing with fractional-order calculus, and is written as [75]:

$$L\left[{}_0D_t^{-\alpha} f(t)\right] = \frac{1}{s^{\alpha}} F(s), \quad \alpha > 0 \quad (\text{A3.10})$$

where  $D^{-\alpha}$  defines fractional integration if  $\alpha$  is negative and 0 is the initial time instant.

The  $\alpha^{\text{th}}$  order integral of a function  $f(t)$  is written as [100]:

$$D^{-\alpha} f(t) = I^{\alpha} f(t) = \frac{t^{\alpha-1}}{\Gamma(\alpha)} * f(t) = \frac{1}{\Gamma(\alpha)} \int_0^t (t-\tau)^{\alpha-1} f(\tau) d\tau, t > 0, \alpha > 0 \quad (\text{A3.11})$$

where  $I^{\alpha}$  is the fractional-order integral operator.

Using convolution, (A3.11) can be rewritten as:

$$I^{\alpha} f(t) = \Phi_{\alpha}(t) * f(t), \alpha > 0 \quad (\text{A3.12})$$

where  $\Phi_{\alpha}(t) = \frac{t^{\alpha-1}}{\Gamma(\alpha)}$

The integration property can also be interpreted as the Laplace transform of a time-domain convolution and is written as [86]:

$$L[{}_0D_t^{-\alpha} f(t)] = L\left[\frac{t^{\alpha-1}}{\Gamma(\alpha)} * f(t)\right] = \frac{1}{s^{\alpha}} F(s), \alpha > 0 \quad (\text{A3.13})$$

The differentiation property is another useful property when dealing with fractional-order calculus and when the initial conditions are zero it is expressed as [75]:

$$L[{}_0D_t^{\alpha} f(t)] = s^{\alpha} F(s), \alpha > 0 \quad (\text{A3.14})$$

When the initial conditions are non-zero, the differentiation property in fractional-order calculus is expressed in a similar way as in integer-order calculus as [101]:

$$L[{}_0D_t^\alpha f(t)] = s^\alpha F(s) - s^{\alpha-1}f(0) - s^{\alpha-2}f'(0) - \dots - s^{\alpha-n}f^{n-1}(0) \quad (\text{A3.15a})$$

$$L[{}_0D_t^\alpha f(t)] = s^\alpha F(s) - \sum_{k=0}^{n-1} s^{\alpha-n-1} f^{(k)}(0) \quad (\text{A3.15b})$$

Fractional differentiation and fractional integration are linear operations [101]:

$${}_0D_t^\alpha (af(t) + bg(t)) = a({}_0D_t^\alpha f(t)) + b({}_0D_t^\alpha g(t)), \quad \forall a, b, f(t), g(t) \quad (\text{A3.16})$$

The additive index law (semigroup property)

$${}_0D_t^\alpha {}_0D_t^\beta f(t) = {}_0D_t^{\beta} {}_0D_t^\alpha f(t) = {}_0D_t^{\beta+\alpha} f(t) \quad (\text{A3.17})$$

holds under zero initial condition.

### A3.3 Definitions of Fractional-Order Calculus

In comparison to the derivative operator for integer-order, i.e.  $D = \frac{d}{dt}$ , the fractional-order derivative operator can be denoted as:

$$D^\alpha = \frac{d^\alpha}{dt^\alpha} \quad (\text{A3.18})$$

where  $\alpha \in \mathbb{R}$  [76] Since analytical solutions of fractional-order differentiations and integrals are complicated, the authors in [102] proposed more simplified methods when dealing with fractional-order calculus. The commonly used definitions to solve fractional-order differentiations and integral are the Riemann-Liouville, Grünwald-Letnikov and Caputo definitions listed below.

The Riemann-Liouville and Grünwald-Letnikov definitions are more suitable to describe fractional-order calculus problems with zero-initial conditions whilst the Caputo definition is useful in discussing systems with nonzero initial conditions [86].

### A3.3.1 The Grünwald-Letnikov Definition

The  $\alpha^{\text{th}}$  order Grünwald-Letnikov (G-L) derivative of a function  $f(t)$  is written as [103] and [8] :

$${}_{t_0}^{GL}D_t^\alpha f(t) = \lim_{h \rightarrow 0} \frac{1}{h^\alpha} \sum_{j=0}^{\lceil (t-t_0)/h \rceil} (-1)^j \binom{\alpha}{j} f(t-jh) \quad (\text{A3.19})$$

where  $\binom{\alpha}{j} = \frac{\Gamma(\alpha+1)}{\Gamma(j+1)\Gamma(\alpha-j+1)}$ , the notation GL indicates “Grünwald-Letnikov definition”, the operator  $D^\alpha$  defines fractional differentiation or integration depending on the sign of  $\alpha$ . If  $\alpha$  is negative then  $D^{-\alpha}$  will define fractional integration, and if  $\alpha$  is positive then  $D^\alpha$  will define fractional differentiation,  $\Gamma(\bullet)$  is the Euler’s Gamma function,  $\lceil \bullet \rceil$  means to round off to the nearest integer,  $t_0$  is the initial time instant and  $h$  is the finite sampling interval.

### A3.3.2 The Riemann-Liouville Definition

The  $\alpha^{\text{th}}$  order Riemann-Liouville (R-L) derivative of a function  $f(t)$  is written as [75] and [8]:

$${}_{t_0}^{RL}D_t^\alpha f(t) = \frac{d^n}{dt^n} \left[ {}_{t_0}^{RL}D_t^{-(n-\alpha)} f(t) \right] \quad (\text{A3.20a})$$

$${}_{t_0}^{RL}D_t^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \left( \int_{t_0}^t \frac{f(\tau)}{(t-\tau)^{\alpha+1-n}} d\tau \right) \quad (\text{A3.20b})$$

where the notation RL indicates “Riemann-Liouville definition”,  $n$  is an integer such that  $n-1 < \alpha < n$  and  $t_0$  is the initial time instant.

The Riemann-Liouville definition first carries out the fractional-order integration of function  $f(t)$  before applying the integer-order derivative to the result. Since in the Riemann-Liouville definition there is fractional-order integration, and this calls for the need to define the Riemann-Liouville fractional-order integration, which according to [103] and [8] is expressed as:

$${}_{t_0}^{RL}D_t^{-\alpha} f(t) = \frac{1}{\Gamma(\alpha)} \int_{t_0}^t \frac{f(\tau)}{(t-\tau)^{1-\alpha}} d\tau \quad (\text{A3.21})$$

where  $0 < \alpha < 1$ .

### A3.3.3 The Caputo Definition

Caputo derivatives are useful in discussing systems with nonzero initial conditions [104]. The  $\alpha^{\text{th}}$  order Caputo (C) derivative of a function  $f(t)$  is written as [75]:

$${}_{t_0}^C D_t^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \int_{t_0}^t \frac{f^{(n)}(\tau)}{(t-\tau)^{\alpha+1-n}} d\tau \quad (\text{A3.22})$$

where the notation C indicates “Caputo definition”. The Caputo definition first carries out the integer-order differentiation of function  $f(t)$  before applying the fractional-order integration to the result. Since in the Caputo definition there is fractional-order integration, and this calls for the need to define the Caputo fractional-order integration, which according to [105] is written as:

$${}_{t_0}^C D_t^{-\alpha} f(t) = \frac{1}{\Gamma(\alpha)} \int_{t_0}^t \frac{f(\tau)}{(t-\tau)^{1-\alpha}} d\tau \quad (\text{A3.23})$$

### A3.4 The Poisson Moment Functional Filtering for Continuous-Time Fractional-Order System Identification

To handle the estimation of  $U_{0(i-1)\alpha,N}$  and  $Y_{0(i-1)\alpha,N}$  the PMF transform of order  $r$  is applied to (3.20), resulting in:

$$\mathbf{P}_t^r \left[ \mathbf{Y}_{0(i-1)\alpha,N} \right] = \mathbf{\Gamma}_i \mathbf{P}_t^r \left[ \mathbf{X}_N \right] + \mathbf{\Omega}_i \mathbf{P}_t^r \left[ \mathbf{U}_{0(i-1)\alpha,N} \right] \quad (\text{A3.24})$$

where  $\mathbf{P}_t^r \left[ D^\alpha g(t) \right]$  is the  $r^{\text{th}}$  order PMF transform of signal  $D^\alpha g(t)$  at time instant  $t$  and is given by the following convolution product [70]:

$$\mathbf{P}_t^r \left[ D^\alpha g(t) \right] = D^\alpha g(t) * p_r(t) = \int_0^t D^\alpha g(\tau) p_r(t-\tau) d\tau \quad (\text{A3.25})$$

where  $p_r(t)$  is the  $r^{\text{th}}$  order Poisson pulse function expressed as:

$$p_r(t) = \frac{\beta^{r+1} t^r e^{-\lambda t}}{r!} \quad (\text{A3.26})$$

where  $r \in \mathbb{N}$ ,  $r \geq i$ , whereby  $i$  is the user-defined index and  $\lambda \in \mathbb{R}^+$  and  $\beta \in \mathbb{R}^+$  are the Poisson filter constant and gain respectively.

The  $r^{\text{th}}$  order PMF of  $D^\alpha g(t)$  can be measured as the output at time instant  $t$  of a cascaded low-pass filter chain of  $(r+1)$  identical stages, each with a transfer function expressed as [70]:

$$P_r(s) = \left( \frac{\beta}{s + \lambda} \right) \quad (\text{A3.27})$$

According to [71], if we consider two  $\alpha^{\text{th}}$  differentiable functions  $g(t)$  and  $f(t)$ , then

$$g(t) * D^\alpha f(t) = L^{-1} [G(s)s^\alpha F(s)] = L^{-1} [s^\alpha G(s)F(s)] = D^\alpha g(t) * f(t) \quad (\text{A3.28})$$

In (A3.28) it can be seen that the  $\alpha^{\text{th}}$  order derivative of  $f(t)$  can be transmitted to the  $\alpha^{\text{th}}$  order derivative of  $g(t)$  and vice-versa. Applying the property of (A3.28) to the  $(i-1)\alpha$ -order fractional derivatives of the input data, the following PMF based filtered input data is obtained:

$$\mathbf{P}_t^r [\mathbf{U}_{0(i-1)\alpha, N}] = \begin{bmatrix} g(t) * \mathbf{u}(t) \\ D^\alpha g(t) * \mathbf{u}(t) \\ \vdots \\ D^{(i-1)\alpha} g(t) * \mathbf{u}(t) \end{bmatrix} \quad (\text{A3.29})$$

where  $f(t)$  in (A3.28) is replaced by  $\mathbf{u}(t)$ .

In (A3.29) the  $\alpha^{\text{th}}$  order derivatives of  $g(t)$  can be easily obtained using the Riemann–Liouville, Grünwald–Letnikov or Caputo definitions. Applying the property of (A3.28) to the  $(i-1)\alpha$ -order fractional derivatives of the output data and the noise, the following PMF based filtered output data and noise is obtained:

$$\mathbf{P}_t^r [\mathbf{Y}_{0(i-1)\alpha, N}] = \begin{bmatrix} g(t) * \mathbf{y}(t) \\ D^\alpha g(t) * \mathbf{y}(t) \\ \vdots \\ D^{(i-1)\alpha} g(t) * \mathbf{y}(t) \end{bmatrix} \quad (\text{A3.30})$$

and

$$\mathbf{P}_t^r [\mathbf{N}_{0(i-1)\alpha, N}] = \begin{bmatrix} g(t) * \mathbf{n}(t) \\ D^\alpha g(t) * \mathbf{n}(t) \\ \vdots \\ D^{(i-1)\alpha} g(t) * \mathbf{n}(t) \end{bmatrix} \quad (\text{A3.31})$$

respectively.

According to [62] the PMF filter order and the PMF filter parameters are chosen as follows:

The PMF filter order,  $r$  must respect the following condition:

$$r \geq i \tag{A3.32}$$

and for the PMF filter parameters, it is generally assumed that:

$$\lambda = \beta \tag{A3.33}$$

This assumption helps to reduce the number of design parameters.

## Chapter 4

# Massive MIMO Channel Equalization Using Integer-Order State-Space Models

### 4.1 Introduction

Channel equalization as a means of mitigating the effects of inter-symbol interference (ISI) caused by the massive multiple-input multiple-output (MIMO) frequency-selective wireless channels is discussed in this chapter. The equalization technique employed in our work takes from works by Zhang and Bitmead [106], Zhang and Bitmead [107] and Al-Dhahir and Sayed [108] where they proposed the use of the minimum mean square error – decision feedback equalizer (MMSE-DFE) to combat the effects of ISI in a multiple-input multiple-output (MIMO) system. The choice of the MMSE-DFE is that it benefits from both the advantages of the MMSE and the DF equalizers.

The input-output block data in the integer-order state-space equalization model is presented just like in the extensively studied FIR equalization model. The length of input-output block data is dependent on the number of feedforward filter taps used in the construction of the equalizer. This similarity in presentation will allow for the design of the integer-order state-space equalization model to be just a modification of the already widely studied FIR equalization model. Unlike in most studies where the feedback filter coefficients are calculated independent of the feedforward filter coefficients, and then using some relationship to find one from the other, in this study we will jointly calculate the feedforward and feedback filter coefficients using a technique outlined in [106]. The performance of channel equalization using state-space modelling will then be compared with the channel equalization for the finite impulse response (FIR) model.

## 4.2 Inter-Symbol Interference

As the signal is transmitted through the communication channel it can suffer from inter-symbol interference (ISI) which is caused by multipath propagation or frequency-selective channels. Frequency selective channels are those channels where the transmitted signals are subjected to variations in amplitude and phase as they propagate through the transmission medium. In frequency selective channels different frequency components of the transmitted signal experience different fading magnitudes. In this the coherence bandwidth of the channel is smaller than the bandwidth of the transmitted signal. But in flat fading channels, the different frequency components of the transmitted signal suffer the same fading magnitude. In this the coherence bandwidth of the channel is larger than the bandwidth of the signal. ISI is when the receiver receives the desired signal and in addition to that it also receives multiple versions of the desired signal with different delays.

Considering Figure 4.1, in an ideal situation the transmitted signal (which is made up of several symbols) doesn't suffer any reflection or refraction and as a result the receiver only receives one version (A) of the transmitted symbols (S1, S2, S3 and S4) [109].

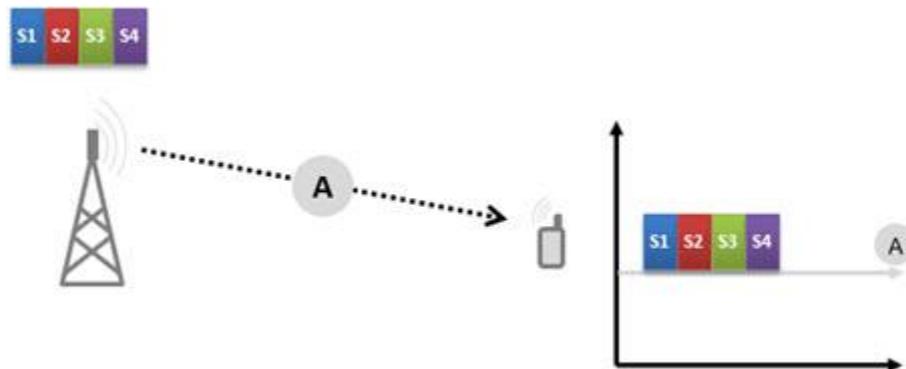


Figure 4. 1: Ideal transmission

But in real life situation the transmitted symbols can be reflected or refracted on trees, building, cars and other different objects resulting in the transmitted symbols reaching the receiver via multiple paths. These multiple paths can be of different lengths, i.e. one

path can take longer time than the other, which results in the different versions of the transmitted symbols arriving at the receiver at different times. These delays in arrival mean that part or all of a given symbol can spread into the subsequent symbols. Figure 4.2 shows the receiver receiving the desired transmitted symbols, A and several versions of the transmitted symbols, B and C [109].

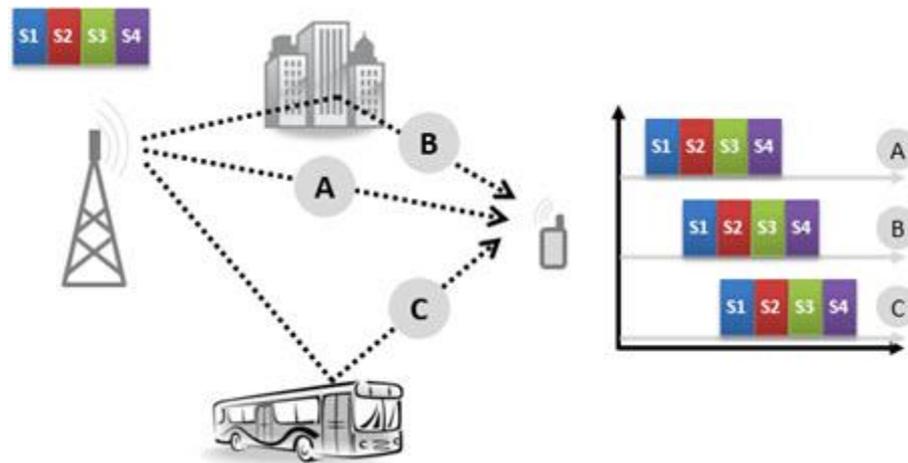


Figure 4. 2: Real transmission

At the receiver end, the different versions of the transmitted symbols, i.e. A, B and C are then summed up resulting in signal 1 as shown in Figure 4.3. Here there is an overlap in the summed-up symbols thus causing ISI. For example in signal 1, symbol S1 is interfered with by symbol S2 and symbol S3 [109].

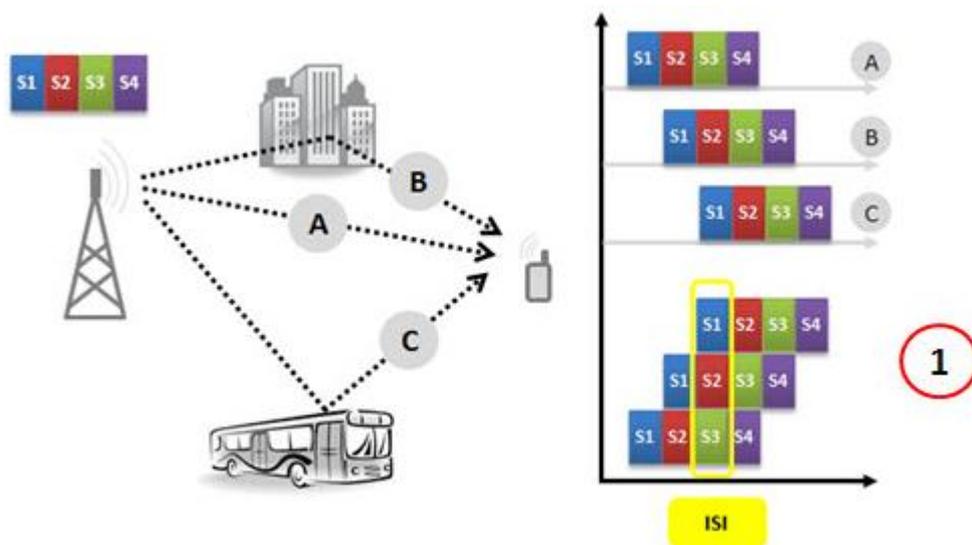


Figure 4. 3: Effects of inter-symbol interference (ISI)

Generally speaking in a system where there are length- $L$  ISI channel paths, the received signal,  $y(t)$  taking ISI into consideration can be written as:

$$y(t) = \underbrace{h_0 u(t)}_{\text{desired signal}} + \underbrace{h_1 u(t-1) + h_2 u(t-2) + \dots + h_L u(t-L)}_{\text{ISI signal}} + \underbrace{n(t)}_{\text{noise}} \quad (4.1)$$

where  $h_0$  is the unreflected or unrefracted channel path,  $h_1, h_2 \dots h_L$  are the different reflected or refracted multiple paths,  $u(t)$  is the transmitted signal,  $u(t-1), u(t-2) \dots u(t-L)$  are the delayed versions of the transmitted signal. The block diagram of a multipath fading channel is shown in Figure 4.4 [110].

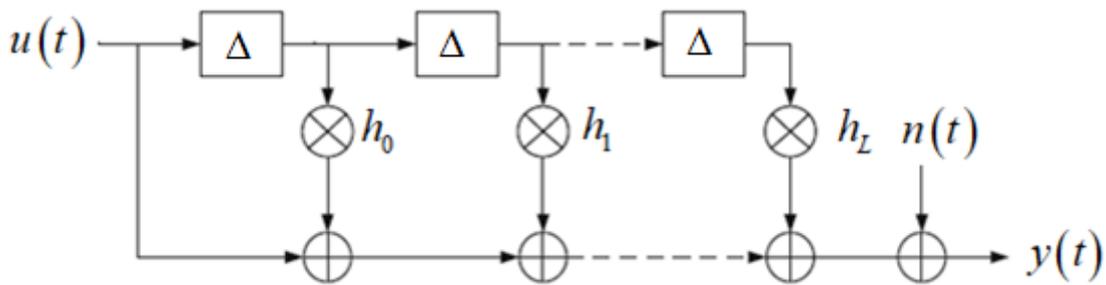


Figure 4. 4: Block diagram of a multipath fading channel

where  $\Delta$  denotes the delay.

### 4.3 Channel Equalization

Equalization is the process of removing the effects of inter-symbol interference (ISI) and noise on the received signal thus allowing for better recovery of the originally transmitted symbols at the receiver end [111]. The idea with equalization is to reduce the mean square error (MSE), i.e. the difference between the desired transmitted symbols and the signal at the output of the equalizer.

Equalizers can be classified into linear and nonlinear equalizers as shown in Figure 4.5. Linear equalizers are simple and easy to implement. The transfer function of

linear equalizers can be thought of as the inverse transfer function of the communication channel. Linear equalizers usually don't have a feedback path to change the subsequent outputs of the equalizer and these may include matched filter (MF), zero forcing (ZF) and minimum mean square error (MMSE) equalizers. Nonlinear equalizers are usually needed when the channel suffers from too much distortion that cannot be easily mitigated by linear equalizers [111]. Nonlinear equalizers usually have a feedback path to change the subsequent outputs of the equalizer and may include decision-feedback (DF) and maximum likelihood sequence estimation (MLSE) equalizers [112]. Equalizers can be further classified into non-adaptive and adaptive equalizers. Non-adaptive equalizers are ideal when the channel is static or time-invariant and adaptive equalizers are ideal when the channel is time varying.

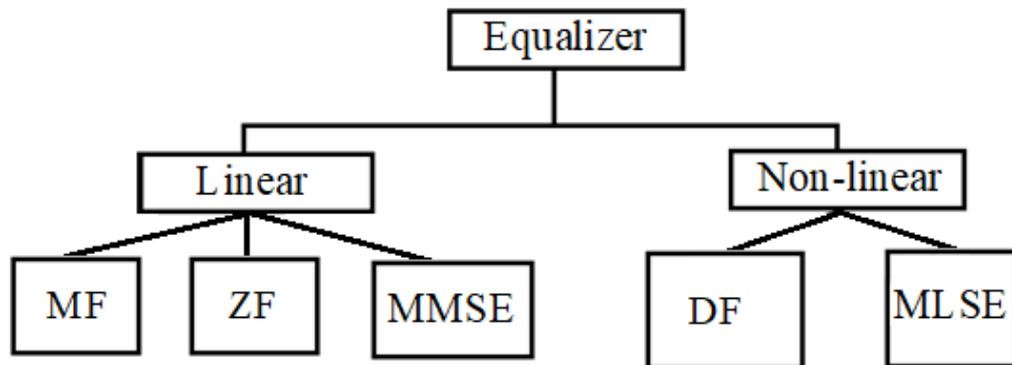


Figure 4. 5: Summary of linear and nonlinear equalizers

We begin our discussion on equalizers by first having a look at linear equalizers whose block diagram is as shown in Figure 4.6 [113], where  $u$  is the transmitted signal,  $H$  is the channel matrix,  $n$  is the additive noise at the receiver,  $y$  is the received signal,  $W$  is the equalizer matrix which is also referred to as the equalizer coefficients,  $\hat{u}$  is the recovered transmitted signal after equalization,  $e$  is the recovered transmitted signal error vector and MSE is the mean square error which is the common measure of quality in communication.

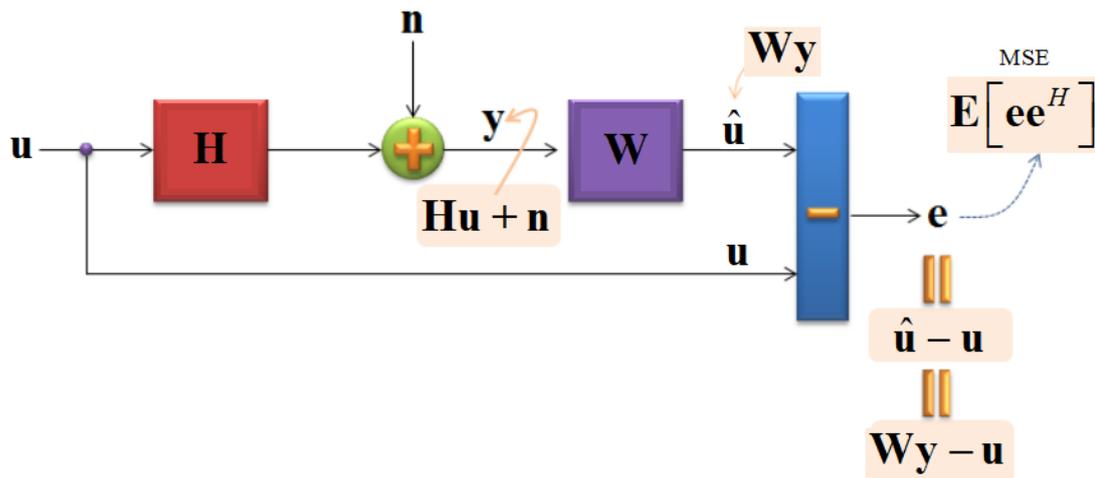


Figure 4. 6: Block diagram for linear equalizers

In analysing the following equalizers a communication system with the following received signal is considered:

$$y = \mathbf{H}\mathbf{u} + n \quad (4.2)$$

Under linear equalization the first equalizer to be discussed is the matched filter (MF) equalizer.

### 4.3.1 Matched Filter Equalizer

Matched filter equalizers maximise the signal-to-noise ratio (SNR) but do not necessarily cancel the inter-symbol interference, i.e. they are optimal with respect to noise, but ignore the ISI. In time domain the matched filter is a time-reversed and conjugated version of the ISI channel [114].

With reference to (4.2), the matched filter equalizer coefficients  $\mathbf{W}_{MF}$  can be written as:

$$\mathbf{W}_{MF} = \mathbf{H}^H \quad (4.3)$$

where  $[\cdot]^H$  is the Hermitian matrix.

Applying the MF equalizer to the received signal (4.2), the recovered transmitted signal after MF equalization,  $\hat{\mathbf{u}}_{MF}$  can then be expressed as:

$$\hat{\mathbf{u}}_{MF} = \mathbf{W}_{MF} \mathbf{y} = \mathbf{H}^H \mathbf{y} = \mathbf{H}^H (\mathbf{H} \mathbf{u} + \mathbf{n}) = \mathbf{H}^H \mathbf{H} \mathbf{u} + \mathbf{H}^H \mathbf{n} \quad (4.4)$$

The recovered transmitted signal error vector after the MF equalizer,  $\mathbf{e}_{MF}$  is given by:

$$\mathbf{e}_{MF} \triangleq \hat{\mathbf{u}}_{MF} - \mathbf{u} = \mathbf{H}^H \mathbf{H} \mathbf{u} + \mathbf{H}^H \mathbf{n} - \mathbf{u} \quad (4.5)$$

At the output of the MF equalizer the signal-to-noise ratio (SNR),  $SNR_{MF}$  can be written as [115] and [116]:

$$SNR_{MF} = \frac{\mathbf{H}^H \mathbf{H} \mathbf{H}^H \mathbf{H}}{\mathbf{H}^H \mathbf{H}} SNR \quad (4.6)$$

The next linear equalizer to discuss is the zero forcing (ZF) equalizer.

### 4.3.2 Zero Forcing Equalizer

Zero forcing equalizers minimise the ISI but at the cost of probably enhancing the noise, i.e. it is optimal with respect to ISI but ignores the noise. The zero forcing equalizer is the pseudo inverse of the channel matrix,  $\mathbf{H}$  in (4.2) and the zero forcing equalizer coefficients  $\mathbf{W}_{ZF}$  can be expressed as [117]:

$$\mathbf{W}_{ZF} = \mathbf{H}^\dagger = (\mathbf{H}^H \mathbf{H})^{-1} \mathbf{H}^H \quad (4.7)$$

Applying the ZF equalizer to the received signal (4.2), the recovered transmitted signal after ZF equalization,  $\hat{\mathbf{u}}_{ZF}$  can then be expressed as:

$$\hat{\mathbf{u}}_{ZF} = \mathbf{W}_{ZF} \mathbf{y} = \mathbf{H}^\dagger \mathbf{y} = (\mathbf{H}^H \mathbf{H})^{-1} \mathbf{H}^H \mathbf{y} = \mathbf{u} + (\mathbf{H}^H \mathbf{H})^{-1} \mathbf{H}^H \mathbf{n} \quad (4.8)$$

The recovered transmitted signal error vector after the ZF equalizer,  $\mathbf{e}_{ZF}$  is given by:

$$\mathbf{e}_{ZF} \triangleq \hat{\mathbf{u}}_{ZF} - \mathbf{u} = (\mathbf{H}^H \mathbf{H})^{-1} \mathbf{H}^H \mathbf{n} \quad (4.9)$$

According to [117] the SNR at the output of the ZF equalizer for data stream  $k$ ,  $SNR_{ZF,k}$  can be written as:

$$SNR_{ZF,k} = \frac{1}{\left[ (\mathbf{H}^H \mathbf{H})^{-1} \right]_{k,k}} SNR \quad (4.10)$$

where  $k$  ranges from one to total number of transmit antennas, i.e.  $k = 1, 2, \dots, m$ , and  $[\cdot]_{k,k}$  denotes the  $k^{\text{th}}$  diagonal element.

The last linear equalizer to consider is the minimum mean square error (MMSE) equalizer.

### 4.3.3 Minimum Mean Square Error Equalizer

The MMSE equalizer does not completely eliminate the ISI but instead minimises the ISI and the noise power at the output of the equalizer [118]. MMSE equalizers provide a trade-off between ISI cancellation and probable noise enhancement, i.e. it optimally weighs both the ISI and noise.

The MMSE equalizer minimises the MSE (mean square error) of the received signal. Since the MSE is the common measure of the quality, a lower value of MSE means a good equalizer and a high value of MSE implies the equalizer is not so good. In this case the MSE is given as:

$$MSE = E \left[ \|\mathbf{u} - \hat{\mathbf{u}}_{MMSE}\|^2 \right] = E \left[ \mathbf{e} \mathbf{e}^H \right] \quad (4.11)$$

where  $\mathbf{e} = \mathbf{u} - \hat{\mathbf{u}}_{MMSE}$  is the estimation error vector and  $\hat{\mathbf{u}}_{MMSE}$  is the MMSE estimate of  $\mathbf{u}$  or the recovered transmitted signal after MMSE equalization whose expression is given in (4.13) below.

In MMSE equalization the MMSE equalizer coefficients  $\mathbf{W}_{MMSE}$  are chosen to satisfy the following minimization problem:

$$\begin{aligned} \mathbf{W}_{MMSE} &= \arg \min MSE \\ \mathbf{W}_{MMSE} &= \arg \min E \left[ \|\mathbf{u} - \hat{\mathbf{u}}_{MMSE}\|^2 \right] \\ \mathbf{W}_{MMSE} &= \arg \min E \left[ \mathbf{e} \mathbf{e}^H \right] \end{aligned} \quad (4.12)$$

The recovered transmitted signal after the MMSE equalizer,  $\hat{\mathbf{u}}_{MMSE}$  can be expressed as:

$$\hat{\mathbf{u}}_{MMSE} = \mathbf{W}_{MMSE} \mathbf{y} \quad (4.13)$$

Using the orthogonality principle which states that  $E[\mathbf{e} \mathbf{y}^H] = \mathbf{0}$  [110] the MMSE equalizer coefficients  $\mathbf{W}_{MMSE}$  can be derived as follows:

$$E[\mathbf{e} \mathbf{y}^H] = E[(\mathbf{u} - \hat{\mathbf{u}}_{MMSE}) \mathbf{y}^H] = \mathbf{0} \quad (4.14a)$$

$$E[(\mathbf{u} - \mathbf{W}_{MMSE} \mathbf{y}) \mathbf{y}^H] = \mathbf{0} \quad (4.14b)$$

$$E[\mathbf{u} \mathbf{y}^H - \mathbf{W}_{MMSE} \mathbf{y} \mathbf{y}^H] = \mathbf{0} \quad (4.14c)$$

$$\underbrace{E[\mathbf{u} \mathbf{y}^H]}_{\mathbf{R}_{uy}} - \mathbf{W}_{MMSE} \underbrace{E[\mathbf{y} \mathbf{y}^H]}_{\mathbf{R}_{yy}} = \mathbf{0} \quad (4.14d)$$

$$\mathbf{R}_{uy} - \mathbf{W}_{MMSE} \mathbf{R}_{yy} = \mathbf{0} \quad (4.14e)$$

where  $\mathbf{R}_{uy}$  is the cross-correlation matrix of  $\mathbf{u}$  and  $\mathbf{y}$ , and  $\mathbf{R}_{yy}$  is the auto-correlation matrix of  $\mathbf{y}$ .

Using (4.14e), the MMSE equalizer coefficients  $\mathbf{W}_{MMSE}$  can then be written as:

$$\mathbf{W}_{MMSE} = \mathbf{R}_{uy} \mathbf{R}_{yy}^{-1} \quad (4.15)$$

$\mathbf{R}_{uy}$  in (4.15) can be expanded as follows:

$$\mathbf{R}_{uy} = E[\mathbf{u}\mathbf{y}^H] = E[\mathbf{u}(\mathbf{H}\mathbf{u} + \mathbf{n})^H] = E[\mathbf{u}\mathbf{u}^H \mathbf{H}^H + \mathbf{u}\mathbf{n}^H] = E[\mathbf{u}\mathbf{u}^H \mathbf{H}^H] + \underbrace{E[\mathbf{u}\mathbf{n}^H]}_0 \quad (4.16a)$$

$$\mathbf{R}_{uy} = E[\mathbf{u}\mathbf{u}^H \mathbf{H}^H] = \mathbf{R}_{uu} \mathbf{H}^H \quad (4.16b)$$

where  $\mathbf{R}_{uu}$  is the auto-correlation matrix of  $\mathbf{u}$  and is written as:

$$\mathbf{R}_{uu} = E_s \mathbf{I} \quad (4.17)$$

where  $E_s$  is the power of the transmitted signal and  $\mathbf{I}$  is the identity matrix.

In turn  $\mathbf{R}_{yy}$  in (4.15) can be expanded as follows:

$$\mathbf{R}_{yy} = E[\mathbf{y}\mathbf{y}^H] = E[(\mathbf{H}\mathbf{u} + \mathbf{n})(\mathbf{H}\mathbf{u} + \mathbf{n})^H] = E[\mathbf{H}\mathbf{u}\mathbf{u}^H \mathbf{H}^H + \mathbf{H}\mathbf{u}\mathbf{n}^H + \mathbf{n}\mathbf{u}^H \mathbf{H}^H + \mathbf{n}\mathbf{n}^H] \quad (4.18a)$$

$$\mathbf{R}_{yy} = E[\mathbf{H}\mathbf{u}\mathbf{u}^H \mathbf{H}^H] + \underbrace{E[\mathbf{H}\mathbf{u}\mathbf{n}^H]}_0 + \underbrace{E[\mathbf{n}\mathbf{u}^H \mathbf{H}^H]}_0 + E[\mathbf{n}\mathbf{n}^H] \quad (4.18b)$$

$$\mathbf{R}_{yy} = \mathbf{H}\mathbf{R}_{uu}\mathbf{H}^H + \mathbf{R}_m \quad (4.18c)$$

where  $\mathbf{R}_m$  is the auto-correlation matrix of  $\mathbf{n}$  and is written as:

$$\mathbf{R}_m = N_0\mathbf{I} \quad (4.19)$$

where  $N_0$  is the noise power at the receiver end and  $\mathbf{I}$  is the identity matrix.

Substituting (4.16b) and (4.18c) into (4.15), the MMSE equalizer coefficients  $\mathbf{W}_{MMSE}$  can be rewritten as:

$$\mathbf{W}_{MMSE} = \mathbf{R}_{uy}\mathbf{R}_{yy}^{-1} = \left( \mathbf{H}^H\mathbf{H} + \frac{1}{SNR}\mathbf{I} \right)^{-1} \mathbf{H}^H \quad (4.20)$$

According to [117] the SNR at the output of the MMSE equalizer for data stream  $k$ ,  $SNR_{MMSE,k}$  can be written as:

$$SNR_{MMSE,k} = \frac{SNR}{\left( \mathbf{H}^H\mathbf{H} + \frac{1}{SNR}\mathbf{I} \right)^{-1}_{k,k}} - 1 \quad (4.21)$$

where  $k$  ranges from one to total number of transmit antennas, i.e.  $k = 1, 2, \dots, m$  and  $[\cdot]_{k,k}$  denotes the  $k^{\text{th}}$  diagonal element. The MMSE equalizer is a better performing linear equalizer [114], but its performance is affected when the channels suffer from severe ISI. To address this shortcoming of the linear equalizers the discussion on equalizers is then extended to nonlinear equalizers namely the decision feedback equalizers (DFE).

### 4.3.4. Decision Feedback Equalizer

In linear equalization the linear equalizers (filters) may introduce additional noise variance to the output signal thus leading to degraded system performance. This drawback of linear equalizers can be avoided by using the decision feedback equalizer (DFE). The DFE consists of feedforward and feedback filters [119]. The feedforward filter reduces the effect of ISI from future symbols, i.e. the feedforward filter is similar to the linear equalizers previously discussed whilst the feedback filter reduces the effect of ISI from past symbols [120]. The effect of the previous symbol on the current symbol is called post-cursor ISI and the effect of the next symbol on the current symbol is called pre-cursor ISI. Since the DFE can only estimate the post-cursors, typically it needs to be used in combination with a linear equalizer [121]. The decision feedback equalizers have noise reduction as compared to the linear equalizers which suffer from noise enhancement [122]. The disadvantages of decision feedback equalizers are their complexity compared to linear equalizers and the error propagation in the feedback loop, especially if the detection has not been done correctly. To avoid the problem of error propagation, it is assumed that no erroneous decisions are passed into the feedback filter.

Just like in the case of linear equalizers, a communication system with received signal (4.2) is considered when analysing the DFE. The DFE block diagram is as shown in Figure 4.7 [123].

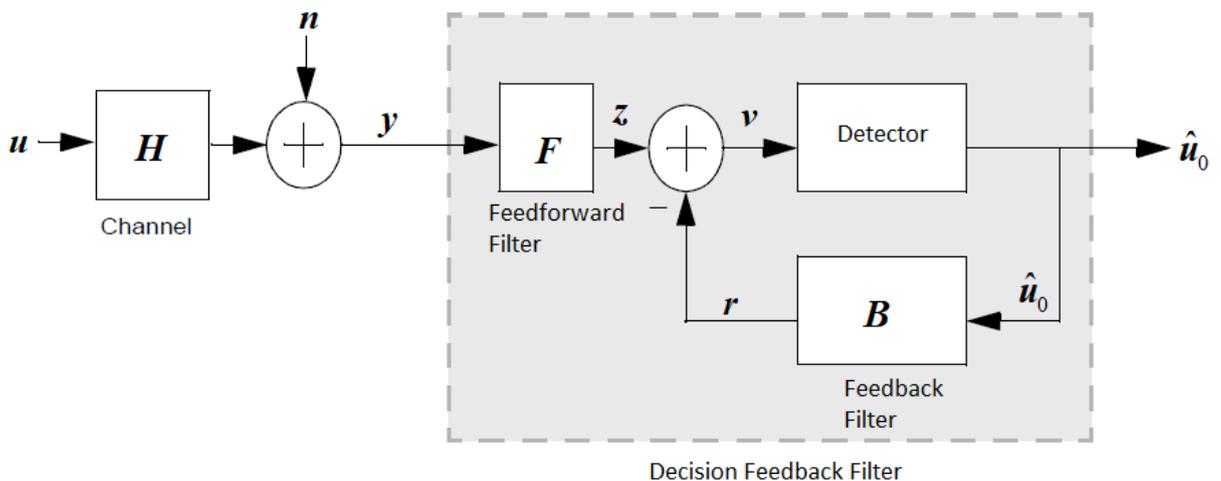


Figure 4. 7: Block diagram for DFE

The DFE uses the already detected or estimated symbols to minimise their interference from future symbols [124]. That is, the interference on the current symbol that was caused by previous symbols is subtracted. The received signal  $\mathbf{y}$  is passed through a feedforward filter,  $\mathbf{F}$ . The role of the feedforward filter is to minimise the effects of the ISI and also of the noise from future symbols [125]. The signal is then passed through a detector to obtain an estimate of the originally transmitted signal,  $\hat{\mathbf{u}}_0$ . The detector chooses symbol,  $s_i$  from the set of  $S$  possible symbols  $\{s_i\}$ ,  $i = 0, \dots, S - 1$  transmitted over the channel such that the error between the chosen symbol and the transmitted symbol is minimised [126]. The nonlinearity of the DFE arises from the inclusion of the detector in the feedback loop which has nonlinear characteristics [127]. Detectors may include the least squares (LS) detector, minimum mean square error (MMSE) detector, maximum likelihood (ML) detector, maximum a posteriori (MAP) detector, etc. After detection the detected symbols are fed back to the DFE through the feedback filter,  $\mathbf{B}$ . The role of the feedback filter is to minimise ISI that arises from the previously detected symbols [125]. The fed back symbols are then subtracted from the incoming symbols to help minimise the effects of the previously detected symbols from the incoming signal. DFE is based on the principle that once you have estimated the current transmitted symbol, you can exactly remove the ISI contribution of that symbol to future received symbols.

Considering the DFE, the communication over the effective channel can be expressed as [128]:

$$\begin{aligned}
 \underbrace{\begin{bmatrix} \mathbf{y}(t) \\ \mathbf{y}(t-1) \\ \vdots \\ \mathbf{y}(t-T+1) \end{bmatrix}}_{\mathbf{y}(t)} &= \underbrace{\begin{bmatrix} \mathbf{H}_0 & \mathbf{H}_1 & \cdots & \mathbf{H}_{L-1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_0 & \mathbf{H}_1 & \cdots & \mathbf{H}_{L-1} & \mathbf{0} & \cdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{H}_0 & \mathbf{H}_1 & \cdots & \mathbf{H}_{L-1} \end{bmatrix}}_{\mathcal{Q}} \underbrace{\begin{bmatrix} \mathbf{u}(t) \\ \mathbf{u}(t-1) \\ \vdots \\ \mathbf{u}(t-T-2L+1) \end{bmatrix}}_{\mathbf{u}(t)} \\
 &+ \underbrace{\begin{bmatrix} \mathbf{n}(t) \\ \mathbf{n}(t-1) \\ \vdots \\ \mathbf{n}(t-T+1) \end{bmatrix}}_{\mathbf{N}(t)}
 \end{aligned} \tag{4.22}$$

which can be written more compactly as:

$$\mathbf{Y}(t) = \mathbf{Q}\mathbf{U}(t) + \mathbf{N}(t) \quad (4.23)$$

where  $T$  is the feedforward filter length and  $L$  the feedback filter length.

In the DFE, the feedforward filter,  $\mathbf{F}$  and feedback filter,  $\mathbf{B}$  coefficients need to be calculated with respect to the channel matrix  $\mathbf{Q}$  [129]].

The signal at the output of the feedforward filter,  $\mathbf{z}$  is expressed as:

$$\mathbf{z} = \mathbf{F}^H \mathbf{y} \quad (4.23)$$

where  $\mathbf{F}$  represents the feedforward filter coefficients.

The signal at the output of the feedback filter,  $\mathbf{r}$  is expressed as:

$$\mathbf{r} = \mathbf{B}^H \hat{\mathbf{u}}_0 \quad (4.24)$$

where  $\mathbf{B}$  represents the feedback filter coefficients and  $\hat{\mathbf{u}}_0$  the previously detected symbol.

The signal from the output of the feedback filter,  $\mathbf{r}$  is then subtracted from the incoming signal at the output of the feedforward filter,  $\mathbf{z}$  to obtain the signal at the output of the DFE,  $\mathbf{v}$  as:

$$\mathbf{v} = \mathbf{z} - \mathbf{r} = \mathbf{F}^H \mathbf{y} - \mathbf{B}^H \hat{\mathbf{u}}_0 \quad (4.25)$$

This signal is a linear combination of the current symbols and the past symbols of the received signal,  $\mathbf{y}$  and previous detected symbols of  $\hat{\mathbf{u}}_0$  [127]. Signal  $\mathbf{v}$  is then passed through the detector to obtain an estimate of the original transmitted signal. Assuming

correct detection of previous symbols, the estimation error vector between the symbols at the output of the detector and the transmitted symbols is written as:

$$\mathbf{e} = \mathbf{u}_0 - \mathbf{v} \quad (4.26)$$

$$\mathbf{e} = \mathbf{B}^H \mathbf{u}_0 - \mathbf{F}^H \mathbf{y} \quad (4.27)$$

where  $\mathbf{u}_0$  is the previously transmitted symbol.

The goal of decision feedback equalizers is to optimise the feedforward,  $\mathbf{F}$  and feedback,  $\mathbf{B}$  filter coefficients so that the MSE of the equalizer output is minimised. Given that the MMSE equalizer is a better performing linear equalizer, it is then chosen as the linear equalizer to use for the feedforward filter in the DFE and the resulting equalizer is termed the minimum mean square error – decision feedback equalizer (MMSE-DFE).

According to [106] the arithmetic SNR at the input of the detector for the MMSE-DFE is:

$$ASNR_{MMSE-DFE} = \frac{tr(\mathbf{R}_{uu})}{tr(\mathbf{R}_{ee,\min})} \quad (4.28)$$

where  $tr(\bullet)$  denotes the trace and the input auto-correlation matrix,  $\mathbf{R}_{uu}$  and the error auto-correlation matrix,  $\mathbf{R}_{ee,\min}$  are defined as:

$$\mathbf{R}_{uu} = E[\mathbf{u}\mathbf{u}^H] \quad (4.29)$$

and

$$\mathbf{R}_{ee,\min} = E[\mathbf{e}\mathbf{e}^H] \quad (4.30)$$

respectively

### 4.3.5 Maximum Likelihood Sequence Estimation Equalizer

For the sake of completeness, a brief overview of the maximum likelihood sequence estimation (MLSE) equalizers is presented. In the previous equalization methods the equalizers at the receiver end first attempted to remove the effects of inter-symbol interference (ISI) and/or noise before making a decision on the transmitted signal on a symbol-by-symbol basis. But in the case of MLSE equalization the receiver attempts to recover the entire transmitted sequence of symbols [130]. Due to this MLSE equalizers suffer from high computational complexity if the transmitted sequence is long. The MLSE equalizer compares the received noisy sequence with all possible multi-path noise-free received sequences and selects the closest one.

Again, considering a communication system with received signal (4.2), the received noise-free signal,  $\mathbf{y}_{NF}$  can be expressed as:

$$\mathbf{y}_{NF} = \mathbf{H}\mathbf{u} \quad (4.31)$$

In MLSE equalization the recovered transmitted sequence,  $\hat{\mathbf{u}}_{MLSE}$  is chosen to satisfy the following minimization problem:

$$\hat{\mathbf{u}}_{MLSE} = \arg \min \sum |\mathbf{y} - \mathbf{y}_{NF}|^2 \quad (4.32)$$

This section presented a general discussion on the different linear and nonlinear equalizers. The next section focuses on the MMSE-DFE as the equalizer of choice in addressing the shortcomings associated with ISI at the receiver end. This equalizer benefits from the advantages of the MMSE and the DF equalizers. The discussion is also extended to how the feedforward,  $\mathbf{F}$  and feedback,  $\mathbf{B}$  filter coefficients are calculated in the MMSE-DFE and how this equalizer can be implemented in massive MIMO systems using state-space models. Firstly, the discussion focuses on the MMSE-DFE for the finite impulse response (FIR) massive MIMO channel model.

## 4.4 Minimum Mean Square Error – Decision Feedback Equalizer for the Finite Impulse Response Massive MIMO Channel Model

A massive MIMO wireless system with a base station equipped with  $m$  transmitting antenna elements, and a terminal station equipped with  $p$  receiving antenna elements having length- $L$  ISI channel paths is considered:

$$\mathbf{y}(t) = \sum_{l=0}^{L-1} \mathbf{H}_l \mathbf{u}(t-l) + \mathbf{n}(t) \quad (2.1)$$

where  $\mathbf{y}(t)$  is the  $p \times 1$  received signal vector,  $\mathbf{u}(t-l)$  is the  $m \times 1$  transmitted symbols vector at time  $(t-l)$ ,  $\mathbf{n}(t) \sim CN(0, N_0 \mathbf{I}_p)$  is the  $p \times 1$  additive white Gaussian noise vector at the receiver side at time  $t$ , with  $N_0$  being the noise power and  $\mathbf{I}_p$  is the  $p \times p$  identity matrix and  $\mathbf{H}_l$  is the  $p \times m$   $l^{\text{th}}$  path massive MIMO channel matrix coefficients.

Taking a length  $N_f$  block of input-output data into consideration and assuming that the channel is time-invariant over this block, (2.1) can be expressed as [106]:

$$\underbrace{\begin{bmatrix} \mathbf{y}(t+\Delta) \\ \mathbf{y}(t+\Delta-1) \\ \vdots \\ \mathbf{y}(t+\Delta-N_f+1) \end{bmatrix}}_{\mathbf{Y}_{FIR}(t)} = \underbrace{\begin{bmatrix} \mathbf{H}_0 & \mathbf{H}_1 & \cdots & \mathbf{H}_{L-1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_0 & \mathbf{H}_1 & \cdots & \mathbf{H}_{L-1} & \mathbf{0} & \cdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{H}_0 & \mathbf{H}_1 & \cdots & \mathbf{H}_{L-1} \end{bmatrix}}_{\mathbf{H}_{FIR}} \underbrace{\begin{bmatrix} \mathbf{u}(t+\Delta) \\ \mathbf{u}(t+\Delta-1) \\ \vdots \\ \mathbf{u}(t+\Delta-N_f-L+2) \end{bmatrix}}_{\mathbf{U}_{FIR}(t)} + \underbrace{\begin{bmatrix} \mathbf{n}(t+\Delta) \\ \mathbf{n}(t+\Delta-1) \\ \vdots \\ \mathbf{n}(t+\Delta-N_f+1) \end{bmatrix}}_{\mathbf{N}_{FIR}(t)} \quad (4.33)$$

where  $N_f$  is the length of the feedforward filter matrix or the number of feedforward filter taps and  $\Delta$  is the equalizer's decision delay. The decision delay helps in determining which symbol is detected at the current time,  $t$ .

A more compact representation of (4.33) is written as:

$$\mathbf{Y}_{FIR}(t) = \mathbf{H}_{FIR} \mathbf{U}_{FIR}(t) + \mathbf{N}_{FIR}(t) \quad (4.34)$$

where  $\mathbf{Y}_{FIR}(t)$  is the effective FIR output vector,  $\mathbf{U}_{FIR}(t)$  is the effective FIR input vector,  $\mathbf{N}_{FIR}(t)$  is the effective noise vector and  $\mathbf{H}_{FIR}$  is the effective FIR channel matrix.

The block diagram for the DFE for the FIR channel model is shown in Figure 4.8 [107]

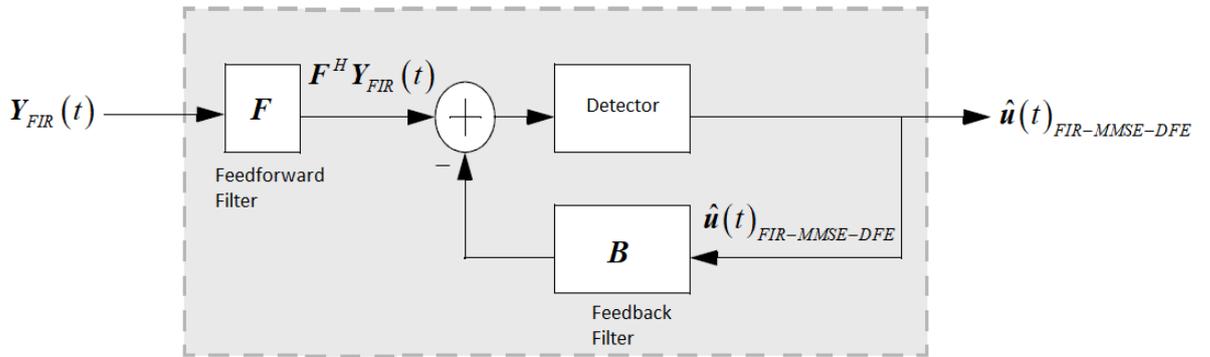


Figure 4. 8: DFE for FIR channel models

In MMSE-DFE the matrix of the feedforward filter coefficients,  $\mathbf{F}^H$  is expressed as:

$$\mathbf{F}^H = [\mathbf{F}_0^H, \mathbf{F}_1^H, \dots, \mathbf{F}_{N_f-1}^H] \quad (4.35)$$

with  $N_f$  matrix taps  $\mathbf{F}_i^H$ , each of size  $p \times m$ , where  $N_f$  is the length of the feedforward filter matrix or the number of feedforward filter taps, and the matrix of the feedback filter coefficients,  $\mathbf{B}^H$  is expressed as:

$$\mathbf{B}^H = [\mathbf{B}_0^H, \mathbf{B}_1^H, \dots, \mathbf{B}_{N_b-1}^H] \quad (4.36)$$

with  $N_b$  matrix taps  $\mathbf{B}_i^H$ , each of size  $m \times m$ , where  $N_b$  is the length of the feedback filter matrix or the number of feedback filter taps.

In the MMSE-DFE when  $N_b = 0$ , i.e. no feedback filters, the MMSE-DFE can be treated as the MMSE linear equalizer.

In most cases the feedback filter coefficients,  $\mathbf{B}$  are calculated independent of the feedforward filter coefficients  $\mathbf{F}$ , and using the relationship  $\mathbf{F} = \mathbf{B}\mathbf{R}_{uy}\mathbf{R}_{yy}^{-1}$  the feedforward filter coefficients can then be obtained from the feedback filter coefficients, where  $\mathbf{R}_{uy}$  is the cross-correlation matrix of  $\mathbf{u}(t)$  and  $\mathbf{y}(t)$ , and  $\mathbf{R}_{yy}$  is the auto-correlation matrix of  $\mathbf{y}(t)$ . But according to [131] and [106], the feedforward and feedback filter coefficient matrices can be jointly calculated by defining a new matrix of combined filter coefficients,  $\mathbf{W}$  as:

$$\mathbf{W} \triangleq \begin{bmatrix} \mathbf{F} \\ \mathbf{B} \end{bmatrix} \quad (4.37)$$

where  $\mathbf{F}$  are the feedforward filter coefficients and  $\mathbf{B}$  are the feedback filter coefficients.

A term  $\mathbf{Z}(t)$  which is a combination of the effective output vector  $\mathbf{Y}_{FIR}(t)$  and the previously detected symbols,  $\mathbf{D}(t)$  can also be defined as:

$$\mathbf{Z}(t) \triangleq \begin{bmatrix} \mathbf{Y}_{FIR}(t) \\ \mathbf{D}(t) \end{bmatrix} \quad (4.38)$$

where

$$\mathbf{D}(t) \triangleq \begin{bmatrix} \mathbf{u}(t) \\ \mathbf{u}(t-1) \\ \vdots \\ \mathbf{u}(t-N_b+1) \end{bmatrix} \quad (4.39)$$

is a vector of previously detected symbols, and assuming correct detection, these are the same as the transmitted symbols and  $N_b$  is the length of the feedback filter.

Matrix  $\mathbf{Z}(t)$  is important in the joint calculation of the feedforward and feedback filter coefficients.

The matrix of the combined filter coefficients,  $\mathbf{W}$  when optimised using the criterion defined in [132] becomes:

$$\mathbf{W}_{MMSE} = \arg \min_{F,B} E \left[ \left| U(t) - \mathbf{WZ}(t) \right|^2 \right] \quad (4.40)$$

which simplifies to:

$$\mathbf{W}_{MMSE} = \arg \min_{F,B} E \left[ \left| \mathbf{u}(t) - \hat{\mathbf{u}}(t)_{FIR-MMSE-DFE} \right|^2 \right] \quad (4.41)$$

where  $\hat{\mathbf{u}}(t)_{SS-MMSE-DFE}$  is the equalized signal.

The auto-correlation matrix,  $\mathbf{R}_{ZZ}$  of  $\mathbf{Z}(t)$  is defined as:

$$\mathbf{R}_{ZZ} = E \left[ \mathbf{Z}(t) \mathbf{Z}^H(t) \right] \quad (4.42)$$

Substituting (4.37) into (4.39) we yields:

$$\mathbf{R}_{ZZ} = \begin{bmatrix} E[\mathbf{Y}_{FIR}(t)\mathbf{Y}_{FIR}^H(t)] & E[\mathbf{Y}_{FIR}(t)\mathbf{D}^H(t)] \\ E[\mathbf{D}(t)\mathbf{Y}_{FIR}^H(t)] & E[\mathbf{D}(t)\mathbf{D}^H(t)] \end{bmatrix} \quad (4.43)$$

Using (4.33) in (4.40) results in:

$$\mathbf{R}_{ZZ} = \begin{bmatrix} \frac{P_s}{m} \mathbf{H}\mathbf{H}^H + N_0 \mathbf{I} & \mathbf{H}E[\mathbf{U}_{FIR}(t)\mathbf{D}^H(t)] \\ E[\mathbf{D}(t)\mathbf{U}_{FIR}^H(t)]\mathbf{H}^H & \frac{P_s}{m} \mathbf{I} \end{bmatrix} \quad (4.44)$$

where  $P_s$  is the total transmit power which is assumed to be evenly distributed between all the  $m$  transmit antenna elements.

The cross-correlation matrix,  $\mathbf{R}_{Zu}$  of  $\mathbf{Z}(t)$  and  $\mathbf{u}(t)$  is defined as:

$$\mathbf{R}_{Zu} = E[\mathbf{Z}(t)\mathbf{u}^H(t)] \quad (4.45)$$

Using (4.38) and then (4.34) in (4.45) results in:

$$\mathbf{R}_{Zu} = E \begin{bmatrix} \mathbf{H}E[\mathbf{U}_{FIR}(t)\mathbf{u}^H(t)] \\ \mathbf{0} \end{bmatrix} \quad (4.46)$$

According to [106], using (4.44) and (4.46) the MMSE solution of the MMSE-DFE can be rewritten as:

$$\mathbf{W}_{MMSE} = \mathbf{R}_{ZZ}^{-1} \mathbf{R}_{Zu} \quad (4.47)$$

These are the combined filter coefficients that minimise the mean square error.

Using (4.47) and (4.38), the estimate of the transmitted symbols,  $\hat{\mathbf{u}}(t)_{FIR-MMSE-DFE}$  at the output of the MMSE-DFE having considered FIR channel modelling can be expressed as:

$$\hat{\mathbf{u}}(t)_{FIR-MMSE-DFE} = \mathbf{W}_{MMSE}^H \mathbf{Y}_{FIR}(t) \quad (4.48)$$

Owing to the fact that the state-space models give good performance results when it comes to system identification. The idea is to then extend the discussion on identification to channel equalization using state-space models. The discussion on the MMSE-DFE for FIR massive MIMO channel model is then extended to the MMSE-DFE for the integer-order state-space massive MIMO channel model.

## 4.5 Minimum Mean Square Error – Decision Feedback Equalizer for the Integer-Order State-Space Massive MIMO Channel Model

Using continuous-time integer-order state-space modelling, the dynamics of the linear time invariant length- $L$  ISI channel massive MIMO system can be expressed as:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad (2.62a)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) + \mathbf{n}(t) \quad (2.62b)$$

where  $\mathbf{x}(t)$  is the  $n \times 1$  state vector,  $\dot{\mathbf{x}}(t)$  is the time derivative of  $\mathbf{x}(t)$ ,  $\mathbf{u}(t)$  is the  $m \times 1$  input vector,  $\mathbf{y}(t)$  is the  $p \times 1$  output vector,  $\mathbf{n}(t)$  is the  $p \times 1$  additive white Gaussian noise vector,  $\mathbf{A}$  is the  $n \times n$  system matrix and it describes the dynamics of the system, i.e. the eigenvalues of the system,  $\mathbf{B}$  is the  $n \times m$  input matrix and it describes the linear transformation by which the inputs influence the next state,  $\mathbf{C}$  is the  $p \times n$  output matrix, and it describes how the state is transferred to the output,  $\mathbf{D}$  is

the  $p \times m$  feed-forward matrix, and in continuous-time systems it is usually  $\mathbf{0}_{p \times m}$ ,  $\dot{\mathbf{x}}(t)$  is the derivative of  $\mathbf{x}(t)$

Having obtained matrices  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$  and  $\mathbf{D}$  using the continuous-time integer-order MOESP algorithm, these matrices can then be used for massive MIMO channel equalization in the context of integer-order state-space channel modelling. Following the equalization procedure outline in [106] which uses the MMSE-DFE to recover the transmitted signal, a length  $N_f$  block of the input-output data, i.e. the blockwise data model for (2.62) can be expressed as:

$$\begin{aligned}
& \underbrace{\begin{bmatrix} \mathbf{P}_t^r \left[ \frac{d^\Delta \mathbf{y}(t)}{dt^\Delta} \right] \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-1)} \mathbf{y}(t)}{dt^{(\Delta-1)}} \right] \\ \vdots \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-N_f+1)} \mathbf{y}(t)}{dt^{(\Delta-N_f+1)}} \right] \end{bmatrix}}_{\mathbf{Y}_{SS}(t)} = \underbrace{\begin{bmatrix} \mathbf{CA}^{N_f-1} \\ \vdots \\ \mathbf{CA} \\ \mathbf{C} \end{bmatrix}}_{\mathbf{r}} \mathbf{x}(t + \Delta - N_f + 1) \\
& + \underbrace{\begin{bmatrix} \mathbf{D} & \mathbf{CB} & \mathbf{CAB} & \dots & \mathbf{CA}^{N_f-2} \mathbf{B} \\ \mathbf{0} & \mathbf{D} & \mathbf{CB} & \dots & \mathbf{CA}^{N_f-3} \mathbf{B} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{D} & \mathbf{CB} \\ \mathbf{0} & \dots & \dots & \mathbf{0} & \mathbf{D} \end{bmatrix}}_{\mathbf{\Omega}} \underbrace{\begin{bmatrix} \mathbf{P}_t^r \left[ \frac{d^\Delta \mathbf{u}(t)}{dt^\Delta} \right] \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-1)} \mathbf{u}(t)}{dt^{(\Delta-1)}} \right] \\ \vdots \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-N_f+1)} \mathbf{u}(t)}{dt^{(\Delta-N_f+1)}} \right] \end{bmatrix}}_{\mathbf{U}_{SS}(t)} \\
& + \underbrace{\begin{bmatrix} \mathbf{P}_t^r \left[ \frac{d^\Delta \mathbf{n}(t)}{dt^\Delta} \right] \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-1)} \mathbf{n}(t)}{dt^{(\Delta-1)}} \right] \\ \vdots \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-N_f+1)} \mathbf{n}(t)}{dt^{(\Delta-N_f+1)}} \right] \end{bmatrix}}_{\mathbf{N}_{SS}(t)} \tag{4.49}
\end{aligned}$$

where  $N_f$  is the length of the feedforward filter matrix or the number of feedforward filter taps and  $\Delta$  is the equalizer's decision delay. The decision delay helps in determining which symbol is detected at the current time,  $t$ .

A more compact representation of (4.49) can be written as:

A more compact representation of (4.49) can be written as:

$$\mathbf{Y}_{SS}(t) = \mathbf{\Gamma}\mathbf{x}(t + \Delta - N_f + 1) + \mathbf{\Omega}\mathbf{U}_{SS}(t) + \mathbf{N}_{SS}(t) \quad (4.50)$$

where  $\mathbf{Y}_{SS}(t)$  is the effective integer-order state-space output vector or effective integer-order state-space received signal,  $\mathbf{U}_{SS}(t)$  is the effective integer-order state-space input vector or effective integer-order state-space transmitted signal,  $\mathbf{x}(t + \Delta - N_f + 1)$  is the state vector,  $\mathbf{N}_{SS}(t)$  is the effective integer-order state-space noise vector,  $\mathbf{\Gamma}$  is the observability matrix and  $\mathbf{\Omega}$  is the Toeplitz matrix.

The block diagram for the DFE for the state-space channel model is shown in Figure 4.9 [107].

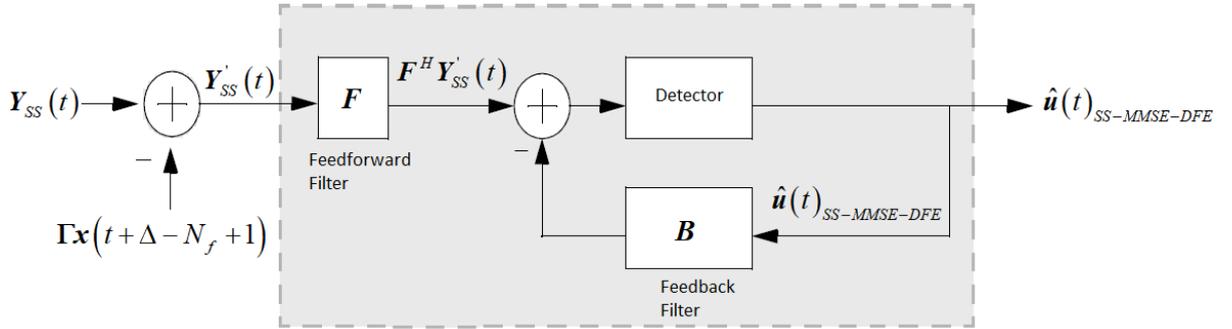


Figure 4. 9: DFE for state-space channel models

With reference to [106] the blockwise data model (4.49) can be decomposed in terms of the impulse response as follows:

$$\begin{aligned}
\underbrace{\begin{bmatrix} \mathbf{P}_t^r \left[ \frac{d^\Delta \mathbf{y}(t)}{dt^\Delta} \right] \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-1)} \mathbf{y}(t)}{dt^{(\Delta-1)}} \right] \\ \vdots \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-N_f+1)} \mathbf{y}(t)}{dt^{(\Delta-N_f+1)}} \right] \end{bmatrix}}_{\mathbf{Y}_{SS}(t)} &= \underbrace{\begin{bmatrix} \mathbf{H}_{N_f} & \mathbf{H}_{N_f+1} & \cdots \\ \vdots & \vdots & \vdots \\ \mathbf{H}_2 & \mathbf{H}_3 & \cdots \\ \mathbf{H}_1 & \mathbf{H}_2 & \cdots \end{bmatrix}}_{\mathbf{\Gamma}\mathbf{x}(t+\Delta-N_f+1), \text{ distant past}} \underbrace{\begin{bmatrix} \mathbf{P}_t^r \left[ \frac{d^{(\Delta-N_f)} \mathbf{u}(t)}{dt^{(\Delta-N_f)}} \right] \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-N_f-1)} \mathbf{u}(t)}{dt^{(\Delta-N_f-1)}} \right] \\ \vdots \end{bmatrix}} \\
&+ \underbrace{\begin{bmatrix} \mathbf{H}_0 & \cdots & \mathbf{H}_{L-1} & 0 \\ 0 & \mathbf{H}_0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & \mathbf{H}_0 \end{bmatrix}}_{\mathbf{\Omega}\mathbf{U}_{SS}(t), \text{ recent past}} \underbrace{\begin{bmatrix} \mathbf{P}_t^r \left[ \frac{d^\Delta \mathbf{u}(t)}{dt^\Delta} \right] \\ \vdots \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-N_f+1)} \mathbf{u}(t)}{dt^{(\Delta-N_f+1)}} \right] \end{bmatrix}} + \underbrace{\begin{bmatrix} \mathbf{P}_t^r \left[ \frac{d^\Delta \mathbf{n}(t)}{dt^\Delta} \right] \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-1)} \mathbf{n}(t)}{dt^{(\Delta-1)}} \right] \\ \vdots \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-N_f+1)} \mathbf{n}(t)}{dt^{(\Delta-N_f+1)}} \right] \end{bmatrix}}_{\mathbf{N}_{SS}(t)} \\
&\quad (4.51)
\end{aligned}$$

Since the term  $\mathbf{\Gamma}\mathbf{x}(t+\Delta-N_f+1)$  is a contribution from distant past symbols it can be subtracted from the effective integer-order state-space output vector or the effective integer-order state-space received signal,  $\mathbf{Y}_{SS}(t)$  resulting in:

$$\mathbf{Y}'_{SS}(t) = \mathbf{Y}_{SS}(t) - \mathbf{\Gamma}\mathbf{x}(t+\Delta-N_f+1) + \mathbf{N}_{SS}(t) \quad (4.52)$$

$$\mathbf{Y}'_{SS}(t) = \left[ \mathbf{\Gamma}\mathbf{x}(t+\Delta-N_f+1) + \mathbf{\Omega}\mathbf{U}_{SS}(t) \right] - \mathbf{\Gamma}\mathbf{x}(t+\Delta-N_f+1) + \mathbf{N}_{SS}(t) \quad (4.53)$$

$$\mathbf{Y}'_{SS}(t) = \mathbf{\Omega}\mathbf{U}_{SS}(t) + \mathbf{N}_{SS}(t) \quad (4.54)$$

This signal is then passed to the feedforward filter which helps in minimising the effects of ISI and noise from future symbols. The signal at the output of the feedforward filter, is expressed as  $\mathbf{F}^H \mathbf{Y}'_{SS}(t)$ , where  $\mathbf{F}$  represents the feedforward filter coefficients.

Signal  $F^H Y_{SS}'(t)$  is then passed through the detector to obtain an estimate of the originally transmitted signal, which is expressed as  $\hat{\mathbf{u}}(t)_{SS-MMSE-DFE}$ . After detection, the detected symbols,  $\hat{\mathbf{u}}(t)_{SS-MMSE-DFE}$  are fed back to the DFE through the feedback filter,  $\mathbf{B}$ . The fed back symbols are then subtracted from the incoming symbols to help minimise the effects of the previously detected symbols from the incoming signal,  $Y_{SS}'(t)$ .

Next, we outline the procedure to follow when calculating the feedforward and feedback filter coefficients. We start by defining matrix  $\mathbf{W}$  which is a combination of the feedforward and feedback filter coefficient matrices expressed as:

$$\mathbf{W} \triangleq \begin{bmatrix} \mathbf{F} \\ \mathbf{B} \end{bmatrix} \quad (4.55)$$

Expressing  $\mathbf{W}$  as a combination of the feedforward and feedback filter coefficients allows for the joint calculation of these matrix coefficients, thus circumventing the need to first calculate the feedforward filter coefficients and then using these to calculate the feedback filter coefficients and vice-versa.

A term  $\mathbf{Z}_{SS}(t)$  which is a combination of the effective output vector  $Y_{SS}'(t)$  and the previously detected symbols,  $\mathbf{D}(t)$  can also be defined as:

$$\mathbf{Z}_{SS}(t) \triangleq \begin{bmatrix} Y_{SS}'(t) \\ \mathbf{D}(t) \end{bmatrix} \quad (4.56)$$

where

$$\mathbf{D}(t) \triangleq \begin{bmatrix} \mathbf{P}_t^r [\mathbf{u}(t)] \\ \vdots \\ \mathbf{P}_t^r \left[ \frac{d^{(1-N_b)} \mathbf{u}(t)}{dt^{(1-N_b)}} \right] \end{bmatrix} \quad (4.57)$$

is a vector of previously detected symbols, and assuming correct detection, these are the same as the transmitted symbols,  $N_b$  is the length of the feedback filter

Matrix  $\mathbf{Z}_{SS}(t)$  plays a great role in the joint calculation of the feedforward and feedback filter coefficients.

The matrix of the combined filter coefficients,  $\mathbf{W}$  when optimised using the criterion defined in [132] becomes:

$$\mathbf{W}_{MMSE} = \arg \min_{F,B} E \left[ \left| U(t) - \mathbf{W} \mathbf{Z}_{SS}(t) \right|^2 \right] \quad (4.58)$$

which simplifies to:

$$\mathbf{W}_{MMSE} = \arg \min_{F,B} E \left[ \left| \mathbf{u}(t) - \hat{\mathbf{u}}(t)_{SS-MMSE-DFE} \right|^2 \right] \quad (4.59)$$

where  $\hat{\mathbf{u}}(t)_{SS-MMSE-DFE}$  is the integer-order equalized signal.

Using (4.56) the auto-correlation matrix,  $\mathbf{R}_{Z_{SS}Z_{SS}}$  of  $\mathbf{Z}_{SS}(t)$  can be defined as:

$$\mathbf{R}_{Z_{SS}Z_{SS}} = E \left[ \mathbf{Z}_{SS}(t) \mathbf{Z}_{SS}^H(t) \right] \quad (4.60)$$

Substituting (4.56) into (4.60) results in:

$$\mathbf{R}_{Z_{SS}Z_{SS}} = \begin{bmatrix} E \left[ \mathbf{Y}'_{SS}(t) (\mathbf{Y}'_{SS})^H(t) \right] & E \left[ \mathbf{Y}'_{SS}(t) \mathbf{D}^H(t) \right] \\ E \left[ \mathbf{D}(t) (\mathbf{Y}'_{SS})^H(t) \right] & E \left[ \mathbf{D}(t) \mathbf{D}^H(t) \right] \end{bmatrix} \quad (4.61)$$

Using (4.54) in (4.61) yields:

$$\mathbf{R}_{Z_{SS}Z_{SS}} = \begin{bmatrix} \frac{P_s}{m} \boldsymbol{\Omega} \boldsymbol{\Omega}^H + N_0 \mathbf{I} & \boldsymbol{\Omega} E[\mathbf{U}_{SS}(t) \mathbf{D}^H(t)] \\ E[\mathbf{D}(t) \mathbf{U}_{SS}^H(t)] \boldsymbol{\Omega}^H & \frac{P_s}{m} \mathbf{I} \end{bmatrix} \quad (4.62)$$

where  $P_s$  is the total transmit power which is assumed to be evenly distributed between all the  $m$  transmit antenna elements.

The cross-correlation matrix,  $\mathbf{R}_{Z_{SS}u}$  of  $\mathbf{Z}_{SS}(t)$  and  $\mathbf{u}(t)$  is defined as:

$$\mathbf{R}_{Z_{SS}u} = E[\mathbf{Z}_{SS}(t) \mathbf{P}_t^r [\mathbf{u}(t)]^H] \quad (4.63)$$

Using (4.56) and then (4.54) in (4.63) yields:

$$\mathbf{R}_{Z_{SS}u} = E \begin{bmatrix} \boldsymbol{\Omega} E[\mathbf{U}_{SS}(t) \mathbf{P}_t^r [\mathbf{u}(t)]^H] \\ \mathbf{0} \end{bmatrix} \quad (4.64)$$

According to [106] and using (4.62) and (4.64) the MMSE solution of the MMSE-DFE can be rewritten as:

$$\mathbf{W}_{MMSE} = \mathbf{R}_{Z_{SS}Z_{SS}}^{-1} \mathbf{R}_{Z_{SS}u} \quad (4.65)$$

Using (4.65) and (4.56), the estimate of the transmitted symbols at the output of the MMSE-DFE,  $\hat{\mathbf{u}}(t)_{SS-MMSE-DFE}$  having considered integer-order state-space channel modelling can be expressed as:

$$\hat{\mathbf{u}}(t)_{SS-MMSE-DFE} = \mathbf{W}_{MMSE}^H \mathbf{Z}_{SS}(t) \quad (4.66)$$

Having presented the theoretical framework for the MMSE-DFE for the FIR massive MIMO channel model and MMSE-DFE for the integer-order state-space massive

MIMO channel model, the following section presents the MATLAB simulation results and analysis for the two models.

## 4.6 Simulation Results

To study the performance of the MMSE-DFE for the FIR massive MIMO channel model and the MMSE-DFE for the inter-order state-space massive MIMO channel model, simulations were run in MATLAB based on the parameters in Table 4.1. The choice of these parameters was based on [106] where they stated that for the MMSE-DFE, given the number of channel paths,  $L$ , the decision delay of the equalizer is given as  $\Delta = L + 2$ , the optimal feedforward filter length is,  $N_f = \Delta + 1$  and the optimal feedback filter length is,  $N_b = L$ . The integer-order MOESP identified system of order  $n = 7$  was considered.

Table 4. 1: System model and channel equalization parameters for integer-order model

Symbol	Description	Value
$L$	Number of channel paths	3
$\Delta$	Decision delay of the equalizer	5
$N_b$	Number of feedback filter taps	3
$N_f$	Number of feedforward filter taps	6

Figure 4.10 shows the performances of the MMSE-DFE for the FIR massive MIMO channel model and the MMSE-DFE for the integer-order state-space massive MIMO channel model for the optimal feedforward filter length of  $N_f = 6$  and the optimal feedback filter length of  $N_b = 3$  for system order  $n = 7$  compared with the originally transmitted signal in each transmitting antenna.

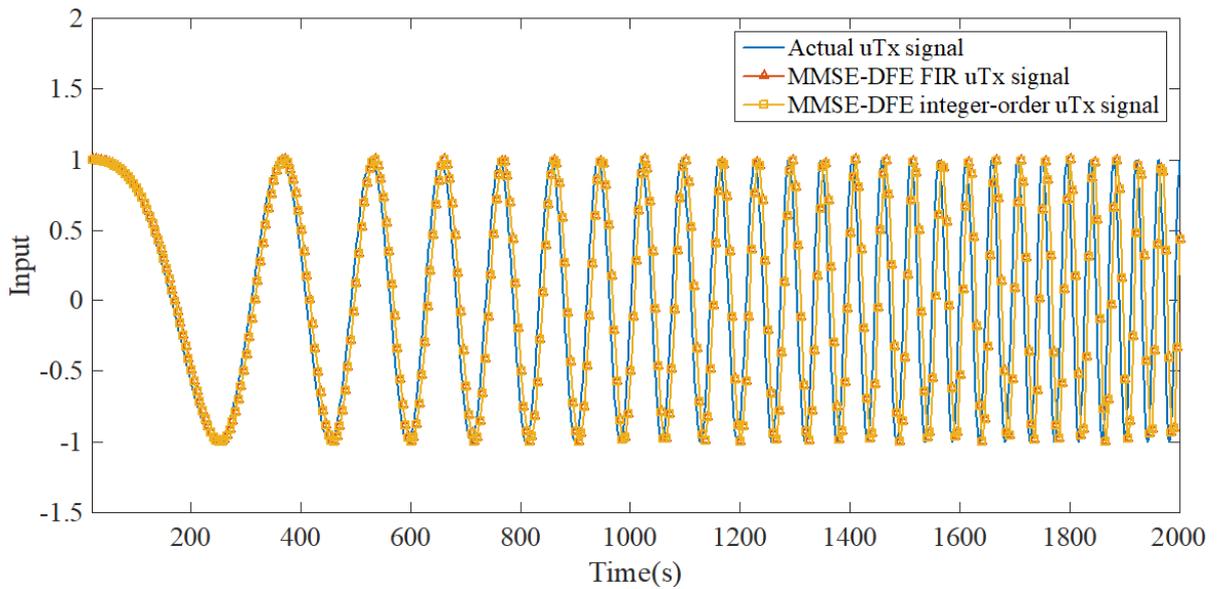


Figure 4. 10: Plot of the input signal, MMSE-DFE signal for the FIR massive MIMO channel model and MMSE-DFE signal for the integer-order state-space massive MIMO channel model in each transmitting antenna

We then zoomed in to show a clear picture of our results and Figure 4.11 shows the zoomed in results comparing the input signal, MMSE-DFE signal for the FIR massive MIMO channel model and MMSE-DFE signal for the integer-order state-space massive MIMO channel model in each transmitting antenna for system order  $n = 7$ .

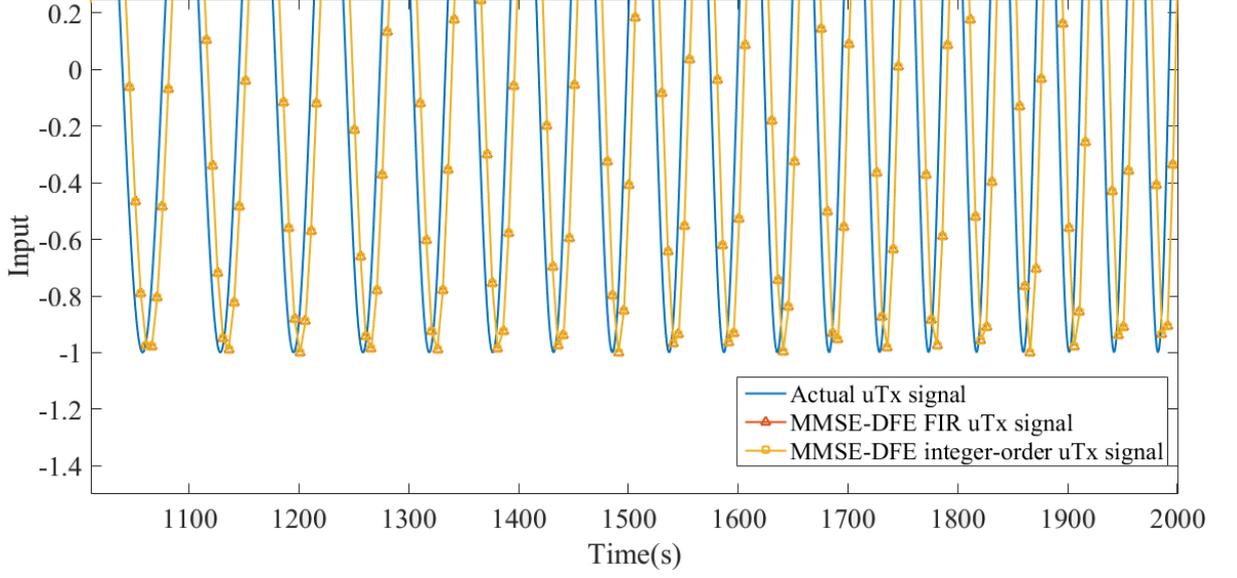


Figure 4. 11: Plot of the input signal, MMSE-DFE signal for the FIR massive MIMO channel model and MMSE-DFE signal for the integer-order state-space massive MIMO channel model in each transmitting antenna (zoomed in)

It can be seen from Figure 4.10 and 4.11 that the MMSE-DFE for the integer-order state-space massive MIMO and the MMSE-DFE for the FIR massive MIMO channel models are both capable of carrying out the desired equalization to recover the transmitted signal.

The integer-order equalized signal,  $\hat{\mathbf{u}}(t)_{SS-MMSE-DFE}$  was then compared with the originally transmitted signal,  $\mathbf{u}(t)$  in terms of the MSE which is expressed as:

$$MSE = \frac{1}{N} \sum_{t=1}^N \left| \mathbf{u}(t) - \hat{\mathbf{u}}(t)_{SS-MMSE-DFE} \right|^2 \quad (4.67)$$

where  $N$  is the number of samples.

The MSE tests for different feedforward,  $N_f$ , feedback,  $N_b$  filter lengths and decision delays,  $\Delta$  are shown in Table 4.2 to Table 4.4. For clarity in comparison, the MSE values have been written to eight decimal places.

Table 4. 2: MSE for  $N_f = 6$ ,  $\Delta = 5$  and different  $N_b$

Number of feedforward filter taps, $N_f$	Number of feedback filter taps, $N_b$	MSE (FIR)	MSE (integer-order)
6	2	0.15273745	0.15273741
6	3	0.15273753	0.15273387

Table 4. 3: MSE for  $N_b = 3$ ,  $\Delta = 5$  and different  $N_f$

Number of feedforward filter taps, $N_f$	Number of feedback filter taps, $N_b$	MSE (FIR)	MSE (integer-order)
3	5	0.15273921	0.15273382
3	6	0.15273753	0.15273387

Table 4. 4: MSE for  $N_f = 6$ ,  $N_b = 3$  and different  $\Delta$

Decision delay, $\Delta$	MSE (FIR)	MSE (integer-order)
4	0.15273740	0.15273910
5	0.15273753	0.15273387

In Tables 4.2 and 4.3 it can be observed that for the optimal values of  $N_f = 6$  and  $N_b = 3$  the MMSE-DFE for the integer-order state-space massive MIMO channel model outperforms the MMSE-DFE for the FIR massive MIMO channel model. This shows the viability of integer-order state-space models in system identification, i.e. channel estimation and also in channel equalization. It can also be observed that for the integer-order state-space massive MIMO channel model changing  $N_f$  and  $N_b$  influences the MSE performance. Reduced values of  $N_f$  and  $N_b$  lead to increased MSE values which means that the performance of the MMSE-DFE equalizers get compromised by the reduction of these parameters. In Table 4.4 different values of  $\Delta$  were considered for  $N_f = 6$ ,  $N_b = 3$ . It can be observed that the MMSE-DFE for the integer-order state-space massive MIMO channel model still outperforms the MMSE-DFE for the FIR massive MIMO channel model.

## 4.7 Summary

Here, the equalization of the massive multiple-input multiple-output (MIMO) frequency-selective wireless channels having considered the finite impulse response (FIR) channel model and the integer-order state-space channel model was presented. The minimum mean square error – decision feedback equalizer (MMSE-DFE) was selected as the equalizer of choice in discussing equalization in the two channel models owing to the fact that it benefits from both the advantages of the MMSE and the DF equalizers, i.e. the MMSE equalizer is a better performing linear equalizer and the DF equalizer is able to minimise the ISI. The MMSE-DFE was developed for both the FIR channel model and the integer-order state-space channel model with reference to some work by previous authors. The observability matrix of the integer-order state-space equalizer was constructed using the matrices obtained through the inter-order MOESP algorithm. The performance of the proposed equalizer was studied for both the FIR channel model and the integer-order state-space channel model through analytical modelling in MATLAB. The performance results showed that the MMSE-DFE for the integer-order state-space massive MIMO channel model performed better than the MMSE-DFE for the FIR massive MIMO channel model especially for the optimal feedforward and feedback filter lengths. This continues to show the viability of the integer-order state-space models in system identification, i.e. channel estimation and also in channel equalization.

## Chapter 5

# Massive MIMO Channel Equalization Using Fractional-Order Models

### 5.1 Introduction

The main contribution of this chapter is on channel equalization in the context of fractional-order models. The equalization technique employed here takes from works by Zhang and Bitmead [106], Zhang and Bitmead [107] and Al-Dhahir and Sayed [108] where they proposed the use of the minimum mean square error – decision feedback equalizer (MMSE-DFE) to combat the effects of ISI in a multiple-input multiple-output (MIMO) system. The application of the MMSE-DFE in the equalization of the fractional-order massive MIMO channel model is a novelty in wireless communications. The choice of the MMSE-DFE is that it benefits from both the advantages of the MMSE and the DF equalizers. The input-output block data in the fractional-order equalization model is presented just like in the extensively studied FIR equalization model. The length of input-output block data is dependent on the number of feedforward filter taps used in the construction of the equalizer. The feedforward and feedback filter coefficients will be jointly calculated using the technique outlined in [106]. The performance of channel equalization using fractional-order modelling will then be compared with channel equalization using integer-order state-space modelling and also channel equalization for the finite impulse response (FIR) model.

### 5.2 Minimum Mean Square Error – Decision Feedback Equalizer for the Fractional-Order Channel Model

Since the continuous-time state-space representation of commensurate fractional-order systems is the same as that of integer-order systems [88] and the analysis of the MOESP fractional-order model follows that of the classical MOESP model as proposed

in [65] and [84], we conclude that the MMSE-DFE for the commensurate fractional-order system will also follow the procedure outlined in [106] for the MMSE-DFE for the state-space model. Based on this, the equalization of the fractional-order channel model is presented in the following section.

Using continuous-time fractional-order state-space modelling, the dynamics of the linear time invariant length- $L$  ISI channel massive MIMO system can be expressed as:

$$D^\alpha \mathbf{x}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad (5.1a)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) + \mathbf{n}(t) \quad (5.1b)$$

where  $\mathbf{x}(t)$  is the  $n \times 1$  state vector,  $\mathbf{u}(t)$  is the  $m \times 1$  input vector,  $\mathbf{y}(t)$  is the  $p \times 1$  output vector,  $\mathbf{n}(t)$  is the  $p \times 1$  additive white Gaussian noise vector,  $\mathbf{A}$  is the  $n \times n$  system matrix and it describes the dynamics of the system, i.e. the eigenvalues of the system,  $\mathbf{B}$  is the  $n \times m$  input matrix and it describes the linear transformation by which the inputs influence the next state,  $\mathbf{C}$  is the  $p \times n$  output matrix, and it describes how the state is transferred to the output,  $\mathbf{D}$  is the  $p \times m$  feed-forward matrix,  $\alpha$  is the commensurate fractional-order and  $D^\alpha$  is the fractional derivative of order  $\alpha$ .

Having obtained matrices  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$  and  $\mathbf{D}$  using the fractional-order system identification algorithm, these matrices can then be used for massive MIMO channel equalization in the context of fractional-order channel modelling. Following the equalization procedure outline in [106] which uses the MMSE-DFE to recover the transmitted signal, a length  $N_f$  block of the input-output data, i.e. the blockwise data model for the commensurate fractional-order model (5.1) can be expressed as:

$$\begin{aligned}
& \underbrace{\begin{bmatrix} \mathbf{P}_t^r \left[ \frac{d^{\Delta\alpha} \mathbf{y}(t)}{dt^{\Delta\alpha}} \right] \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-1)\alpha} \mathbf{y}(t)}{dt^{(\Delta-1)\alpha}} \right] \\ \vdots \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-N_f+1)\alpha} \mathbf{y}(t)}{dt^{(\Delta-N_f+1)\alpha}} \right] \end{bmatrix}}_{\mathbf{Y}_{FO}(t)} = \underbrace{\begin{bmatrix} \mathbf{CA}^{N_f-1} \\ \vdots \\ \mathbf{CA} \\ \mathbf{C} \end{bmatrix}}_{\mathbf{\Gamma}} \mathbf{x}(t + \Delta - N_f + 1) \\
& + \underbrace{\begin{bmatrix} \mathbf{D} & \mathbf{CB} & \mathbf{CAB} & \dots & \mathbf{CA}^{N_f-2} \mathbf{B} \\ \mathbf{0} & \mathbf{D} & \mathbf{CB} & \dots & \mathbf{CA}^{N_f-3} \mathbf{B} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{D} & \mathbf{CB} \\ \mathbf{0} & \dots & \dots & \mathbf{0} & \mathbf{D} \end{bmatrix}}_{\mathbf{\Omega}} \underbrace{\begin{bmatrix} \mathbf{P}_t^r \left[ \frac{d^{\Delta\alpha} \mathbf{u}(t)}{dt^{\Delta\alpha}} \right] \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-1)\alpha} \mathbf{u}(t)}{dt^{(\Delta-1)\alpha}} \right] \\ \vdots \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-N_f+1)\alpha} \mathbf{u}(t)}{dt^{(\Delta-N_f+1)\alpha}} \right] \end{bmatrix}}_{\mathbf{U}_{FO}(t)} \\
& + \underbrace{\begin{bmatrix} \mathbf{P}_t^r \left[ \frac{d^{\Delta\alpha} \mathbf{n}(t)}{dt^{\Delta\alpha}} \right] \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-1)\alpha} \mathbf{n}(t)}{dt^{(\Delta-1)\alpha}} \right] \\ \vdots \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-N_f+1)\alpha} \mathbf{n}(t)}{dt^{(\Delta-N_f+1)\alpha}} \right] \end{bmatrix}}_{\mathbf{N}_{FO}(t)}
\end{aligned} \tag{5.2}$$

where  $N_f$  is the length of the feedforward filter matrix or the number of feedforward filter taps and  $\Delta$  is the equalizer's decision delay. The decision delay helps in determining which symbol is detected at the current time,  $t$ .

A more compact representation of (5.2) can be written as:

$$\mathbf{Y}_{FO}(t) = \mathbf{\Gamma} \mathbf{x}(t + \Delta - N_f + 1) + \mathbf{\Omega} \mathbf{U}_{FO}(t) + \mathbf{N}_{FO}(t) \tag{5.3}$$

where  $\mathbf{Y}_{FO}(t)$  is the effective fractional-order output vector,  $\mathbf{U}_{FO}(t)$  is the effective fractional-order input vector,  $\mathbf{x}(t+\Delta-N_f+1)$  is the state vector,  $\mathbf{N}_{FO}(t)$  is the effective fractional-order noise vector,  $\mathbf{\Gamma}$  is the observability matrix and  $\mathbf{\Omega}$  is the Toeplitz matrix.

The block diagram for the DFE for the fractional-order channel model is shown in Figure 5.1 [107]

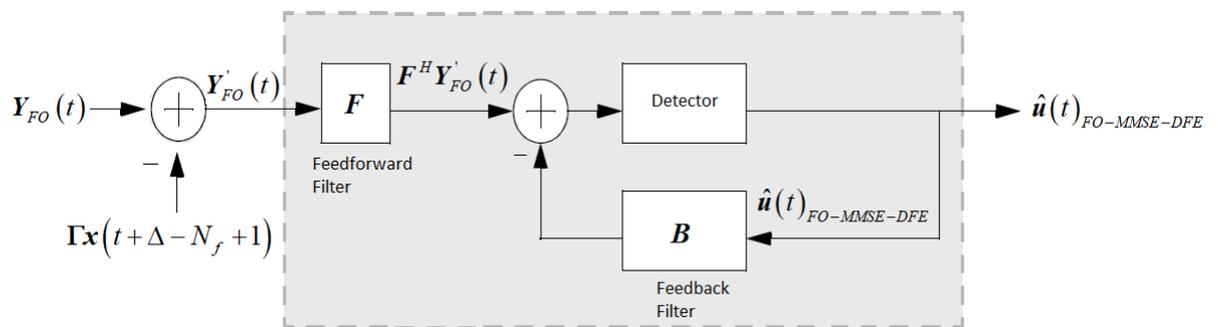


Figure 5. 1: DFE for fractional-order channel models

With reference to [106] the blockwise data model (5.2) can be decomposed in terms of the impulse response as follows:

$$\begin{aligned}
& \underbrace{\begin{bmatrix} \mathbf{P}_t^r \left[ \frac{d^{\Delta\alpha} \mathbf{y}(t)}{dt^{\Delta\alpha}} \right] \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-1)\alpha} \mathbf{y}(t)}{dt^{(\Delta-1)\alpha}} \right] \\ \vdots \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-N_f+1)\alpha} \mathbf{y}(t)}{dt^{(\Delta-N_f+1)\alpha}} \right] \end{bmatrix}}_{\mathbf{Y}_{FO}(t)} = \underbrace{\begin{bmatrix} \mathbf{H}_{N_f} & \mathbf{H}_{N_f+1} & \cdots \\ \vdots & \vdots & \vdots \\ \mathbf{H}_2 & \mathbf{H}_3 & \cdots \\ \mathbf{H}_1 & \mathbf{H}_2 & \cdots \end{bmatrix}}_{\Gamma \mathbf{x}(t+\Delta-N_f+1), \text{ distant past}} \underbrace{\begin{bmatrix} \mathbf{P}_t^r \left[ \frac{d^{(\Delta-N_f)\alpha} \mathbf{u}(t)}{dt^{(\Delta-N_f)\alpha}} \right] \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-N_f-1)\alpha} \mathbf{u}(t)}{dt^{(\Delta-N_f-1)\alpha}} \right] \\ \vdots \end{bmatrix}} \\
& + \underbrace{\begin{bmatrix} \mathbf{H}_0 & \cdots & \mathbf{H}_{L-1} & 0 \\ 0 & \mathbf{H}_0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & \mathbf{H}_0 \end{bmatrix}}_{\Omega \mathbf{U}_{FO}(t), \text{ recent past}} \underbrace{\begin{bmatrix} \mathbf{P}_t^r \left[ \frac{d^{\Delta\alpha} \mathbf{u}(t)}{dt^{\Delta\alpha}} \right] \\ \vdots \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-N_f+1)\alpha} \mathbf{u}(t)}{dt^{(\Delta-N_f+1)\alpha}} \right] \end{bmatrix}} + \underbrace{\begin{bmatrix} \mathbf{P}_t^r \left[ \frac{d^{\Delta\alpha} \mathbf{n}(t)}{dt^{\Delta\alpha}} \right] \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-1)\alpha} \mathbf{n}(t)}{dt^{(\Delta-1)\alpha}} \right] \\ \vdots \\ \mathbf{P}_t^r \left[ \frac{d^{(\Delta-N_f+1)\alpha} \mathbf{n}(t)}{dt^{(\Delta-N_f+1)\alpha}} \right] \end{bmatrix}}_{\mathbf{N}_{FO}(t)} \quad (5.4)
\end{aligned}$$

Since the term  $\Gamma \mathbf{x}(t+\Delta-N_f+1)$  is a contribution from distant past symbols it can be subtracted from the effective fractional-order output vector,  $\mathbf{Y}_{FO}(t)$  resulting in:

$$\mathbf{Y}'_{FO}(t) = \mathbf{Y}_{FO}(t) - \Gamma \mathbf{x}(t+\Delta-N_f+1) + \mathbf{N}_{FO}(t) \quad (5.5)$$

$$\mathbf{Y}'_{FO}(t) = \left[ \Gamma \mathbf{x}(t+\Delta-N_f+1) + \Omega \mathbf{U}_{FO}(t) \right] - \Gamma \mathbf{x}(t+\Delta-N_f+1) + \mathbf{N}_{FO}(t) \quad (5.6)$$

$$\mathbf{Y}'_{FO}(t) = \Omega \mathbf{U}_{FO}(t) + \mathbf{N}_{FO}(t) \quad (5.7)$$

This signal is then passed to the feedforward filter which helps in minimising the effects of ISI and noise from future symbols. The signal at the output of the feedforward filter, is expressed as  $\mathbf{F}^H \mathbf{Y}'_{FO}(t)$ , where  $\mathbf{F}$  represents the feedforward filter coefficients. Signal  $\mathbf{F}^H \mathbf{Y}'_{FO}(t)$  is then passed through the detector to obtain an estimate of the

originally transmitted signal, which is expressed as  $\hat{\mathbf{u}}(t)_{FO-MMSE-DFE}$ . After detection, the detected symbols,  $\hat{\mathbf{u}}(t)_{FO-MMSE-DFE}$  are fed back to the DFE through the feedback filter,  $\mathbf{B}$ . The fed back symbols are then subtracted from the incoming symbols to help minimise the effects of the previously detected symbols from the incoming signal,  $\mathbf{Y}'_{FO}(t)$ .

Next, we outline the procedure to follow when calculating the feedforward and feedback filter coefficients. We start by defining matrix  $\mathbf{W}$  which is a combination of the feedforward and feedback filter coefficient matrices expressed as:

$$\mathbf{W} \triangleq \begin{bmatrix} \mathbf{F} \\ \mathbf{B} \end{bmatrix} \quad (5.8)$$

Expressing  $\mathbf{W}$  as a combination of the feedforward and feedback filter coefficients allows for the joint calculation of these matrix coefficients, thus circumventing the need to first calculate the feedforward filter coefficients and then using these to calculate the feedback filter coefficients and vice-versa.

A term  $\mathbf{Z}_{FO}(t)$  which is a combination of the effective fractional-order output vector,  $\mathbf{Y}'_{FO}(t)$  and the previously detected symbols,  $\mathbf{D}(t)$  can also be defined as:

$$\mathbf{Z}_{FO}(t) \triangleq \begin{bmatrix} \mathbf{Y}'_{FO}(t) \\ \mathbf{D}(t) \end{bmatrix} \quad (5.9)$$

where

$$\mathbf{D}(t) \triangleq \begin{bmatrix} \mathbf{P}_t^r [\mathbf{u}(t)] \\ \vdots \\ \mathbf{P}_t^r \left[ \frac{d^{(1-N_b)\alpha} \mathbf{u}(t)}{dt^{(1-N_b)\alpha}} \right] \end{bmatrix} \quad (5.10)$$

is a vector of previously detected symbols, and assuming correct detection, these are the same as the transmitted symbols,  $N_b$  is the length of the feedback filter.

Matrix  $\mathbf{Z}_{FO}(t)$  plays a great role in the joint calculation of the feedforward and feedback filter coefficients.

The matrix of the combined filter coefficients,  $\mathbf{W}$  when optimised using the criterion defined in [132] becomes:

$$\mathbf{W}_{MMSE} = \arg \min_{F,B} E \left[ \left| U(t) - \mathbf{W} \mathbf{Z}_{FO}(t) \right|^2 \right] \quad (5.11)$$

which simplifies to:

$$\mathbf{W}_{MMSE} = \arg \min_{F,B} E \left[ \left| \mathbf{u}(t) - \hat{\mathbf{u}}(t)_{FO-MMSE-DFE} \right|^2 \right] \quad (5.12)$$

where  $\hat{\mathbf{u}}(t)_{FO-MMSE-DFE}$  is the fractional-order equalized signal.

Using (5.9) the auto-correlation matrix,  $\mathbf{R}_{Z_{FO}Z_{FO}}$  of  $\mathbf{Z}_{FO}(t)$  can be defined as:

$$\mathbf{R}_{Z_{FO}Z_{FO}} = E \left[ \mathbf{Z}_{FO}(t) \mathbf{Z}_{FO}^H(t) \right] \quad (5.13)$$

Substituting (5.9) into (5.13) results in:

$$\mathbf{R}_{Z_{FO}Z_{FO}} = \begin{bmatrix} E \left[ \mathbf{Y}'_{FO}(t) (\mathbf{Y}'_{FO})^H(t) \right] & E \left[ \mathbf{Y}'_{FO}(t) \mathbf{D}^H(t) \right] \\ E \left[ \mathbf{D}(t) (\mathbf{Y}'_{FO})^H(t) \right] & E \left[ \mathbf{D}(t) \mathbf{D}^H(t) \right] \end{bmatrix} \quad (5.14)$$

Using (5.7) in (5.14) yields:

$$\mathbf{R}_{Z_{FO}Z_{FO}} = \begin{bmatrix} \frac{P_s}{m} \boldsymbol{\Omega} \boldsymbol{\Omega}^H + N_0 \mathbf{I} & \boldsymbol{\Omega} E[\mathbf{U}_{FO}(t) \mathbf{D}^H(t)] \\ E[\mathbf{D}(t) \mathbf{U}_{FO}^H(t)] \boldsymbol{\Omega}^H & \frac{P_s}{m} \mathbf{I} \end{bmatrix} \quad (5.15)$$

where  $P_s$  is the total transmit power which is assumed to be evenly distributed between all the  $m$  transmit antenna elements.

The cross-correlation matrix,  $\mathbf{R}_{Z_{FO}u}$  of  $\mathbf{Z}_{FO}(t)$  and  $\mathbf{u}(t)$  is defined as:

$$\mathbf{R}_{Z_{FO}u} = E[\mathbf{Z}_{FO}(t) \mathbf{P}_t^r [\mathbf{u}(t)]^H] \quad (5.16)$$

Using (5.9) and then (5.7) in (5.16) yields:

$$\mathbf{R}_{Z_{FO}u} = E \begin{bmatrix} \boldsymbol{\Omega} E[\mathbf{U}_{FO}(t) \mathbf{P}_t^r [\mathbf{u}(t)]^H] \\ \mathbf{0} \end{bmatrix} \quad (5.17)$$

According to [106] and using (5.15) and (5.17) the MMSE solution of the MMSE-DFE can be rewritten as:

$$\mathbf{W}_{MMSE} = \mathbf{R}_{Z_{FO}Z_{FO}}^{-1} \mathbf{R}_{Z_{FO}u} \quad (5.18)$$

Using (5.18) and (5.9), the estimate of the transmitted symbols at the output of the MMSE-DFE,  $\hat{\mathbf{u}}(t)_{FO-MMSE-DFE}$  having considered fractional-order channel modelling can be expressed as:

$$\hat{\mathbf{u}}(t)_{FO-MMSE-DFE} = \mathbf{W}_{MMSE}^H \mathbf{Z}_{FO}(t) \quad (5.19)$$

Having discussed on the MMSE-DFE for the fractional-order channel model it is important in communications to then study the performance of the equalized system in terms of the symbol error rate. In the following section a discussion on the symbol error

rate performance of the MMSE-DFE for the fractional-order channel model is presented.

### 5.3 Symbol Error Rate Performance

The bit error rate (BER) or symbol error rate (SER) is one of the performance parameters used in communication systems and it refers to the ratio of the number of bits or symbols incorrectly received to the total number of bits or symbols transmitted during a specified time interval. In ideal communication systems this ratio is suppose to be zero but in reality there are errors experienced in the communication systems especially those affected by multipath fading, thus this can never be zero. The best performing communication systems are ones that are able to reduce the BER or SER to the smallest possible value.

The performance of the massive MIMO system was studied in terms of the signal-to-noise ratio and the average symbol error rate (SER). According to [119] and [123] the output SNR of the unbiased MMSE-DFE can be expressed as:

$$SNR_{MMSE-DFE-U} = SNR[\mathbf{R}_{ee,\min}]^{-1} - 1 \quad (5.20)$$

and the output SNR of the biased MMSE-DFE can be written as:

$$SNR_{MMSE-DFE} = SNR[\mathbf{R}_{ee,\min}]^{-1} \quad (5.21)$$

where  $\mathbf{R}_{ee,\min}$  is the covariance matrix of the error vector which can be expressed as:

$$\mathbf{R}_{ee,\min} = E[ee^H] \quad (5.22a)$$

$$\mathbf{R}_{ee,\min} = E\left[|\mathbf{u}(t) - \hat{\mathbf{u}}(t)_{FO-MMSE-DFE}(t)|^2\right] \quad (5.22b)$$

According [133] the average SER having considered quadrature phase shift keying (QPSK) modulation for the frequency-selective quasi-static Rayleigh fading channel can then be written as:

$$P_{QPSK} = \frac{1}{2} \left( 1 - \sqrt{\frac{\gamma}{2 + \gamma}} \right) \quad (5.23)$$

The average SER having considered M-ary phase shift keying (PSK) modulation for the frequency-selective quasi-static Rayleigh fading channel can be expressed as [134]:

$$P_{M-PSK} = \frac{1}{\pi} \int_0^{(M-1)\pi/M} M_\gamma \left( \frac{g_{PSK}}{\sin^2 \theta} \right) d\theta \quad (5.24)$$

where  $g_{PSK} = \sin^2(\pi / M)$ .

After some manipulations of (5.24), the average BER for binary phase shift keying (BPSK) modulation for the frequency-selective quasi-static Rayleigh fading channel can then be written as [135]:

$$P_{BPSK} = \frac{1}{2} \left( 1 - \sqrt{\frac{\gamma}{1 + \gamma}} \right) \quad (5.25)$$

With reference to [136] and using the moment generation function (MGF) approach, the average SER having considered M-ary quadrature amplitude modulation (M-QAM) for the frequency-selective quasi-static Rayleigh fading channel can be expressed as:

$$P_{M-QAM} = \frac{4(\sqrt{M} - 1)^{\pi/2}}{\pi\sqrt{M}} \int_0^{\pi/2} M_\gamma \left( -\frac{g_{QAM}}{\sin^2 \theta} \right) d\theta - \frac{4(\sqrt{M} - 1)^2}{\pi M} \int_0^{\pi/4} M_\gamma \left( -\frac{g_{QAM}}{\sin^2 \theta} \right) d\theta \quad (5.26)$$

where  $M$  is the number of constellation points,

$$g_{QAM} = \frac{3}{2(M-1)} \quad (5.27)$$

and

$$M_\gamma(s) = (1 - s\gamma)^{-1} \quad (5.28)$$

is the MGF of the SNR,  $\gamma$  for Rayleigh fading.

The MGF approach is considered on the bases that it is a bit easier to work with as it avoids the need to find the probability density function (PDF) of the output SNR which can be a bit tedious for high-order systems [137]. Substituting (5.28) into (5.26) and after some manipulations, equation (5.26) can then be expressed as [138]:

$$P_{M-QAM} = \frac{1}{M} \left[ M - 1 - \frac{2a(\sqrt{M}-1)}{\sqrt{1+a^2}} - \frac{4a(\sqrt{M}-1)^2}{\pi\sqrt{1+a^2}} \tan^{-1} \left( \frac{a}{\sqrt{1+a^2}} \right) \right] \quad (5.29)$$

where  $a = \sqrt{\frac{3\gamma}{2(M-1)}}$

Replacing  $\gamma$  with  $SNR_{MMSE-DFE}$  in (5.25), (5.23) and (5.29), the average SER for BPSK, QPSK and M-QAM having considered the MMSE-DFE signals can then be expressed as:

$$P_{BPSK} = \frac{1}{2} \left( 1 - \sqrt{\frac{SNR_{MMSE-DFE}}{1 + SNR_{MMSE-DFE}}} \right) \quad (5.30)$$

$$P_{QPSK} = \frac{1}{2} \left( 1 - \sqrt{\frac{SNR_{MMSE-DFE}}{2 + SNR_{MMSE-DFE}}} \right) \quad (5.31)$$

and

$$P_{M-QAM} = \frac{1}{M} \left[ M - 1 - \frac{2a(\sqrt{M}-1)}{\sqrt{1+a^2}} - \frac{4a(\sqrt{M}-1)^2}{\pi\sqrt{1+a^2}} \tan^{-1} \left( \frac{a}{\sqrt{1+a^2}} \right) \right] \quad (5.32)$$

where  $a = \sqrt{\frac{3SNR_{MMSE-DFE}}{2(M-1)}}$

respectively.

Next, we present MATLAB simulation results and analysis of the MMSE-DFE for the fractional-order massive MIMO channel model and these are compared with the MMSE-DFE for FIR massive MIMO channel model and MMSE-DFE for the integer-order massive MIMO channel model results. The symbol error rate performance for the FIR, integer-order and fractional order massive MIMO channel models having applied MMSE-DFE are also presented.

## 5.4 Simulation Results

To study the performance of the MMSE-DFE for the fractional-order massive MIMO channel model, simulations were run in MATLAB based on the parameters in Table 5.1. The choice of these parameters was based on [106] owing to the fact that the continuous-time state-space representation of commensurate fractional-order systems is the same as that of integer-order systems [88]. In [106] they stated that given the number of channel paths,  $L$ , the decision delay of the equalizer is given as  $\Delta = L + 2$ , the optimal feedforward filter length is,  $N_f = \Delta + 1$  and the optimal feedback filter length is,  $N_b = L$ . The fractional-order MOESP identified system with fractional-order of order  $\alpha = 0.1$  was considered.

Table 5. 1: System model and channel equalization parameters for fractional-order model

Symbol	Description	Value
$\alpha$	Fractional-order	0.1
$L$	Number of channel paths	3
$\Delta$	Decision delay of the equalizer	5
$N_b$	Number of feedback filter taps	3
$N_f$	Number of feedforward filter taps	6

Figure 5.2 shows the performance of the MMSE-DFE for the fractional-order massive MIMO channel model for the optimal feedforward filter length of  $N_f = 6$  and the optimal feedback filter length of  $N_b = 3$  compared with the originally transmitted signal in each transmitting antenna for fractional-order  $\alpha = 0.1$ .

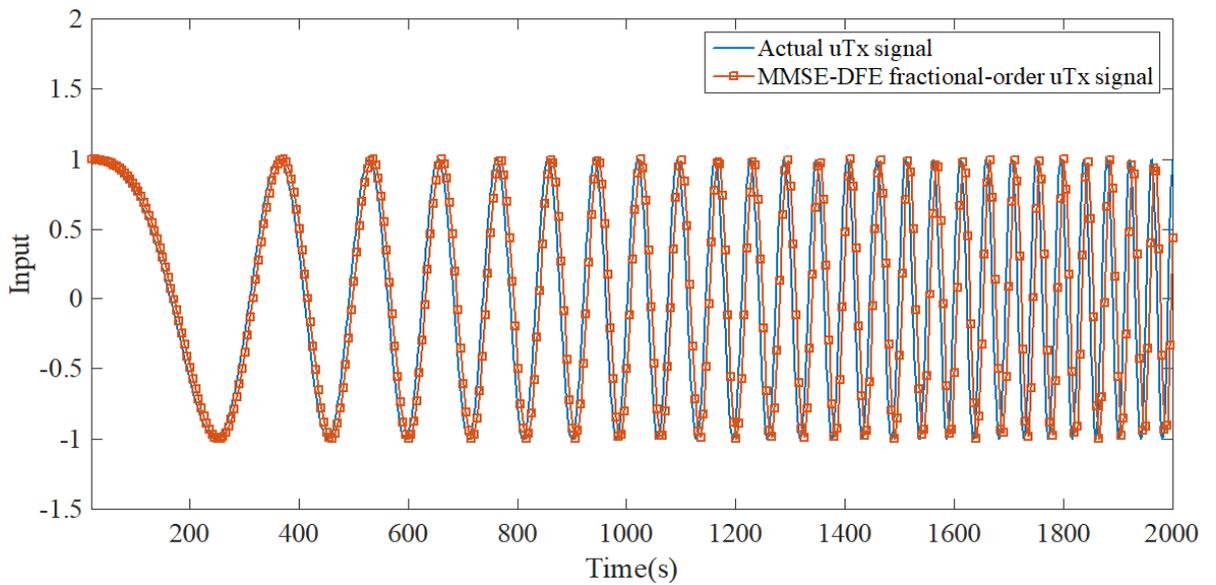


Figure 5. 2: Plot of the input signal and MMSE-DFE signal for the fractional-order massive MIMO channel model in each transmitting antenna

We then zoomed in to show a clear picture of our results and Figure 5.3 shows the zoomed in results comparing the input signal and the MMSE-DFE signal for the fractional-order massive MIMO channel model in each transmitting antenna for  $\alpha = 0.1$ .

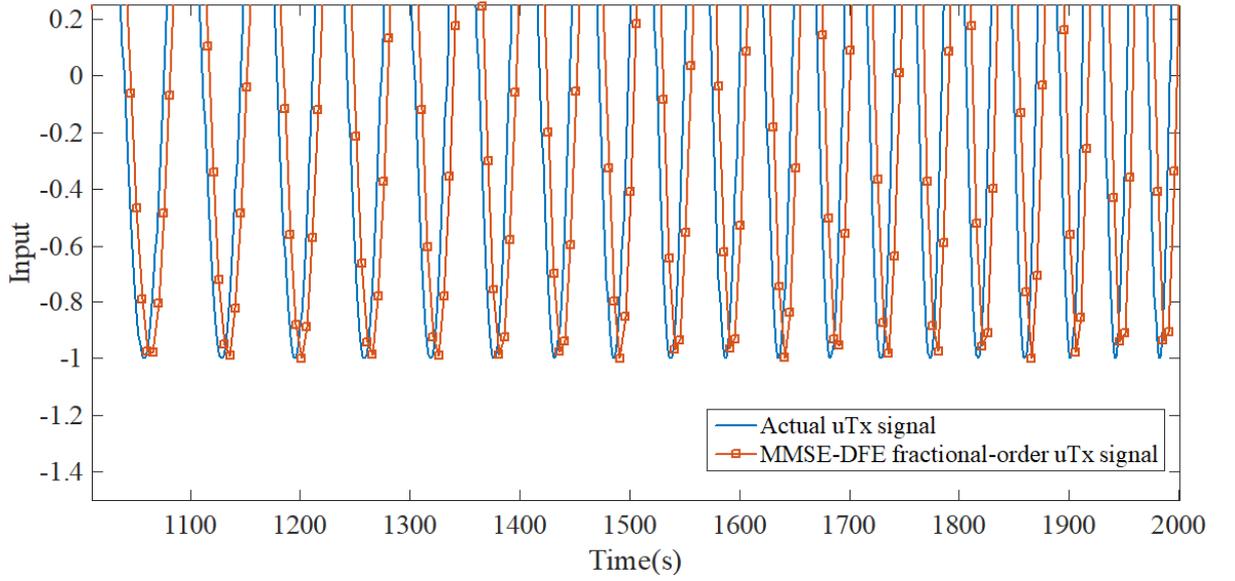


Figure 5. 3: Plot of the input signal and MMSE-DFE signal for the fractional-order massive MIMO channel model in each transmitting antenna (zoomed in)

We can observe from Figure 5.2 and Figure 5.3 that the MMSE-DFE for the fractional-order massive MIMO channel model is capable of carrying out the desired equalization to recover the transmitted signal.

The fractional-order equalized signal,  $\hat{\mathbf{u}}(t)_{FO-MMSE-DFE}$  was then compared with the originally transmitted signal,  $\mathbf{u}(t)$  in terms of the MSE which is expressed as:

$$MSE = \frac{1}{N} \sum_{t=1}^N \left| \mathbf{u}(t) - \hat{\mathbf{u}}(t)_{FO-MMSE-DFE} \right|^2 \quad (5.16)$$

where  $N$  is the number of samples and  $\hat{\mathbf{u}}(t)_{FO-MMSE-DFE}$  is the fractional-order equalized signal.

The MSE tests for different feedforward,  $N_f$ , feedback,  $N_b$  filter lengths and decision delays,  $\Delta$  are shown in Table 5.2 to Table 5.4. For clarity in comparison, the MSE values have been written to eight decimal places.

Table 5. 2: MSE for  $N_f = 6$ ,  $\Delta = 5$  and different  $N_b$

Number of feedforward filter taps, $N_f$	Number of feedback filter taps, $N_b$	MSE (FIR)	MSE (integer-order)	MSE (fractional-order)
6	2	0.15273745	0.15273741	0.15273741
6	3	0.15273753	0.15273387	0.15273387

Table 5. 3: MSE for  $N_b = 3$ ,  $\Delta = 5$  and different  $N_f$

Number of feedforward filter taps, $N_f$	Number of feedback filter taps, $N_b$	MSE (FIR)	MSE (integer-order)	MSE (fractional-order)
3	5	0.15273921	0.15273382	0.15273382
3	6	0.15273753	0.15273387	0.15273387

Table 5. 4: MSE for  $N_f = 6$ ,  $N_b = 3$  and different  $\Delta$

Decision delay, $\Delta$	MSE (FIR)	MSE (integer-order)	MSE (fractional-order)
4	0.15273740	0.15273910	0.15273910
5	0.15273753	0.15273387	0.15273387

In Tables 5.2 and 5.3 it can be observed that the optimal choices of  $N_f$  and  $N_b$  that give better results for the MMSE-DFE for the fractional-order massive MIMO channel model are  $N_f = 6$  and  $N_b = 3$  respectively. In Table 5.4, different decision delay values were considered for  $N_f = 6$ ,  $N_b = 3$ . It can be observed that  $\Delta = 5$  is the delay choice that gives better results.

Next, we present the simulation results for the symbol error rate performance for the fractional-order massive MIMO channel model in comparison with the FIR and integer-order channel models. For our simulations we considered BPSK, QPSK and 256-QAM modulated signals. Figures 5.4 to 5.6 show the SER performances for the

FIR, integer-order and fractional-order massive MIMO channel models having applied MMSE-DFE plotted against the signal-to-noise ratio for BPSK, QPSK and 256-QAM modulated signals.

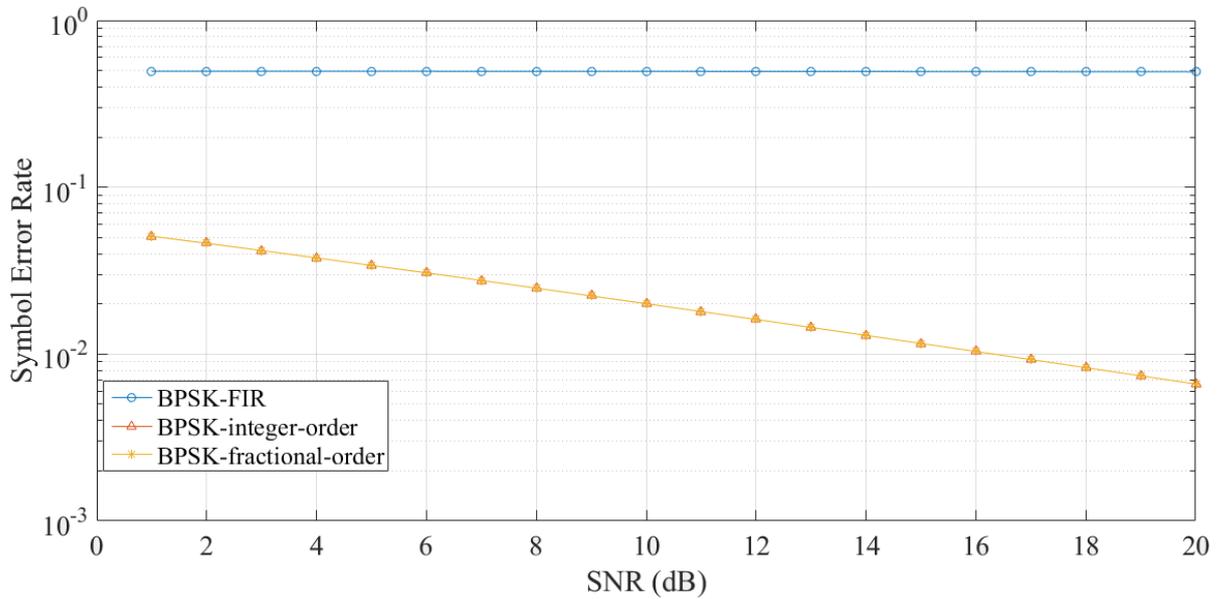


Figure 5. 4: Plot of SER vs. SNR for the MMSE-DFE signal for the FIR, integer-order and fractional-order massive MIMO channel models for BPSK modulated signals

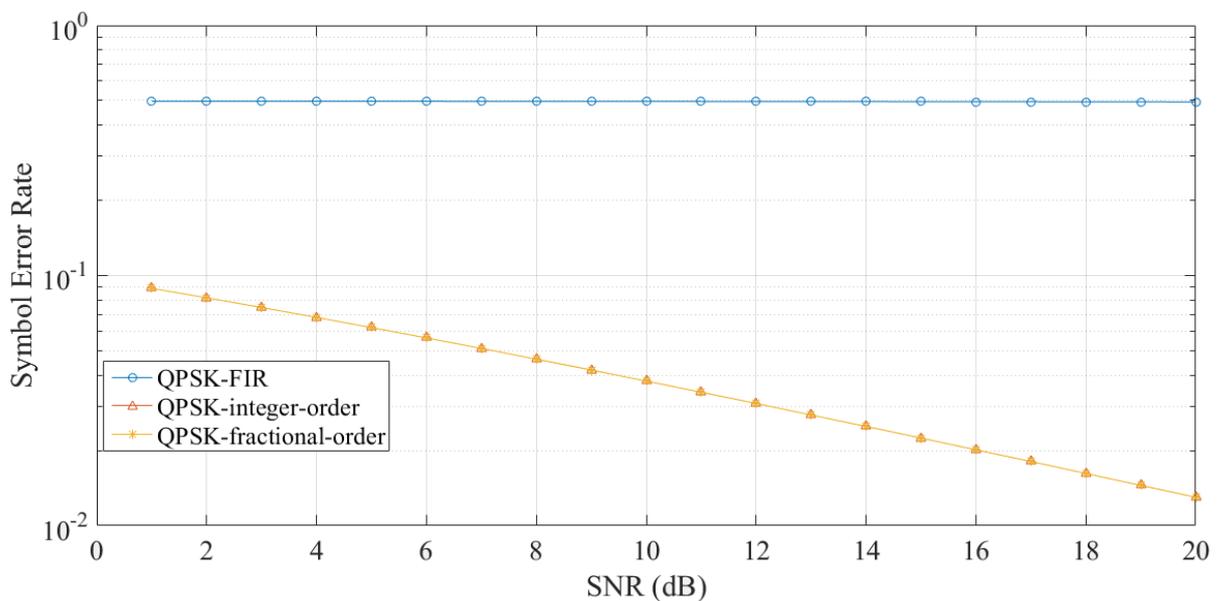


Figure 5. 5: Plot of SER vs. SNR for the MMSE-DFE signal for the FIR, integer-order and fractional-order massive MIMO channel models for QPSK modulated signals

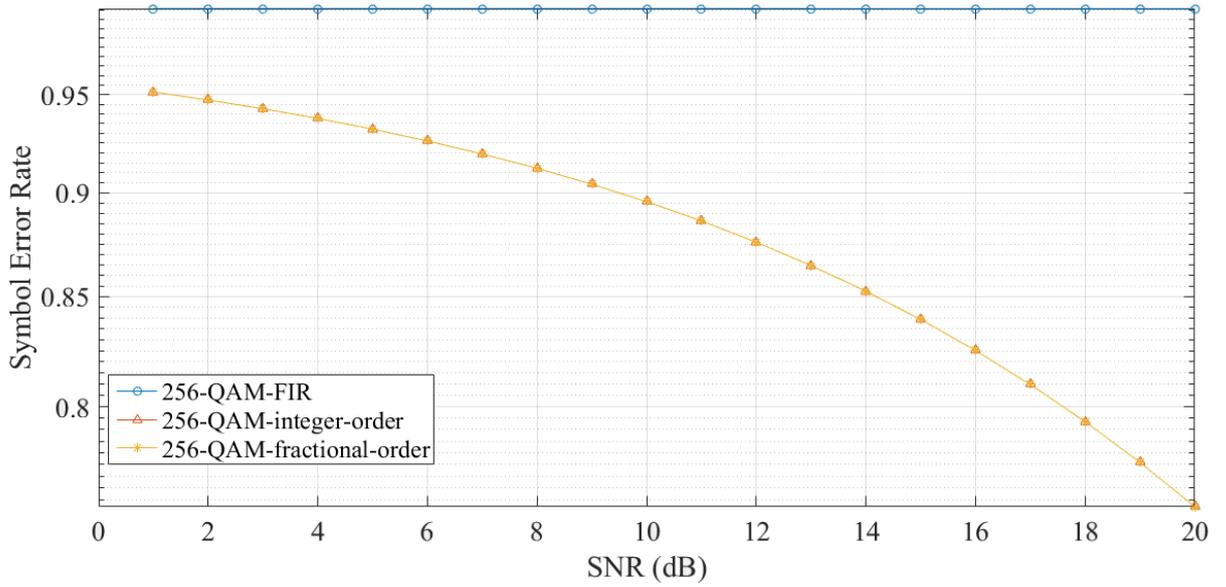


Figure 5. 6: Plot of SER vs. SNR for the MMSE-DFE signal for the FIR, integer-order and fractional-order massive MIMO channel models for 256QAM modulated signals

We can observe from Figure 5.4 to Figure 5.6 that the MMSE-DFE for the fractional-order massive MIMO channel model has lower symbol error rate results compared to the MMSE-DFE for the FIR massive MIMO channel model. This implies that the fractional-order MMSE-DFE model can be a more robust choice than the FIR MMSE-DFE model when it comes to massive MIMO channel equalization. It can also be observed that the SER performance of the MMSE-DFE integer-order state-space channel model of order  $n = 7$  compares with that of the fractional-order channel model of fractional-order  $\alpha = 0.7$  which continues to confirm the MSE results we previously presented. This proves the viability of the fractional-order state space models in channel equalization.

## 5.5 Summary

We presented the equalization of the massive multiple-input multiple-output (MIMO) frequency-selective wireless channels having considered the fractional-order channel model. The minimum mean square error – decision feedback equalizer (MMSE-DFE) was selected as the equalizer of choice owing to the fact that it benefits from both the advantages of the MMSE and the DF equalizers, i.e. the MMSE equalizer is a better performing linear equalizer and the DF equalizer is able to minimise the ISI. The

observability matrix of the fractional-order equalizer was constructed using the matrices obtained through the fractional-order MOESP identification algorithm. The performance of the proposed equalizer was then studied using analytical modelling in MATLAB. The performance results showed that the MMSE-DFE for the fractional-order massive MIMO channel model performed better than the MMSE-DFE for the FIR massive MIMO channel model. The fractional-order massive MIMO channel model also showed an improved symbol error rate performance when compared to the FIR massive MIMO channel model. This implies that the fractional-order massive MIMO channel model can be a more robust choice than the FIR massive MIMO channel model when it comes to channel estimation and equalization. These results continue to show the viability of the fractional-order models in system identification, i.e. channel estimation and in channel equalization in wireless communications.

# Chapter 6

## Conclusions and Future Work

### 6.1 Conclusions

The main focus of this thesis was on five main topics: i) Addressing the issue of IQ imbalance which degrades the performance of massive MIMO systems, through IQ imbalance compensation, ii) Massive MIMO continuous-time system identification using the integer-order subspace algorithm, iii) Massive MIMO continuous-time system identification using the fractional-order algorithm, iv) Massive MIMO channel equalization using the integer-order subspace model and v) Massive MIMO channel equalization using the fractional-order model. In studying the effect of IQ imbalance on the performance of the massive MIMO system we considered IQ imbalance at the receiver end. IQ imbalance compensation was then applied to mitigate the effects of this RF impairment on the massive MIMO system. The IQ imbalance compensated system was shown to outperform the one without IQ imbalance compensation, showing the importance of addressing issues associated with RF impairments in massive MIMO systems. In addressing the issue of IQ imbalance in the massive MIMO system we assumed that the channel state information is known, but in reality, the channel information is not readily available meaning that it needs to be estimated first. To estimate the massive MIMO channel, we used subspace system identification algorithms. In this thesis the MOESP algorithm, a subset of subspace system identification algorithms was considered when studying the dynamics of the open loop massive MIMO system. The performance evaluation was performed over continuous-time domain using the input and output data having considered the influence of the measurement noise and the chirp signal was used as the excitation signal. Before applying the MOESP identification algorithm the input and output data was first transformed using the Poisson moment functional (PMF) approach, which is a filtering method that helps with overcoming the problem associated with the input-output data time-derivatives in continuous-time system identification. In both the integer-order and

fractional-order massive MIMO continuous-time system identification, the main idea was to have a model that would accurately model the system dynamics, i.e. to accurately find matrices  $A$ , system matrix, which describes the dynamics of the system, i.e. the eigenvalues of the system,  $B$ , input matrix, which describes the linear transformation by which the inputs influence the next state,  $C$ , output matrix, which describes how the state is transferred to the output and  $D$ , feed-forward matrix. Then in both the integer-order and fractional-order massive MIMO channel equalization, the main idea was to have a channel equalization model that would give an accurate estimate of the transmitted symbols,  $\hat{\mathbf{u}}(t)$ .

A discussion on the use of the continuous-time integer-order state-space models to represent massive multiple-input multiple-output (MIMO) frequency-selective wireless channels for better system identification or better channel estimation was presented in chapter 2. The integer-order multiple-input multiple-output output-error state space (MOESP) algorithm was used to identify the linear-time invariant continuous-time massive MIMO frequency-selective wireless channels. The input-output data was filtered using the PMF filter to help overcome the problem associated with the input-output data time-derivatives in the continuous-time system identification, after which the MOESP identification algorithm outlined for discrete-time system identification was then applied to the PMF filtered input-output data. MATLAB simulations were then run to identify the massive MIMO system using the continuous-time MOESP subspace system identification algorithm. The results showed an improvement in the performance of the continuous-time MOESP subspace system identification algorithm as the system order was increased. This showed that the MOESP algorithm could be used to identify the dynamics of the massive MIMO system, thus it could be used for massive MIMO channel estimation.

In Chapter 3, the fractional-order system identification algorithm was discussed as an alternative to the integer-order system identification algorithm. Fractional-order calculus is very important, especially in explaining many events which traditional mathematics cannot explain. It has also been observed that many real-world physical systems are well characterized by fractional-order differential equations rather than using classical integer-order models. The fractional-order time-derivatives of the input-

output data are generally not measured meaning that the input-output matrices are not known, as a result the classical subspace methods originally developed for the identification of discrete-time models cannot be directly adapted for the identification of continuous-time fractional-order models. To address this problem the Poisson moment functional (PMF) approach was used when dealing with continuous-time fractional-order system identification, where the input-output data is first filtered using the PMF filter after which the MOESP identification algorithm outlined for discrete-time system identification was then applied to the PMF filtered input-output data. The MATLAB simulation results showed that the proposed continuous-time MOESP fractional-order system identification algorithm could be used to identify the dynamics of the massive MIMO system. Due to the challenge involved in dealing with fractional-order transfer functions the discussion was extended to the use of rational transfer functions that could be used to approximate these fractional-order transfer functions. Meaning that whenever there is a fractional-order transfer function in system identification there is need to replace it with an easier to handle approximate rational transfer function. The Oustaloup continuous-time approximation method was used to present the fractional-order transfer function of the massive MIMO frequency-selective wireless channels as an approximated rational transfer function. The performance of the Oustaloup method was studied through MATLAB simulations and it was found to be a very close match to representing the massive MIMO system.

The discussion on the use of channel equalization as a means of mitigating the effects of inter-symbol interference (ISI) caused by the massive multiple-input multiple-output (MIMO) frequency-selective wireless channels was presented in chapter 4. The equalizer of choice was the MMSE-DFE as it benefits from both the advantages of the MMSE and the DF equalizers. The input-output block data in the integer-order state-space equalization model was presented just like in the extensively studied FIR equalization model. The length of input-output block data is dependent on the number of feedforward filter taps used in the construction of the equalizer. Unlike in most studies where the feedback filter coefficients and feedforward filter coefficients are calculated independently in this thesis these were jointly calculated. Using MATLAB simulations, the performance of channel equalization using integer-order state-space modelling was then compared with the performance of channel equalization using the

finite impulse response (FIR) model. The results showed that the MMSE-DFE for the integer-order massive MIMO channel model performed better than the MMSE-DFE for the FIR massive MIMO channel model especially for the optimal feedforward and feedback filter lengths.

Chapter 5 focused on the discussion on channel equalization in the context of fractional-order models where the minimum mean square error – decision feedback equalizer (MMSE-DFE) was chosen as the equalizer of choice. The input-output block data in the fractional-order equalization model was presented just like in the extensively studied FIR equalization model. The length of input-output block data is dependent on the number of feedforward filter taps used in the construction of the equalizer. The feedback filter and feedforward filter coefficients were then jointly calculated. The performance of channel equalization using fractional-order modelling was then compared with channel equalization using integer-order state-space modelling and channel equalization for the finite impulse response (FIR) model through MATLAB simulations where it was shown that the performance of the fractional-order equalization model was better than the FIR equalization model. The system performance was also studied in terms of the SER. It was observed from the MATLAB results that the MMSE-DFE for the fractional-order massive MIMO channel model had a lower symbol error rate compared to the MMSE-DFE for the FIR massive MIMO channel model. This implies that the fractional-order MMSE-DFE model can be a more robust choice than the FIR MMSE-DFE model when it comes to massive MIMO channel equalization. This proves the viability of the fractional-order state space models in channel equalization.

All these conclusions demonstrate the validity of the fractional-order models in the identification and equalization of the massive MIMO frequency-selective wireless channels.

## 6.2 Future Work

One of the main focus points of this thesis was on the identification and equalization of the quasi-static or slowly-varying, frequency-selective Rayleigh fading massive MIMO channels using the integer-order and fractional-order algorithms. But this research could be extended to the identification and equalization of the time-varying, frequency-selective Rayleigh fading massive MIMO channels using these proposed algorithms as time-varying channels are usually the case in wireless mobile communications. Time-varying channels are always varying in time due to the mobility of the transmitter and/or receiver unlike quasi-static or time-invariant channels where there is no change within each frame, but the change occurs between frames. The mobility of transmitter and/or receiver is known to cause the Doppler effect which affects channel estimation and also degrades channel equalization. Doppler effect is the change in frequency of a signal due to the motion between the transmitter and/or receiver. So it would be interesting to study the performance of the identification and equalization algorithms under these conditions.

In studying the performance of the identification and equalization algorithms in the massive MIMO system an ideal case free of pilot contamination, phase noise and amplifier nonlinearity was considered. It would be of great interest to consider the effects of pilot contamination and some of the radio frequency (RF) impairments on the performance of the identification algorithms. This would probably be useful in designing more practical massive MIMO systems that would greatly achieve the benefits of multiuser MIMO such as increased capacity, increased data rate, enhanced reliability, reduced latency, improved energy efficiency, improved spectrum efficiency and reduced interference.

In performing massive MIMO system identification, a linear system was considered, but in most cases, real life systems are nonlinear in nature. As a way of making the massive MIMO system mimic a real-life communication system, it would be of great importance to model it using nonlinear models and then perform system identification on the nonlinear system using the integer-order and fractional-order system identifications models.

In addressing the issue of IQ imbalance in the massive MIMO system we assumed perfect channel state information, i.e. the conditions of the channel are known, but this is not the case. In reality the conditions of the channel have to be estimated and having performed channel estimation/ system identification using the integer-order and fractional-order MOESP algorithms, it would be of great interest to extend our study on IQ imbalance and IQ imbalance compensation to incorporate a massive MIMO channel that has been identified using these algorithms.

## References

- [1] O. P. Malik, G. S. Hope, and S. J. Cheng, "Some issues on the practical use of recursive least squares identification in self-tuning control," *Int. J. Control*, vol. 53, no. 5, pp. 1021–1033, 1991.
- [2] P. Van Overschee and B. De Moor, "Subspace identification for linear systems, theory - implementation - applications", Kluwer Academic Publishers, 1996.
- [3] H. Garnier and P. C. Young, "The advantages of directly identifying continuous-time transfer function models in practical applications," *Int. J. Control*, vol. 87, no. 7, pp. 1319–1338, 2014.
- [4] G. P. Rao and H. Garnier, "Numerical illustrations of the relevance of direct continuous-time model identification", vol. 35, no. 1. *IFAC*, 2002.
- [5] G. P. Rao and H. Unbehauen, "Identification of continuous-time systems," *IEE Proceedings-Control theory and applications*, vol. 153, no. 2, pp. 185-220, 2006.
- [6] P. C. Young and H. Garnier, "Identification and estimation of continuous-time , data-based mechanistic ( DBM ) models for environmental systems," *Environmental modelling & software*, vol. 21, no. 8, pp.1055-1072, 2006.
- [7] L. Ljung, "System Identification: Theory for the User," Englewood Cliffs, NJ: Prentice-Hall, 1987.
- [8] A. Narang, "Identification and control of fractional and integer order systems", PhD., University of Alberta, Canada, 2012.
- [9] J. Schoukens and L. Ljung, "Nonlinear system identification - a user-oriented road map," *arXiv preprint arXiv:1902.00683*, 2019.

- [10] D. J. Murray-Smith, "Methods of system identification , parameter estimation and optimisation applied to problems of modelling and control in engineering and physiology", DSc (Eng), University of Glasgow, UK, 2009.
- [11] R. Isermann and M. Münchhof, "Identification of dynamic systems - an introduction with applications", Springer, 2011.
- [12] D. Chinarro, "System Engineering Applied to Fuenmayor Karst Aquifer (San Julián de Banzo, Huesca) and Collins Glacier (King George Island, Antarctica)", Springer, 2014.
- [13] G.F. Franklin, J.D. Powell, and M.L. Workman, "Digital control of dynamic systems," vol. 3, Menlo Park, CA: Addison-wesley, 1998.
- [14] T. Söderström, "Identification of linear systems in time domain," 2003.
- [15] R. M. Mokhtar, "Continuous time state-space model identification with application to magnetic bearing systems", PhD., RMIT University, Australia, 2008.
- [16] E. G. Larsson, O. Edfors, F. Tufvesson and T. L. Marzetta , "Massive MIMO for next generation wireless systems" , *IEEE Commun. Mag.*, vol. 52 , no. 2 , pp.186-195 , 2014.
- [17] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, O. Edfors, and F. Tufvesson, "Scaling up MIMO : Opportunities and challenges with very large arrays," *IEEE Signal Process. Mag.*, vol. 30, no. 1, pp. 40–60, 2013.
- [18] K. Hosseini, J. Hoydis, S. Ten Brink, and M. Debbah, "Massive MIMO and small cells: How to densify heterogeneous networks," *IEEE Int. Conf. Commun.*, pp. 5442–5447, 2013.

- [19] H. Q. Ngo, E. G. Larsson, and T. L. Marzetta, "Energy and spectral efficiency of very large multiuser MIMO systems," *IEEE Trans. Commun.*, vol. 61, no. 4, pp. 1436–1449, 2013.
- [20] X. Gao, F. Tufvesson, O. Edfors and F. Rusek , "Measured propagation characteristics for very-large MIMO at 2.6 GHz" , *Proc. 46th ASILOMAR*, pp.295 -299 , 2012.
- [21] K. Li, X. Song, M. Omair Ahmad, and M. N. S. Swamy, "An improved multicell MMSE channel estimation in a massive MIMO system," *Int. J. Antennas Propag.*, 2014.
- [22] H. Q. Ngo and E. G. Larsson, "EVD-based channel estimation in multicell multiuser MIMO systems with very large antenna arrays," *Acoust. Speech Signal Process. (ICASSP), 2012 IEEE Int. Conf.*, no. 1, pp. 3249–3252, 2012.
- [23] B. Li, D. Zhu, P. Liang, "Small cell in-band wireless backhaul in massive MIMO systems: A cooperation of next-generation techniques", *IEEE Trans. Wireless Commun.*, vol. 14, no. 12, pp. 7057-7069, Dec. 2015.
- [24] W. Liu, S. Han, C. Yang, and C. Sun, "Massive MIMO or small cell network: Who is more energy efficient?," *IEEE Wirel. Commun. Netw. Conf. Work. WCNCW 2013*, pp. 24–29, 2013.
- [25] H. Q. Ngo, E. G. Larsson, and T. L. Marzetta, "Massive MU-MIMO downlink TDD systems with linear precoding and downlink pilots," *51st Annu. Allert. Conf. Commun. Control. Comput. Allert. 2013*, pp. 293–298, 2013.
- [26] E. Bjornson, J. Hoydis, M. Kountouris, and M. Debbah, "Massive MIMO systems with non-ideal hardware: Energy efficiency, estimation, and capacity limits," *IEEE Trans. Inf. Theory*, vol. 60, no. 11, pp. 7112–7139, 2014.

- [27] N. Kolomvakis, M. Matthaiou, J. Li, M. Coldrey, and T. Svensson, "Massive MIMO with IQ Imbalance: Performance Analysis and Compensation," *Proc. IEEE ICC*, 2015.
- [28] R. Chen, Q. Gao, H. Li, R. Tamrakar, and S. Sun, "Practical channel acquisition for massive MIMO systems in LTE," in *2015 International Conference on Computing, Networking and Communications, ICNC 2015*, 2015, pp. 47–51.
- [29] J. Vieira, F. Rusek, and F. Tufvesson, "Reciprocity calibration methods for massive MIMO based on antenna coupling," in *IEEE Global Communications Conference*, pp. 3708-3712, December 2014.
- [30] A. Ashikhmin and T. Marzetta, "Pilot contamination precoding in multi-cell large scale antenna systems," *IEEE Int. Symp. Inf. Theory - Proc.*, pp. 1137–1141, 2012.
- [31] J. H. Sørensen, E. De Carvalho, and P. Popovski, "Massive MIMO for crowd scenarios: A solution based on random access," *Globecom Workshops (GC Wkshps)*, Dec 2014, pp. 352–357.
- [32] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin and R. Zhang, "An overview of massive MIMO: Benefits and challenges" , *IEEE J. Sel. Topics Signal Process.*, vol. 8 , no. 5 , pp.742 -758 , 2014.
- [33] B. Gopalakrishnan and N. Jindal, "An analysis of pilot contamination on multi-user MIMO cellular systems with many antennas," *IEEE Work. Signal Process. Adv. Wirel. Commun. SPAWC*, pp. 381–385, 2011.
- [34] Z. Wang, C. Qian, L. Dai, J. Chen, C. Sun, and S. Chen, "Location-based channel estimation and pilot assignment for massive MIMO systems," *2015 IEEE Int. Conf. Commun. Work. ICCW 2015*, pp. 1264–1268, 2015.

- [35] R. Mueller, L. Cottatellucci, and M. Vehkaperä, “Blind pilot decontamination,” *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp.192–201, 2014.
- [36] V. Saxena, G. Fodor, and E. Karipidis, “Mitigating pilot contamination by pilot reuse and power control schemes for massive MIMO systems,” in Proc. IEEE VTC-Spring, May 2015.
- [37] A. Kazerouni, “Pilot Contamination in Small-Cell Massive MIMO Systems,” EE360 Project Final Report, Winter 2014, Stanford University.
- [38] E. Björnson, P. Zetterberg, M. Bengtsson and B. Ottersten, "Capacity limits and multiplexing gains of MIMO channels with transceiver impairments", *IEEE Commun. Lett.*, vol. 17, no. 1, pp.91–94, 2013.
- [39] R. Krishnan, M. Reza Khazadi, N. Krishnan, A. Graell, I Amat, T. Eriksson, N. Mazzali and G. Colavolpe, "On the impact of oscillator phase noise on the uplink performance in a massive MIMO-OFDM system", *IEEE Signal Process. Lett.*, 2014.
- [40] S. Zarei, W. Gerstacker, J. Aulin, and R. Schober, “I/Q Imbalance Aware Widely-Linear Receiver for Uplink Multi-Cell Massive MIMO Systems”, *IEEE Transactions on Wireless Communications*, vol. 15, no. 5, pp. 3393–3408, 2016.
- [41] X. Zhang, M. Matthaiou, E. Bjornson, M. Coldrey, and M. Debbah, “On the MIMO capacity with residual transceiver hardware impairments,” *2014 IEEE Int. Conf. Commun. ICC 2014*, pp. 5299–5305, 2014.
- [42] A. Hakkarainen, J. Werner, K. R. Dandekar, and M. Valkama, “Widely-linear beamforming and RF impairment suppression in massive antenna arrays,” *J. Commun. Networks*, vol. 15, no. 4, pp. 383–397, 2013.
- [43] B. Razavi, “Design considerations for direct-conversion receivers,” *IEEE Trans. Circuits Syst. II Analog Digit. Signal Process.*, vol. 44, no. 6, pp. 428–435, 1997.

- [44] H. Oka, C. Ahn, T. Omori, and K. Hashimoto, "IQ imbalance estimation and compensation schemes based on time frequency interferometry for OFDM," *Proc. of IEEE IWSDA'13*, pp.126-129, November 2013.
- [45] G. Fettweis, M. Löhning, D. Petrovic, M. Windisch, P. Zillmann, and W. Rave, "Dirty RF: A new paradigm," *Int. J. Wirel. Inf. Networks*, vol. 14, no. 2, pp. 133–148, 2007.
- [46] A. Tarighat, R. Bagheri, and A. H. Sayed, "Compensation schemes and performance analysis of IQ imbalances in OFDM receivers," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 3257–3268, 2005.
- [47] J. Kaspar, S. A. Choudry, D. Landgrebe, and M. Vielhaber, "Concurrent and geometry-dependent selection of material and joining technology - An initial utility-based systematic decision-making tool," *12th Annu. IEEE Int. Syst. Conf. SysCon 2018 - Proc.*, pp. 1–8, 2018.
- [48] Y. Yoshida, K. Hayashi, H. Sakai, and W. Bocquet, "Analysis and compensation of transmitter IQ imbalances in OFDMA and SC-FDMA systems," *IEEE Trans. Signal Process.*, vol. 57, no. 8, pp. 3119–3129, 2009.
- [49] National Instruments, "Sources of Error in IQ Based RF Signal Generation," Jan. 2007. Available: <http://www.ni.com/tutorial/5657/en/>.
- [50] J. Qi, S. Member, S. Aïssa, and S. Member, "Analysis and compensation of I/Q imbalance in amplify-and-forward cooperative systems," *Proc. 2012 IEEE Wireless Commun. Netw. Conf.*, pp.215-220.
- [51] T.C.W. Schenk, P.F.M. Smulders, and E.R. Fledderus, "Estimation and compensation of TX and RX IQ imbalance in OFDM based MIMO systems," *Proc. IEEE Radio and Wireless Symposium (RWS 2006)*, Jan. 2006, pp. 215-218.

- [52] E. Cetin, I. Kale, and R.C. Morling, "Joint compensation of IQ-imbalance and carrier phase synchronization errors in communication receivers," in *IEEE International Symposium on Circuits and Systems*, pp. 4481-4484, May 2005.
- [53] A. Pitarokoilis , S. K. Mohammed and E. G. Larsson, "Uplink performance of time-reversal MRC in massive MIMO systems subject to phase noise", *IEEETrans. Wireless Commun.*, 2013.
- [54] R. Krishnan, M. R. Khanzadi, N. Krishnan, Y. Wu, A. Graell i Amat, T. Eriksson and R. Schober, "Linear massive MIMO precoders in the presence of phase noise: A large-scale analysis", *IEEE Trans. Veh. Technol.*, vol. 65, no. 5 pp. 3057-3071, 2015.
- [55] T. Pollet, M. Van Bladel, and M. Moeneclaey, "BER sensitivity of OFDM systems to carrier frequency offset and Weiner phase noise", *IEEE Trans. Commun.*, vol. 43, pp.191 -193, 1995.
- [56] S. Wu and Y. Bar-Ness, "OFDM systems in the presence of phase noise: Consequences and solutions," *IEEE Trans. Commun.*, vol. 52, no. 11, pp. 1988–1996, 2004.
- [57] D. Petrovic, W. Rave, and G. Fettweis, "Effects of phase noise on OFDM systems with and without PLL: Characterization and compensation," *IEEE Trans. Commun.*, vol. 55, no. 8, pp. 1607–1616, 2007.
- [58] S. Jnawali, S. Beygi, and H. R. Bahrami, "A low-complexity method to compensate IQ-imbalance and phase noise in MIMO-OFDM systems," *GLOBECOM - IEEE Glob. Telecommun. Conf.*, pp. 1–6, 2011.
- [59] J. Shen, S. Suyama, T. Obara, and Y. Okumura, "Requirements of power amplifier on super high bit rate massive MIMO OFDM transmission using higher frequency bands," *2014 IEEE Globecom Work. GC Wkshps 2014*, pp. 433–437, 2014.

- [60] X. Li, "State Space Estimation of Wireless Fading Channels", MSc., University of Ottawa, Canada, 2002.
- [61] C. Zhang, R. R. Bitmead, "State space modeling for MIMO wireless channels", *Proc. IEEE Int. Conf. Commun. (ICC)*, pp. 2297-2301, May 2005.
- [62] T. Bastogne, H. Garnier, and P. Sibille, "A PMF-based subspace method for continuous-time model identification. Application to a multivariable winding process," *Int. J. Control*, vol. 74, no. 2, pp. 118–132, 2001.
- [63] P. Young, "Parameter estimation for continuous-time models - a survey," *Automatica*, vol. 17, no. 1, pp. 23–39, 1981.
- [64] P. Van Overschee and B. De Moor, "Continuous-time frequency domain subspace system identification," *Signal Processing*, vol. 52, no. 4, pp. 179–194, 1996.
- [65] S. Hachicha, M. Kharrat, and A. Chaari, "N4SID and MOESP algorithms to highlight the ill-conditioning into subspace identification," *Int. J. Autom. Comput.*, vol. 11, no. 1, pp. 30–38, 2014.
- [66] P. Trnka, "Subspace identification methods - technical report", Czech Technical University, Prague, 2005.
- [67] B. Jiang, X. Zhu, D. Huang, and R. D. Braatz, "Canonical variate analysis-based monitoring of process correlation structure using causal feature representation," *J. Process Control*, vol. 32, pp. 109–116, 2015.
- [68] "R. Shi, "Subspace identification methods for process dynamic modeling", PhD., McMaster University, Canada, 2001.

- [69] R. Malti, and M. Thomassin, M, “Differentiation similarities in fractional pseudo-state space representations and the subspace-based methods,” *Fractional Calculus and Applied Analysis*, vol. 16, no. 1, pp.273-287, 2013.
- [70] T. Bastogne, H. Garnier, P. Sibille, and M. Mensler, “PMF-Based Subspace Method for Continuous-Time Model Identification Deterministic Study,” *IFAC Proc. Vol.*, vol. 30, no. 11, pp. 1561–1566, 1997.
- [71] Y. Hu, Y. Fan, Y. Wei, Y. Wang, and Q. Liang, “Subspace-based continuous-time identification of fractional order systems from non-uniformly sampled data,” *Int. J. Syst. Sci.*, vol. 47, no. 1, pp. 122–134, 2016.
- [72] M. Viberg, B. Wahlberg, and B. Ottersten, B.,”Analysis of state space system identification methods based on instrumental variables and subspace fitting,” *Automatica*, vol. 33, no. 9, pp.1603-1616, 1997.
- [73] T. Katayama, “Subspace methods for system identification,” Springer Science & Business Media, 2006.
- [74] M. Verhaegen, "Identification of the deterministic part of MIMO state space models given in innovations form from input-output data," *Automatica*, vol. 30, no. 1, pp.61-74, 1994.
- [75] Z. Li, “Fractional order modeling and control of multi-input-multi-output processes”, PhD., University of California, Merced, USA, 2015.
- [76] M. Ö. Efe, “Fractional order sliding mode control with reaching law approach,” *Turkish J. Electr. Eng. Comput. Sci.*, vol. 18, no. 5, pp. 731–747, 2010.
- [77] U. Siddique and O. Hasan, “Formal analysis of fractional order systems in HOL,” *Proceedings of the International Conference on Formal Methods in Computer-Aided Design*, pp. 163-170, 2011.
- [78] M. Kupper, I. S. Gil, and S. Hohmann, “Distributed and decentralized state

- estimation of Fractional order systems,” *Proc. Am. Control Conf.*, pp. 2765–2771, July 2016.
- [79] J. Sabatier, M. Merveillaut, J. M. Francisco, F. Guillemard, and D. Porcelatto, “Fractional models for lithium-ion batteries,” *Eur. Control Conf.*, pp. 3458–3463, 2013.
- [80] J.-C. Trigeassou, T. Poinot, A. Oustaloup, and F. Levron, “Modeling and Identification of A Non Integer Order System,” *European Control Conference*. pp. 2453–2458, 1999.
- [81] G. E. Carlson, C. A. Halijak, "Approximation of fractional capacitors  $(1/s)(1/n)$  by a regular Newton process", *IEEE Trans. Circuit Theory*, pp. 210-213, June 1964
- .
- [82] A. Charef, H. H. Sun, Y. Y. Tsao, and B. Onaral, “Fractal system as represented by singularity function,” *IEEE Transactions on Automatic Control*, vol. 37, no. 9. pp. 1465–1470, 1992.
- [83] S. Das, and I. Pan, "Fractional order signal processing: introductory concepts and applications", Springer Science & Business Media, 2011, ch. 2, pp. 13-30.
- [84] S. Das, and I. Pan, "Fractional order signal processing: introductory concepts and applications", Springer Science & Business Media, 2011.
- [85] C. A. Monje, Y. Chen, B. M. Vinagre, D. Xue and V. Feliu, “Fractional-order systems and control: fundamentals and applications”, Springer, 2010.
- [86] D. Matignon, “Stability properties for generalized fractional differential systems,” *ESAIM Proc.*, vol. 5, pp. 145–158, 1998.

- [87] S. Hadjiloucas and R. K. H. Galvão, “Fractional order and non-linear system identification algorithms for biomedical applications,” *J. Phys. Conf. Ser.*, vol. 490, no. 1, 2014.
- [88] S. Guermah, S. Djennoune, and M. Bettayeb, “Discrete-Time Fractional-Order Systems: Modeling and Stability Issues,” *Adv. Discret. Time Syst.*, pp. 183–212, 2012.
- [89] M. Thomassin, R. Malti, “Multivariable identification of continuous-time fractional systems”, *Proceedings of the ASME IDETC/CIE Conference*, San Diego, CA, USA, August 30–September 2, 2009, pp. 1187-1195.
- [90] I. Batiha, R. El-Khazali, A. AlSaedi, and S. Momani, “The General Solution of Singular Fractional-Order Linear Time-Invariant Continuous Systems with Regular Pencils,” *Entropy*, vol. 20, no. 6, p. 400, 2018.
- [91] M. Aoun, R. Malti, F. Levron, and A. Oustaloup, “Numerical simulations of fractional systems: An overview of existing methods and improvements,” *Nonlinear Dyn.*, vol. 38, no. 1–4, pp. 117–131, 2004.
- [92] D. Xue, “Fractional-order control systems: fundamentals and numerical implementations”, Walter de Gruyter GmbH & Co KG., 2017, ch. 5, pp. 141-178.
- [93] B. M. Vinagre, I. Podlubny, A. Hernández, and V. Feliu, "Some approximations of fractional order operators used in control theory and applications", *Fractional Calculus & Applied Analysis*, vol. 3, no. 3, 2000, pp. 231-248.
- [94] M. S. Tavazoei, M. Siami, M. Haeri, and S. Bolouki, “Stability Preservation Problem In The Methods That Find Rational Approximation Of Fractional,” *6th EUROMECH Nonlinear Dyn. Conf.*, 2008.

- [95] D.Y. Xue, and Y.Q. Chen, "Sub-optimum H rational approximations to fractional order linear systems," *Proc. of the ASME 2005 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, Long Beach, California, USA, 2005.
- [96] E.W Stacy, "A generalization of the gamma distribution," *The Annals of mathematical statistics*, vol. 33, no. 3 pp. 1187-1192, 1962.
- [97] D. Xue, "Fractional-order control systems: fundamentals and numerical implementations", Walter de Gruyter GmbH & Co KG., 2017, ch. 2, pp. 9-50.
- [98] R.N. Pillai, "On Mittag-Leffler functions and related distributions," *Annals of the Institute of statistical Mathematics*, vol. 42, no. 1, pp. 157-161, 1990.
- [99] Prabhakar, "A singular integral equation with a generalized Mittag Leffler function in the kernel," 1969.
- [100] I. Podlubny, "Fractional-Order Systems and PI Controllers," vol. 44, no. 1, pp. 208–214, 1999.
- [101] L. Zeng, P. Cheng, and W. Yong, "Using Instrumental Variables," *Proc. 30th Chinese Control Conf.*, pp. 1636–1640, 2011.
- [102] L. Dorcak, I Petras, I Kostial, and J. Terpak, "Fractional-order state space models," *arXiv preprint math/0204193*, 2002.
- [103] L. Debnath, "A brief historical introduction to fractional calculus," *Int. J. Math. Educ. Sci. Technol.*, vol. 35, no. 4, pp. 487–501, 2004.
- [104] D. Xue, "Fractional-order control systems: fundamentals and numerical implementations", Walter de Gruyter GmbH & Co KG., 2017, ch. 7, pp. 221-241.

- [105] D. Xue, "Fractional-order control systems: fundamentals and numerical implementations", Walter de Gruyter GmbH & Co KG., 2017, ch. 3, pp. 51-97.
- [106] C. Zhang and R. R. Bitmead, "MIMO equalization with state-space channel models," *IEEE Trans. Signal Process.*, vol. 56, no. 10, pp. 5222–5231, 2008.
- [107] C. Zhang and R. R. Bitmead, "Robust MIMO channel equalization with state-space channel models," *GLOBECOM - IEEE Glob. Telecommun. Conf.*, 2006.
- [108] N. Al-Dhahir, A. H. Sayed, "The finite-length multi-input multi-output MMSE-DFE", *IEEE Trans. Signal Processing*, pp. 2921-2936, Oct. 2000.
- [109] <http://www.telecomhall.com/what-is-isi-inter-symbol-interference-in-lte.aspx>. Accessed 02<sup>nd</sup> September 2019.
- [110] H. Wei, Y. Huang, J. Du, and L. Li, "A universal MMSE-DFE equalizer with its application to WLAN receiver", *Wireless Personal Communications*, vol. 85, no. 4, pp. 2507-2518, 2015.
- [111] A. Agarwal, S. N. Sur, A. K. Singh, H. Gurung, A. K. Gupta, and R. Bera, "Performance analysis of linear and non-linear equalizer in Rician channel", *C3IT-2012*, vol. 4, pp. 687-691, 2012.
- [112] S. Karthikeyan, S. Sasikumar, M. Madheswaran, and S. Sophia. "Comparison of linear and non-linear multiuser detectors with quantum constraints." *Computer Science & Telecommunications*, vol. 26, no. 3, 2010.
- [113] [http://www.sharetechnote.com/html/Communication\\_ChannelModel\\_MMSE.html](http://www.sharetechnote.com/html/Communication_ChannelModel_MMSE.html). Accessed 02<sup>nd</sup> September 2019.
- [114] <http://barry.ece.gatech.edu/6603/handouts/eq.pdf>. Accessed 02<sup>nd</sup> September 2019.

- [115] I. Barhumi, G. Leus, and M. Moonen, "Time Varying FIR Equalization of Doubly Selective Channels," *IEEE Trans. Wireless Commun.*, vol. 4, pp. 202-214, Jan. 2005.
- [116] [http://homepage.ntu.edu.tw/~ihwang/Teaching/Fa17/LectureNotes/Lecture05\\_v0.pdf](http://homepage.ntu.edu.tw/~ihwang/Teaching/Fa17/LectureNotes/Lecture05_v0.pdf). Accessed 02<sup>nd</sup> September 2019.
- [117] Y. Jiang, M. K. Varanasi, and J. Li, "Performance Analysis of ZF and MMSE Equalizers for MIMO Systems - An In-Depth Study of the High SNR Regime," *Inf. Theory, IEEE Trans.*, vol. 57, no. 4, pp. 2008–2026, 2011.
- [118] T. Ahmed, N. Uddin, M. Rahaman, "Performance analysis of zero forcing equalizer in  $2 \times 2$  and  $3 \times 3$  MIMO wireless channel", *Global Journal of Researches in Engineering: Electrical and Electronics Engineering* vol. 14, no. 9, 2014.
- [119] M. L. Ammari, K. Zaouali, and P. Fortier, "Adaptive modulation and decision feedback equalization for frequency-selective MIMO channels," *International Journal of Communication Systems*, vol. 2013.
- [120] A. Stamoulis, G. B. Giannakis, and A. Scaglione, "Block FIR decision-feedback equalizers for filterbank precoded transmissions with blind channel estimation capabilities," *IEEE Trans. Commun.*, vol. 49, pp. 69–83, Jan. 2001.
- [121] <https://www.quora.com/How-does-a-Decision-Feedback-Equalizer-DFE-work>. Accessed 02<sup>nd</sup> September 2019.
- [122] S. Senol, "Performance comparison of adaptive decision feedback equalizer and blind decision feedback equalizer ", MSc., The Middle East Technical University, 2004

- [123] N. Al-Dhahir, J. M. Cioffi, "MMSE decision feedback equalizers: finite-length results", *IEEE Trans. Inform. Theory*, vol. 41, pp. 961-976, July 1995.
- [124] X. Xia, "New precoding for intersymbol interference cancellation using nonmaximally decimated multirate filterbanks with ideal FIR equalizers", *IEEE Trans. Signal Processing*, pp. 2431-2441, Oct. 1997.
- [125] D.D. Falconer, and G.J. Foschini, "Theory of minimum mean-square-error QAM systems employing decision feedback equalization," *Bell System Technical Journal*, vol. 52, no. 10, pp.1821-1849, 1973.
- [126] J.M. Cioffi, "379A Class note: Signal Processing and detection," 2008.
- [127] R. López-Valcarce, "Realizable linear and decision feedback equalizers : properties and connections," *IEEE Transactions on Signal Processing*, vol. 52, no. 3, pp. 757–773, 2004.
- [128] A. Coskun and I. Kale, "Blind multidimensional matched filtering techniques for single input multiple output communications," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 5, pp. 1056–1064, 2010.
- [129] A. Coskun and I. Kale, "Blind correlation-based DFE receiver for the equalization of single input multi output communication channels," in *Wireless Telecommunications Symposium*, pp. 1-5. IEEE, 2009.
- [130] W. S. Leon, "Equalization and estimation for fading channels", PhD., University of Canterbury, New Zealand, 2003.
- [131] A. Coskun and I. Kale, "Multi-dimensional matched filter identification technique for channel equalization deployed in spatial diversity receivers." in *Ph. D. Research in Microelectronics and Electronics*, pp. 36-39. IEEE, 2009.

- [132] A. Scaglione, G. B. Giannakis, and S. Barbarossa, "Redundant Filterbank Precoders and Equalizers Part I: Unification and Optimal Designs," *IEEE Transactions on Signal Processing*, vol. 47, no. 7, pp. 1988–2006, 1999.
- [133] P. Taylor, C. L. Law, and T. H. Teo, "Error performance of narrowband and spread-spectrum modulation with diversity in frequency-selective channel," *International journal of electronics*, vol. 79, no. 4, pp. 421-438, 1995.
- [134] M. K. Simon and M. Alouini, "Digital Communication over Fading Channels," John Wiley & Sons, 2005.
- [135] V. Meghdadi, "BER calculation," in *Wireless Communications*, edited by: Goldsmith A. 2008.
- [136] I. H. Lee, "SER Performance of Enhanced Spatial Multiplexing Codes with ZF / MRC Receiver in Time-Varying Rayleigh Fading Channels," *The Scientific World Journal*, 2014.
- [137] A. C. Fannjiang and P. Yan, "Multi-frequency imaging of multiple targets in Rician fading channels : stability and resolution," *Inverse Problems*, vol. 23, no. 5, p.1801, 2007.
- [138] R.Y. Yen, H.Y. Liu, K.F. Yang, M.C. Hong, and S.C. Nan, S. C. (2007, July), "Performance of OFDM QAM over frequency-selective fading channels," in *IFIP International Conference on Wireless and Optical Communications Networks*, pp. 1-5, IEEE, July 2007.
- [139] L. Carassale, and A. Kareem, "Modeling nonlinear systems by Volterra series," *Journal of engineering mechanics*, vol. 136, no. 6, pp. 801-818, 2009.
- [140] S. Boyd, and L. Chua, L., "Fading memory and the problem of approximating nonlinear operators with Volterra series," *IEEE Transactions on circuits and systems*, vol. 32, no. 11, pp.1150-1161, 1985.

- [141] W. J. Rugh, "Nonlinear System Theory," Baltimore, MD: Johns Hopkins University Press, 1981.
- [142] C. Samakee and P. Phukpattaranont, "Application of MISO Volterra series for modeling subharmonic of ultrasound contrast agent," *International Journal of Computer and Electrical Engineering*, vol. 4, no. 4, p.445, 2012.
- [143] M. Schetzen, "The Volterra and Wiener theories of nonlinear systems." 1980.
- [144] S.A. Billings, "Identification of nonlinear systems—a survey." in *IEE Proceedings D (Control Theory and Applications)*, vol. 127, no. 6, pp. 272-285. IET Digital Library, 1980.
- [145] S. A. Billings, "Nonlinear system identification: NARMAX methods in the time, frequency, and spatio-temporal domains," John Wiley & Sons, 2013.
- [146] E. Eskinat, S.H. Johnson, and W.L Luyben, "Use of Hammerstein models in identification of nonlinear systems," *AIChE Journal*, vol. 37, no. 2, pp. 255-268, 1991.
- [147] R. Abbasi-Asl, R. Khorsandi, B. Vosooghi-Vahda, "Hammerstein-Wiener model: a new approach to the estimation of formal neural information," *Basic and Clinical Neuroscience* vol. 3, no. 4, pp. 45–51, 2012.
- [148] S. A. Billings, and I J. Leontaritis "Parameter estimation techniques for nonlinear systems," *IFAC Proc.*, vol. 15, no. 4, pp. 505–510, 1982.
- [149] G. Zito and I. D. Landau, "A methodology for identification of NARMAX models applied to diesel engines," *IFAC Proceedings*, vol. 38, no. 1, pp. 374-379, 2005.

- [150] S. Chen, and S. A. Billings, "Representations of non-linear systems: the NARMAX model," *International Journal of Control*, vol. 49, no. 3, pp.1013-1032, 1989.
- [151] Y. Gao and M. J. Er, "NARMAX time series model prediction : feedforward and recurrent fuzzy neural network approaches," *Fuzzy sets and systems*, vol. 150, no. 2, pp.331-350, 2005.
- [152] W. S. Sarle, "Neural networks and statistical models," 1994.
- [153] S. Saha, S. Das, A. Acharya, A. Kumar, S. Mukherjee, I. Pan, and A. Gupta, "Identification of nonlinear systems from the knowledge around different operating conditions: A feed-forward multi-layer ANN based approach," in *2nd IEEE International Conference on Parallel, Distributed and Grid Computing*, pp. 413-418, 2012.
- [154] J. E. Sohl and A. R. Venkatachalam, "A neural network approach to forecasting model selection," *Information & Management*, vol. 29, no. 6, pp. 297–303, 1995.
- [155] <http://www.ni.com/white-paper/4028/en/>. Accessed 02<sup>nd</sup> September 2019.
- [156] L. Ljung, "System identification," *Wiley Encyclopedia of Electrical and Electronics Engineering*, 2001.
- [157] <http://zone.ni.com/reference/en-XX/help/372458D-01/lvsysidconcepts/modeldefinitionsarx/>. Accessed 02<sup>nd</sup> September 2019.
- [158] <http://zone.ni.com/reference/en-XX/help/372458D-01/lvsysidconcepts/modeldefinitionsarmax/>. Accessed 02<sup>nd</sup> September 2019.
- [159] <http://zone.ni.com/reference/en-XX/help/371894H-01/lvsysidconcepts/modeldefinitionsbj/>. Accessed 02<sup>nd</sup> September 2019.

- [160] L. Freidovich, “Elements of iterative learning and adaptive control - lecture 5,” Umeå University, 15<sup>th</sup> April 2010.
- [161] M. A. Dahleh, “System identification - lecture 4”, Massachusetts Institute of Technology.

# Appendix A

## Nonlinear Models

### Volterra series models

The Volterra series is similar to the Taylor series, but unlike the Taylor series, the Volterra series has memory [139]. It has been shown that in most cases any time-invariant, causal, nonlinear system with fading memory can be represented by a finite Volterra series [140]. Considering a continuous time-invariant nonlinear systems, the Volterra series expresses the relationship between an input signal  $u(t)$  and its corresponding output signal  $y(t)$  as a sum of multiple convolutions [141] and [142]:

$$y(t) = h_0 + \sum_{n=1}^N \int_a^b \cdots \int_a^b h_n(\tau_1, \dots, \tau_n) \prod_{j=1}^n u(t - \tau_j) d\tau_j \quad (\text{A.1})$$

where  $h_n(\tau_1, \dots, \tau_n)$  is the  $n^{\text{th}}$  order Volterra kernel and  $\tau_n$  is the system memory. If  $N$  is finite, the series is said to be truncated.

The representation of the causal discrete-time Volterra series can be expressed as [143]:

$$y(n) = h_0 + \sum_{p=1}^P \sum_{\tau_1=a}^b \cdots \sum_{\tau_p=a}^b h_p(\tau_1, \dots, \tau_p) \prod_{j=1}^p u(n - \tau_j) \quad (\text{A.2})$$

where  $h_p(\tau_1, \dots, \tau_p)$  are called the discrete-time kernels. If  $P$  is finite, the series operator is said to be truncated.

A Volterra system is completely defined by its constant output and its Volterra kernel coefficients, i.e.  $h_n(\tau_1, \dots, \tau_n)$  or  $h_p(\tau_1, \dots, \tau_p)$  and a common approach is to estimate the Volterra kernel coefficients by an expansion of orthogonal functions [144].

## Block structured models

Block structured models result from the interconnection of linear dynamic elements and nonlinear static elements. Block structured models can be used to simplify the Volterra series models. The fundamental idea of block structured models is to identify the individual blocks within the system based only on the external input-output data [145]. The commonly known block structured models are the Hammerstein model and Wiener model. A combination of these models also exists, resulting in the, Wiener-Hammerstein model and Hammerstein-Wiener model. Figure A.1 shows the Hammerstein model, where a static nonlinearity,  $g(\cdot)$  is followed by linear dynamics,  $H(\cdot)$ , and  $u$  and  $y$  are the input and output respectively [146]. The Wiener model consists of a linear system followed by a static nonlinearity as show in Figure A.2. The Wiener model is the reverse of the Hammerstein model [145]. The Wiener-Hammerstein model consists of a static nonlinear element sandwiched between two dynamic linear elements as shown in Figure A.3, and the Hammerstein-Wiener model consists of one or two static nonlinear blocks in series with a linear block [147].

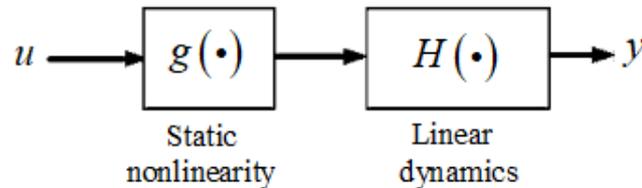


Figure A. 1: Hammerstein model

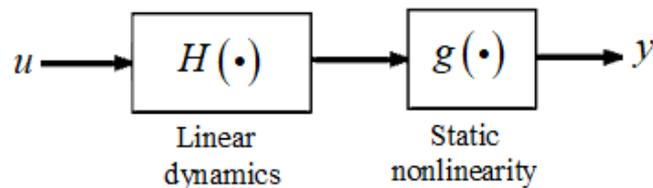


Figure A. 2: Wiener model

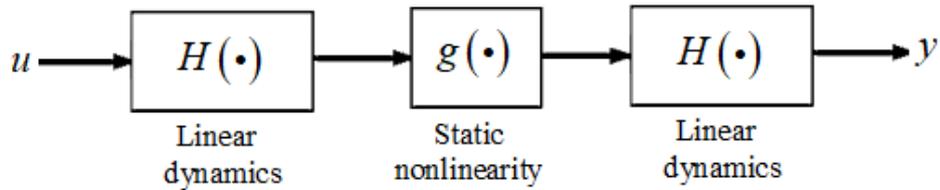


Figure A. 3: The Wiener-Hammerstein model

### The nonlinear autoregressive moving average with exogenous input model

The nonlinear autoregressive moving average with exogenous input (NARMAX) model is an extension of the ARMAX model which is used when dealing with nonlinear systems. The advantage of the NARMAX model is that it can perform system identification with fewer parameters compared to the Volterra series model[145]. The nonlinear autoregressive moving average with exogenous input (NARMAX) model is defined as [148] :

$$y(t) = F(y(t-1), \dots, y(t-n_y), u(t-\tau), \dots, u(t-n_u), e(t-1), \dots, e(t-n_e)) + e(t) \quad (\text{A.3})$$

where  $u(t)$ ,  $y(t)$  and  $e(t)$  are the system input, output and noise respectively, and  $n_u$ ,  $n_y$  and  $n_e$  are the maximum lags for the system input, output and noise,  $F(\bullet)$  is some nonlinear function and  $\tau$  is a time delay. Different model structures can be used to approximate  $F(\bullet)$  in (A.3) and these may include: polynomial and rational models, neural networks and wavelets [149], [150] and [151].

### Neural network models

Neural network (NN) is some form of machine learning algorithm that tries to mimic the neural structure of the brain where computation takes place through simple processes which are then connected into a large network [152] Neural networks predict the behaviour of the system by learning the relationship between the input-output data sets of the system through an iterative training process [153] If an incorrect prediction is made the system adjusts the weights until an output with a small mean-square error is

obtained. This process involves learning the mathematical model of the system and as a result neural networks can be used for system identification. In neural networks the processing units are arranged in layers and typically, a neural network is arranged in three layers, namely the input layer, hidden layers and output layer as shown in Figure A.4.

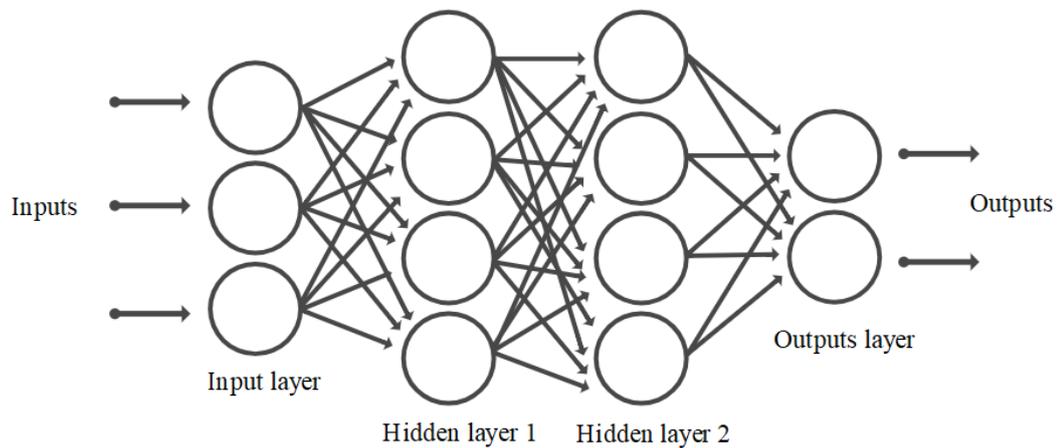


Figure A. 4: Neural networks model

The layers are interconnected through paths called neurons or nodes. Neural networks can be classified as single-layer networks or multi-layer networks. Single-layer networks are the simplest form of neural networks. In single-layer networks there is only one layer of input nodes which directly feeds the single layer of output nodes via a series of weights. Multi-layer networks usually consist of the input layer with a number of nodes, one or more hidden layers and the output layer. In this, each layer of nodes receives inputs from the previous layers [154] The hidden layer and the output layer are responsible for information processing and computation.

## Appendix B

### The General-Linear Polynomial Models

Considering an  $n^{\text{th}}$  order,  $m$  input and  $p$  output state-space linear system, in general, the linear system can be described using the diagram shown in Figure B.1, which is known as the general-linear polynomial model or the general-linear model. The simpler models, i.e. (AR) models, autoregressive with exogenous input (ARX) models, autoregressive moving average with exogenous input (ARMAX) models, Box-Jenkins (BJ) models and output-error (OE) models are a subset of the general-linear polynomial model. According to [155] these models are obtained by setting one or more of  $A(q)$ ,  $B(q)$ ,  $C(q)$  or  $D(q)$  polynomials equal to 1 in Figure B.1.

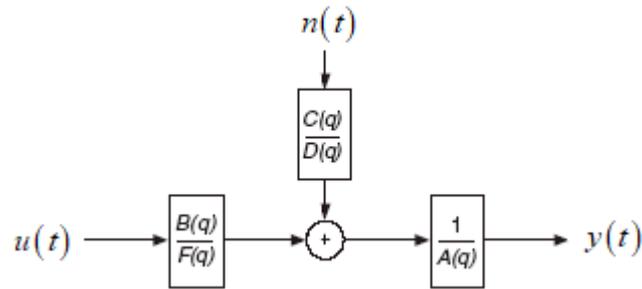


Figure B. 1: Structure of a general-linear model

where  $u(t)$ ,  $y(t)$  and are the input and output of the system respectively and  $n(t)$  is the zero-mean white noise or the disturbance of the system and  $q$  is the delay or shift operator.

The transfer function of Figure B.1 can be expressed as:

$$\frac{y(t)}{u(t)} = G(q) = \frac{b_m q^m + b_{m-1} q^{m-1} + \dots + b_0}{a_n q^n + a_{n-1} q^{n-1} + \dots + a_0} \quad (\text{B.1})$$

where  $a_n, a_{n-1}, \dots, a_0$  and  $b_m, b_{m-1}, \dots, b_0$  are coefficients to be identified.

When using the unit time delay operator,  $q^{-1}$ ,  $A(q)$ ,  $B(q)$ ,  $C(q)$ ,  $D(q)$  and  $F(q)$  can be expressed as [156], [157], [158] and [159]:

$$A(q) = 1 + a_1q^{-1} + \dots + a_nq^{-n} \quad (\text{B.2})$$

$$B(q) = b_0 + b_1q^{-1} + \dots + b_{m-1}q^{-(m-1)} \quad (\text{B.3})$$

$$C(q) = 1 + c_1q^{-1} + \dots + c_pq^{-p} \quad (\text{B.4})$$

$$D(q) = 1 + d_1q^{-1} + \dots + d_{k_d}q^{-k_d} \quad (\text{B.5})$$

$$F(q) = 1 + f_1q^{-1} + \dots + f_{k_f}q^{-k_f} \quad (\text{B.6})$$

where  $k_d$  and  $k_f$  are the  $D$  order and  $F$  order respectively.

### Autoregressive Model

This is a very simple model that is limited in the class of problems it can solve. The Autoregressive (AR) model structure is a process model used in the generation of models where outputs are only dependent on previous outputs. No system inputs or disturbances are used in the modelling. The AR model structure is shown in Figure B.2.

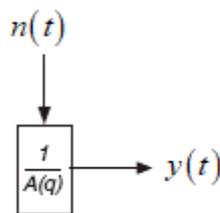


Figure B. 2: Structure of an autoregressive model

### Autoregressive with Exogenous Input Model

The Autoregressive with exogenous input (ARX) model is the simplest model incorporating the input signal. The ARX model is preferable when the model order is high. The disadvantage of the ARX model is that disturbances are part of the system dynamics. The transfer function of the deterministic part of the system and the transfer function of the stochastic part of the system have the same set of poles. This coupling can be unrealistic. The system dynamics and stochastic dynamics of the system do not share the same set of poles all the time. However, you can reduce this disadvantage if you have a good signal-to-noise ratio. When the disturbance,  $n(t)$  of the system is not white noise, the coupling between the deterministic and stochastic dynamics can bias the estimation of the ARX model. Set the model order higher than the actual model order to minimise the equation error, especially when the signal-to-noise ratio is low. However, increasing the model order can change some dynamic characteristics of the model, such as the stability of the model. The ARX model structure is shown in Figure B.3.

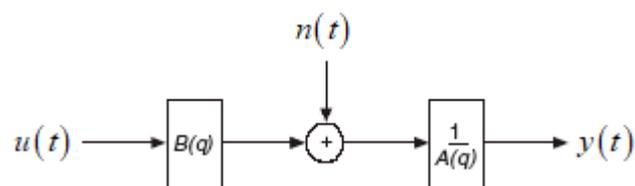


Figure B. 3: Structure of an autoregressive with exogenous input model

### Autoregressive Moving Average with Exogenous Input Model

This model is similar to the ARX model, except that in the autoregressive moving average with exogenous input (ARMAX) model on top of the known inputs the systems is also driven by unmeasured noise. ARMAX models are useful when you have dominating disturbances that enter early in the process, such as at the input. For example, a wind gust affecting an aircraft is a dominating disturbance early in the process. The ARMAX model has more flexibility in the handling of disturbance modelling than the ARX model. The ARMAX model structure is shown in Figure B.4.

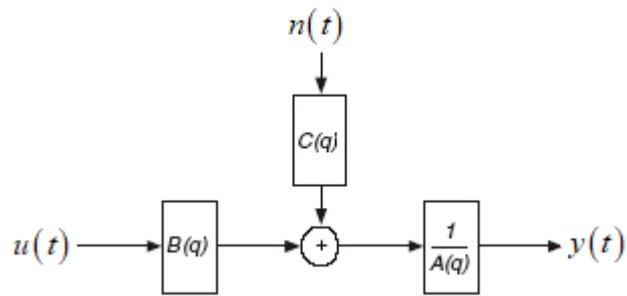


Figure B. 4: Structure of an autoregressive moving average with exogenous input model

### Box-Jenkins Model

The Box-Jenkins (BJ) structure provides a complete model with disturbance properties modelled separately from system dynamics. The Box-Jenkins model is useful when you have disturbances that enter late in the process. For example, measurement noise on the output is a disturbance late in the process. The BJ model structure is shown in Figure B.5.

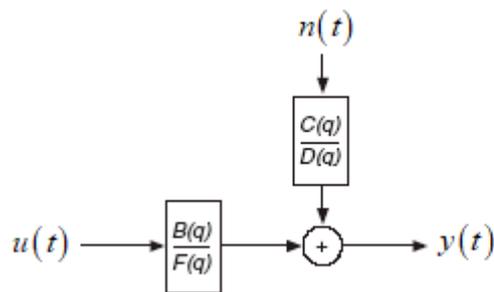


Figure B. 5: Structure of a Box-Jenkins model

### Output-Error Model

The Output-Error (OE) model structure describes the system dynamics separately. No parameters are used for modelling the disturbance characteristics. The OE model structure is shown in Figure B.6.

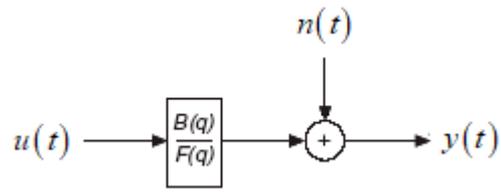


Figure B. 6: Structure of an output-error model

## Appendix C

### Definition of a Persistently Exciting Signal

Given a signal  $u(t)$ , it is said to be persistently exciting of order  $n$  if the limits

$$c(k) = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N u(i)u(i-k), \quad k = 0, 1, \dots, n-1 \quad (\text{C.1})$$

exist [160], and if the matrix  $C_n$  is positive [161], i.e.

$$C_n = \begin{bmatrix} c(0) & c(1) & \dots & c(n-1) \\ c(1) & c(0) & \dots & c(n-2) \\ \vdots & \vdots & \ddots & \vdots \\ c(n-1) & c(n-2) & \dots & c(0) \end{bmatrix} > 0 \quad (\text{C.2})$$