Robust Frameworks for the Observability and Lie Symmetries of Structural Dynamical Systems

Xiaodong Shi

Oriel College

University of Oxford

In Partial Fulfilment of the Requirements for the Degree of

Doctor of Philosophy

Michaelmas 2020

Dedicated to those who believe I can

Yes, I can

Acknowledgements

First of all, I would like to express my deepest appreciation to my supervisors Prof. Manolis Chatzis and Prof. Martin Williams for giving me the chance to pursue my doctoral studies in Oxford, for introducing me to the exciting area of structural dynamics and system identification, for their patient guidance and constant encouragements all along the way in my doctoral journey, and for always providing the perfect balance of independence and advice in my research. It was Prof. Chatzis who led me to look into the observability problem based on which this thesis is developed with fruitful achievements. Prof. Williams's comments, suggestions and questioning can always play as a key for developing and enriching our ideas and research outcomes. They have also helped significantly in my overall development as a researcher through actively involving me in conferences, seminar talks, research collaborations, junior student teaching, etc.

I would like to acknowledge my colleagues and research collaborators, David Burton, Sathvik Dev Velagandula, Xuanli Sun, Emliy Anderson, Wayne Wu, Dr. Kristof Maes, Dr. Anela Bajric-Hodzic and Dr. Sin-Chi Kuok who have provided a lot of technical help and valuable comments and suggestions on my presentations, reports and papers.

A very special thank you should be given to my girlfriend, Qian Ma, who have shared the ups and downs of my studies and life over the last three years. Her uninterrupted support and love make me feel strong when facing any challenges and difficulties. I am also grateful to the warmest friendship brought by Tianning Tang and Yin Liang. Our laughs and happy moments during those Friday nights and holiday journeys are still clearly pictured in front of my eyes. I wish to thank my friend, Penghao Duan. I can always learn a lot from our talks, not only about studies and career development but also about truths of life.

Any attempt to thank my parents, Qimei Chen and Jiyao Shi, would fall short. They have always trusted and encouraged me, and devoted all of their love to me in spite of the space and time differences between us.

Abstract

System identification is an important technique in reconstructing and estimating dynamic states, unknown parameters and unmeasured inputs of dynamical systems using measured input-output signals, and in minimizing the gaps between real engineering systems and their mathematical models. Whether a system for a given setup of sensors can be, in theory, successfully identified is associated with its observability properties.

This thesis is overall devoted to two research directions: 1) developing efficient observability algorithms for handling large and complex dynamical systems and 2) incorporating unmeasured or unknown inputs into robust observability computation and tool. The research is motivated by the need to relax the computational limitation of the existing observability methods that is associated with their high physical memory requirements when used for large and complex real systems, as for example large civil infrastructures encountered in Structural Health Monitoring (SHM). Moreover robust observability computation with the consideration of unmeasured inputs is needed to account for joint state-parameter-input identification problems which have gained increasing attention in recent years.

In particular, two efficient and robust algorithms are proposed to test observability properties in Chapter 2 and Chapter 3. The first algo-

rithm applies to large linear systems with unknown parameters, based on the efficient implementation of the Observability Rank Condition (ORC) method. The second algorithm applies to rational nonlinear systems with unmeasured inputs, based on the extended use of the extended Observability Rank Condition (EORC-DF) and a power series-based computational framework.

In Chapter 4, computational frameworks are developed for Lie symmetries of nonlinear systems with unmeasured inputs. The obtained Lie symmetries can provide an alternative path to approach the observability properties of a system for a given setup of sensors. More importantly, Lie symmetries imply the mathematical relationship between the true solutions of the system's states, parameters and unmeasured inputs and their other possible solutions.

Finally in Chapter 5, the application of observability and Lie symmetry analyses is illustrated through a complex, nonlinear and non-smooth mechanical model. The model is successfully reduced and identified using suitably chosen identification methods.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and motivations

All engineering structures, including not only civil infrastructures such as buildings, bridges, energy plants, wind turbines, roads and rails, but also ships, airplanes, trains and other mechanical systems, are constantly subject to various internal and external factors during their lifetime which adversely affect their health condition. These factors can include, for example, ageing or fatigue during long-term operation, material or geometry deterioration due to environmental impacts, errors and defects induced during construction or manufacturing processes, material imperfections, extreme events or hazards such as an earthquake, strong wind or an explosion, etc [21]. Once damage occurs within a structure, it is important to detect it in an early stage and enable maintenance and strengthening in a proper way. Damage herein can be generally defined as any changes introduced to a system making the system unable to operate in its optimal manner [38]. If such forms of damage are not observed in time, they would accumulate, propagate and eventually lead to a total loss of structural functionality. Catastrophic consequences may be caused by the failure of a structure to the society, including tremendous human life and economic losses.

The process of implementing a damage detection strategy for engineering struc-

Figure 1.1: Structural Health Monitoring (Sydney Harbour Bridge, modified from source: Arcadis [4])

tures is referred to as Structural Health Monitoring (SHM) [34, 41]. The process usually involves the stages of 1) obtaining the measurements of dynamic response of a structure using suitable installations of sensors, 2) extracting damage-related features from the obtained measurements, and 3) statistically analysing these features to detect, locate and quantify the appeared damage or to determine the current health condition of the structure in a global point of view [38]. Among the three stages of SHM process the second stage is normally viewed as the most important and challenging one. One of the means to achieve the goal of accurately 'extracting damage-related features from measured data' is through the use of system identification techniques. The dynamic behaviour of a structure is often described by a physically parameterised model which gives a mathematical relationship between the external excitations applied to the structure (inputs) and the structural response (outputs). A system identification technique can be used to determine the values of the model parameters, which are often direct representations of the current physical properties of the structure, based on the measurements of input-output signals [64].

Those damage-related or damage-sensitive parameters can then be used to proceed to the next stage of SHM. The mathematical model with the determined parameters from system identification, i.e. an updated model of the structure, also plays important roles in performance, vibration control, serviceability and safety analyses of the structure as well as providing predictions of the structural behaviour within for example a future extreme event [114].

Whether a system identification or SHM campaign would be successful depends on various factors, including the model adopted for the structural system, the type, location and quantity of the setup of sensors installed, the quality of measurements, the system identification algorithm used, etc. In theory, this problem is closely related to the observability properties of a structural dynamical system: if a system described by a given mathematical model is observable for a specific sensor setup, the unknown parameters, dynamic states and unmeasured inputs of the system can theoretically be inferred from the input-output measurements; if not, it is then impossible to correctly estimate the values of those quantities which are deemed as unobservable using any system identification algorithm [26]. It has recently been discovered that a system being unobservable implies the existence of Lie symmetries within its model and vice versa [2, 86]. For the unobservable system, an identification algorithm could obtain any of the incorrect sets of values which are admitted by Lie symmetries, and these values may be mistakenly adopted by engineers in practice for the parameters, dynamic states and unmeasured inputs of interest. To avoid such erroneous situations, it is therefore important to conduct a priori observability testing when proposing a system identification campaign. Effective observability and symmetry analyses can provide engineers with guidance on how to design the pattern of sensor installations,

reduce the mathematical model of the system or make suitable model assumptions in order to yield an observable system.

In the following Section 1.2.1, a brief literature review is provided for various system identification methods and algorithms especially those have been widely employed for the purpose of SHM. A special concentration is placed on the recent developments of novel filters and observers which allow for reliable joint state-parameter-unmeasured input estimation. The occurrence and usefulness of these filters and observers are the direct motivation of extending the regular observability testing methods to account for unmeasured inputs.

Section 1.2.2 clarifies the concept of observability this thesis is focused on, and provides a detailed review of the existing methods for testing the observability of nonlinear systems with or without unmeasured inputs. Civil structural systems are typically large and also many of them are complex models used to describe sophisticated physical phenomena or materials with special properties. A common weakness of the existing observability testing methods is that they are often not capable of handling large and complex engineering systems due to their significantly high computational expense, which substantially limits the practical applicability of these methods. Although a considerable number of efficient observability algorithms have been developed for the case of all inputs being measured to alleviate this computational limitation to some extent, those algorithms are still unable to meet the requirements of serving for SHM. Moreover, there is a lack of robust implementations of the observability methods to test systems in the presence of unmeasured or unknown inputs. These drawbacks of the existing observability methods bring the primary motivation of this thesis, that is to develop efficient and robust observability algorithms to tackle

large and complex problems with or without the existence of unmeasured inputs.

Section 1.2.3 gives a literature review of Lie symmetries and their computations of nonlinear systems with fully measured inputs. The secondary motivation of the thesis arises from the need to extend the Lie symmetry computational methods for systems with unmeasured inputs.

The end of this chapter aims at describing the main objectives and contributions as well as demonstrating the organisation of this thesis.

1.2 Literature review

1.2.1 System identification

The process of determining the mathematical model of a dynamical system or the parameters of the model based on input-output or output-only signal measurements is known as system identification [63, 64].

During SHM campaigns dynamic tests of structures can be mainly categorized into three types: free, forced and ambient vibration tests [114]. In a free vibration test, the structure remains oscillating after some initial excitations and the measured structural response signals lying within the free vibration phase are selected for processing [5]. Many identification methods and algorithms are developed based on such free vibration output-only measurements. Least square methods were developed to identify model parameters based on least-square minimisation of the discrepancy between measured and theoretical model outputs [5, 28, 111]. Using continuous and discrete wavelet transform analyses of the measured response of structural systems allows for proper estimation of the modal parameters of the systems, i.e. natural frequencies, viscous damping ratios and mode shapes [60, 61, 84]. Other widely used methods

include the Ibrahim time domain method [49], the complex exponential method [20], the Bayesian modal identification using free vibration data [113], etc.

In a forced vibration test, the structure is continuously subject to some artificially applied dynamic excitations, and both the excitation and the structural response signals need to be measured for system identification [32]. Kalman filtering [45, 53, 54, 92] has long been viewed as the optimal state estimator based on input-output measurements. For dynamical systems mathematically described by linear state-space models, Subspace State-Space Identification (N4SID) techniques were proposed to determine the model matrices by deducing state sequences from the projection of input-output measurements which were proved to be the outputs of non-steady state Kalman filter banks [105, 106, 107]. Other methods based on input-output measurements include the circle fitting method [58], the Ewins-Gleeson method [37], the orthogonal polynomial method [104], etc.

In either the free vibration tests or the forced vibration tests, the structures must not be affected by any human-induced loads, and the environmental loads exerted on the structures must be small enough and can hence be properly neglected. Moreover, it is often difficult to control the artificial excitations in practice, as if they are too small the structure can not be sufficiently excited and if they are too large permanent damage may be caused to the structure. In comparison, an ambient vibration test is allowed to be performed during the regular operation of the structure. Any environmental and human-induced excitations applied to the structure are assumed to be statistically random, and only the structural response is measured for system identification. Modal identification based on ambient vibration data is also known as Operational Modal Analysis (OMA) [17]. A typical OMA method is the Fre-

quency Domain Decomposition (FDD) which serves as a simple and robust extension of the Peak Picking method to extract the modal properties of structural systems based on the singular value decomposition of the Power Spectral Density (PSD) matrix [18, 19, 51]. Other OMA methods include the least square complex frequency method [108], the auto-regression moving model [1], the Bayesian OMA methods [6], etc.

One major drawback of the ambient vibration tests is that simply treating those ambient excitations to be statistically random may bring a significant level of errors and uncertainties on the system identification results. On the other hand, accurately measuring the environmental and human-induced loads on the structures are often too difficult, expensive or even impossible in most of practical cases. To further extend the applicability of the existing methods which require precise knowledge or assumptions on inputs, a number of identification algorithms have been recently developed for estimating the states and parameters of nonlinear systems in the presence of unmeasured inputs. These developed algorithms are also possessed of the ability to track those unmeasured inputs or excitations over time. Gillijns and De Moor [43] proposed a filter for joint input-state estimation for systems with direct transmission, which was shown to be globally optimal in the minimum-variance unbiased sense. Based on the preliminary study in Azam et al. [36], Dertimanis et al. [33] introduced a successive Bayesian observer which recombined the dual and Unscented Kalman filters for addressing the joint state-parameter-input identification problem. Other input estimation methods include the augmented Kalman filter [66], the smoothing algorithm [68], the algebraic unknown input observer [8], etc.

Each of the above reviewed identification methods has its own advantage and

strength under a specific SHM scenario, but it should be noted that none of these methods can succeed in correctly estimating the states, parameters and/or unmeasured inputs of a structural system if the system is unobservable for a given sensor setup. It is therefore of crucial importance to perform observability analyses when proposing a vibration test and the identification campaign to check whether the sensor setup used and the structural model assumed could lead to a fully observable system.

1.2.2 Observability

This section aims to clarify the concept of observability this thesis is focused on, and provides a literature review of the existing methods for testing whether a given nonlinear system with a measurement scheme is observable or not.

1.2.2.1 The observability concepts

Strictly speaking, the concepts of observability can be categorized into theoretical or practical, and the concepts of theoretical observability can be further classified into global, local and local weak as first defined by Hermann and Krener [44]. Theoretical observability is an inherent property of a dynamical system depending solely on the mathematical model of the system and the given input-output measurements which are assumed to be noise-free. A theoretically observable system might be detected practically observable or unobservable if more complicated practical factors are taken into consideration, such as how noisy the measurements are, whether the system is modelled accurately, what identification technique is used, etc. A theoretically unobservable system, however, is guaranteed to be practically unobservable [74]. The theoretical observability is basically a global concept, which is based on the assump-

tion that the input-output signals can be measured over an infinitely long time. For a globally observable system in practice, an identification algorithm may need to travel a substantial distance or a very long time in order to determine its unknown parameters, dynamic states and unmeasured inputs. To alleviate this limitation, a stronger concept is introduced as the local observability which ensures that a system, if detected locally observable, can be identified instantaneously using a small period of measurements. Furthermore, in practice it often suffices to be able to identify the unknown quantities of a system from the neighbours of their values, and the concept of local weak observability is therefore introduced. Intuitively, a system is said to be locally weakly observable if one can instantaneously identify the parameters, dynamic states and unmeasured inputs of a system from their neighbours. An additional benefit of using the local weak observability over the other concepts is that it lends itself to a systematic testing method requiring relatively simpler mathematical tools [44].

The contributions of this thesis, if not specifically explained, are focused on the concept of theoretical local weak observability, and for the sake of brevity the remainder of the thesis will keep referring to 'theoretical local weak observability', 'theoretically locally weakly observable' and 'theoretically locally weakly unobservable' simply as 'observability', 'observable' and 'unobservable' respectively. Moreover, it is worth mentioning that the terminology of identifiability is a subcase of observability specifically used for time-invariant parameters, that is, a parameter being observable or unobservable can also be called identifiable or unidentifiable.

1.2.2.2 Observability testing methods

Starting with the linear observability defined by Kalman [55], a large amount of works have been devoted to investigating the observability properties of nonlinear systems

with fully measured inputs. Hermann and Krener [44] introduced the well-known Observability Rank Condition (ORC) to analyze the local weak observability of analytic nonlinear systems whose equations are smooth and infinitely differentiable. Implementation of the ORC is through rank evaluation of the observability matrix constructed from successive Lie derivative computations. A fast convergence algorithm was suggested by Isidori [50] to implement the ORC for analytic systems which are affine in inputs, and Sontag [93] further generalized the ORC algorithm for nonlinear systems analytic in inputs. The theory and implementation of the ORC will be reviewed for more details in Section 2.1 of Chapter 2, as this is the basic framework for the development in Chapter 2. As a counterpart of the ORC, an approach of algebraic observability was given by Diop and Fliess [35] based on exploring the existence of algebraic relations between the state variables of a system and the time derivatives of its measured inputs and outputs. There also exists a differential algebra approach based on calculating the characteristic set and Grobner basis associated to rational dynamic equations. The approach allows for detecting whether there is a globally unique set of parameters fitting the input-output information of a rational nonlinear system, leading to the development of an identifiability (global) testing software DAISY [10]. Other observability approaches may be found in [81] relying on the power series method, in [103] relying on the local state variable isomorphism method, in [65] relying on the characteristic set or standard bases computations, in [15] relying on the elimination method in differential algebra, etc. The above described observability approaches enable the developments of corresponding automatic algorithms where any arbitrary system belonging to a specified class (e.g. the classes of linear, analytic nonlinear, rational nonlinear systems etc.) can be tested without requiring

in-depth understanding of the mathematical tools from the user.

Many works are also devoted to studying the observability of particular types of systems for engineering applications. Udwadia and Sharma [100], Udwadia et al. [101] and Franco et al. [40] placed concentrations on the global identifiability of structural systems; the works discussed the existence of unique solutions for the unknown physical properties of shear-type structures based on a minimum amount of excitation-sensor instruments. Based on incomplete instrumentation, Mukhopadhyay et al. [75] studied the identifiability requirements of stiffness and mass parameters of shear-type structures under base excitation. Non-smooth systems are often encountered in the physical phenomena associated with damage, sliding, impact etc. Chatzis et al. [26] extended the use of the existing observability methods to investigate non-smooth systems which have varying observability properties throughout their dynamics. Liu et al. [62] adopted a graphical approach derived from the dynamical laws that govern a system to determine the sensors that are necessary to reconstruct the full internal state of a biological system.

It is also worth reviewing several of the practical observability analysis methods which use real measured data, although practical observability is not the focus of this thesis. It should be noted that observability testing provides a theoretical ground for practical observability analysis. A theoretically unobservable system would not become practically observable by changing the identification method used, using sensors of better quality, or repeating the dynamic tests. Katafygiotis and Beck [57] discussed the practical identifiability of structural systems through searching for output-equivalent optimal models. Many practical observability analysis methods are proposed based on the use of Fisher Information Matrix [42] for examining the

correlations between model parameters [39]. Raue et al. [83] introduced a novel approach through exploiting the profile likelihood, which is able to detect practically unidentifiable parameters of nonlinear systems and simultaneously calculate confidence intervals.

As has been mentioned in the previous section, developing and improving identification methods for systems driven by unmeasured inputs has gained increasing attention in the past years. However, the aforementioned observability approaches are all based on the assumption of all inputs applied to a system being measured, and therefore they are not applicable to serve for the joint state-parameter-input estimation problems. It was not until very recently that the extensions of the existing observability approaches for systems where some or potentially all of their inputs are unmeasured drew attention. Martinelli [70] introduced the extended ORC or EORC to gain an insight into the observability of input-affine nonlinear systems with unknown (unmeasured) inputs. Maes et al. [67] provided a further extension, namely the EORC-DF, where the existence of direct feedthrough in output measurements was taken into consideration. The situation of output measurements affected by direct feedthrough is often encountered in practice, as for example the case of acceleration measurement in structural dynamics. The proved effectiveness and usefulness of both the EORC(DF) approaches have been demonstrated using a number of classical mechanical and civil engineering models of modest size in the works. A more detailed review of the EORC-DF will be provided in Section 3.1 of Chapter 3 as this is the basic framework of the development in Chapter 3.

From an application point of view, however, the standard implementations of the ORC and EORC(DF) as well as the other aforementioned automatic observabil-

ity approaches are computationally expensive mainly due to their symbolic nature. When these approaches are implemented for real-world engineering systems, often with a large number of states and complicated equations, the complexity of symbolic computations rapidly becomes overwhelming, resulting in significant growth of the physical memory requirements that are hard to be met in a standard computer and increase of the computational cost [31]. In structural engineering applications, for example, a finite element model of a structure may have up to thousands of degrees of freedom and contain equally many dynamic states and parameters to be identified. Those observability approaches are far from being able to handle systems of such size. Computationally efficient observability algorithms have been explored in several works for the case of all inputs being measured [7, 59, 83, 86, 109]. A semi-numerical algorithm was developed by Sedoglavic [86] for robust testing of rational and polynomial nonlinear systems by making use of a numerical realised variational system and the method of Newton's iteration. Villaverde et al. [109] achieved observability testing of a large nonlinear biological system by decomposing it into tractable subsystems which is performed with a combinatorial optimization metaheuristic. However these algorithms are still unable to meet the computational requirements for large civil infrastructures in SHM or systems with equivalent size. Moreover, to the best of author's knowledge, so far no efficient algorithms have been proposed for the cases of unmeasured inputs. Motivated by these drawbacks of the existing observability algorithms, the major objective of this thesis is to develop efficient and robust algorithms to test the observability of large and complex nonlinear systems with or without unmeasured inputs.

1.2.3 Lie symmetries of nonlinear systems

The concept of Lie symmetry was introduced by the Norwegian mathematician Sophus Lie [22, 76] to define a way of variable transformations that leaves differential equations invariant. Lie symmetry analysis of differential equations provides a powerful and fundamental framework to the exploitation of systematic procedures leading to the integration by quadrature of ordinary differential equations, to the determination of invariant solutions of initial and boundary value problems, to the derivation of conservation laws, to the construction of links between different differential equations that turn out to be equivalent, etc. [9, 72, 78, 95]

Sedoglavic [86] introduced the use of Lie symmetries to nonlinear dynamical systems which are mathematically described by state-space and measurement equations. In this context, Lie symmetries of a nonlinear system can be defined as groups of state transformations which fulfil the state-space and measurement equations of the system leaving its inputs and outputs invariant. The existence of such Lie transformations implies the observability properties of the system: there are infinitely many sets of values of states, i.e. parameters and dynamic states, that are admitted by the Lie transformations fulfilling the mathematical model, i.e. state-space and measurement equations, of the system, and an identification algorithm would not be able to find the true values of the states among them based on the given input-output measurements. The system is therefore unobservable according the definition of observability; on the contrary, an observable system would not contain any Lie symmetry within its mathematical model.

In [86], computation of a group of Lie symmetries for a given nonlinear system is achieved through finding the kernel of the observability matrix which is obtained from

an ORC algorithm. The kernel is then used to analytically solve a set of ordinary differential equations with known initial conditions. The set of ordinary differential equations is suggested by the Lie's First Fundamental Theorem to describe the orbits of Lie symmetries with respect to a free parameter. This method however is computationally cumbersome mainly due to the symbolic computation of the observability matrix; analytically solving the differential equations with the aim of obtaining general forms of Lie symmetries is often computationally intractable or mathematically impossible.

Urguplu [102] and Anguelova [3] presented a more computationally efficient framework to calculate Lie symmetries. In the framework, a system of differential equations is derived by applying the Lie's Invariant Function Theorem to state-space and measurement equations, and using a priori assumptions on the form of Lie symmetries, the system of differential equations can be converted to a linear system and solved systematically. Urguplu [102] and Merkt et al. [73] successfully implemented the framework or a similar version to investigate the translation, scaling, affine, quadratic, Mobius, etc. types of symmetries of biological and engineering nonlinear systems. The limitation of this method however is associated with its inability to deal with general forms of Lie symmetries. In addition, both of the aforementioned computational methods are not capable of handling the situations where parts or potentially all of the inputs are unmeasured for a nonlinear system. To relax this limitation, the secondary objective of this thesis is to extend the existing Lie symmetry computational methods for analytic nonlinear systems with unmeasured inputs.

The exploration of Lie symmetries provides an alternative path to assess the observability of nonlinear dynamical systems [3, 86]. Applications of translation and

scaling symmetries to model reduction and non-dimensionalisation problems are also discussed in [46].

1.3 Objectives and outline

Motivated by the need to address the significant computational complexity problem associated with the implementations of the existing observability approaches, as described in the literature review, this thesis develops two efficient and robust algorithms to test the observability of linear and rational nonlinear systems with or without unmeasured inputs. The developments of the suggested algorithms aim at extending and maximising the applicabilities of the ORC and EORC-DF approaches in particular, to real-world engineering systems that are often large and complex. The secondary contribution of the thesis is devoted to investigating Lie symmetries of analytic nonlinear systems with unmeasured inputs. Robust algorithms are proposed to compute the analytic and power series forms of Lie symmetries for such type of dynamical systems. This research also provides a pioneer study on the applications of the computed Lie symmetries to the problems of observability assessment, model reduction, system identification improvement, etc. In the final research presented in this thesis, the previously proposed observability and Lie symmetry algorithms are applied to the damage-healing hysteretic model, which is a complex, highly nonlinear and non-smooth mechanical model recently introduced to account for the behaviour of self-healing materials. After the processes of 'observability analysis' and 'model reduction', the Unscented Kalman filter (UKF) and the discontinuous Unscented Kalman filter (DUKF) are successfully implemented to identify the model parameters using contaminated earthquake record and simulated output data. Specifically,

this thesis is organised as follows:

- In Chapter 2, *a robust algorithm to test the observability of large linear systems with unknown parameters* is developed. The mathematical derivation of the algorithm is presented in detail. Applications and superior performance of the algorithm are demonstrated using several examples of large linear dynamical models of engineering systems containing up to thousands of parameters and dynamic states. The main content of this chapter refers to the paper entitled '*A robust algorithm to test the observability of large linear systems with unknown parameters*', which has been submitted to the Journal of Mechanical Systems and Signal Processing, and the paper entitled '*Robust computation of the observability of large linear systems with unknown parameters*' [89], which has been published in the Proceedings of the Sixth International Symposium on Life-Cycle Civil Engineering. The author of this thesis is the first author of both the papers.

- In Chapter 3, *an efficient algorithm to test the observability of rational nonlinear systems with unmeasured inputs* is proposed. The mathematical derivation of the algorithm is presented in detail. Applications and superior performance of the algorithm are demonstrated using several examples of large and complex dynamical models of engineering systems. The main content of this chapter refers to the author's paper entitled '*An efficient algorithm to test the observability of rational nonlinear systems with unmeasured inputs*', which is to be submitted to the Journal of Mechanical Systems and Signal Processing.

- In Chapter 4, *Lie symmetries of nonlinear systems with unmeasured inputs* are

studied. Frameworks for computing the analytic and power series forms of Lie symmetries are established. This chapter also explores the potential applications of symmetries to various problems encountered in system identification campaigns. The main content of this chapter refers to the paper entitled '*Lie symmetries, observability and model transformation of nonlinear systems with unknown inputs*' [87], which has been published in the Proceedings of the XI International Conference on Structural Dynamics, and the journal paper entitled '*Lie symmetries of nonlinear systems with unmeasured inputs*' which is under preparation. The author of this thesis is the first author of both the papers.

- In Chapter 5, observability analysis and model reduction using symmetries are performed on the Damage-Healing Hysteretic Model. Accurate identification of the model is achieved through the use of UKF and DUKF techniques. The main content of this chapter refers to the author's paper entitled '*Observability and identification of damage-healing hysteretic model*' [88], which has been published in the Proceedings of the Seventh World Conference on Structural Control and Health Monitoring.

- In Chapter 6, the thesis is concluded with suggestions and foresight for future work.

Chapter 2

A Robust Algorithm to Test the Observability of Large Linear Systems with Unknown Parameters

The objective of this chapter is to propose an efficient algorithmic implementation of the Observability Rank Condition (ORC) to examine the observability of linear systems whose dynamic states and unknown parameters are to be identified. This category of systems covers a wide range of civil and mechanical systems. The chapter derives an explicit expression of the observability matrix of linear systems with unknown parameters. It shows the viability of computing the observability matrix recursively by introducing a product rule involving three-dimensional matrices. The computationally expensive symbolic differentiations are replaced by matrix additions and multiplications. This further allows for numerical realization of the algorithm, and in addition the application of modular operations makes it possible to use floating-point integers which substantially enhances the computation efficiency. Applications and superior performance of the algorithm are demonstrated through several illustrative engineering examples.

2.1 The Observability Rank Condition (ORC)

The first section of this chapter provides a brief review of the mathematical derivation and implementation of the ORC, while the details of the theory can be found in [44, 50, 2].

2.1.1 Theory

Consider a nonlinear system with affine inputs written as the following state-space representation:

$$\begin{aligned} \dot{\boldsymbol{x}}_{\boldsymbol{t}} &= \boldsymbol{F}(\boldsymbol{x}_{\boldsymbol{t}}, \boldsymbol{\theta}) + \sum_{i=1}^{m} \boldsymbol{G}_{\boldsymbol{i}}(\boldsymbol{x}_{\boldsymbol{t}}, \boldsymbol{\theta}) u_i \\ \boldsymbol{y} &= \boldsymbol{h}(\boldsymbol{x}_{\boldsymbol{t}}, \boldsymbol{\theta}) \end{aligned} \tag{2.1}$$

where $\boldsymbol{x}_{\boldsymbol{t}} \in \mathbb{R}^{n_t}$ denotes the time-variant dynamic states, $\boldsymbol{\theta} \in \mathbb{R}^{n_\theta}$ the time-invariant parameters, $u_1, u_2, ..., u_m$ the measured independent inputs and $\boldsymbol{y} \in \mathbb{R}^p$ the output measurement vector. $\boldsymbol{F}, \boldsymbol{G_1}, \boldsymbol{G_2}, ..., \boldsymbol{G_m}$ and $\boldsymbol{h}$ are vectors of nonlinear smooth functions. If the time-invariant parameters of the system are unknown, they can be treated as additional states with zero dynamics, i.e. $\dot{\boldsymbol{\theta}} = \mathbf{0}_{n_\theta \times 1}$. This leads to a state augmentation by including both the dynamic states and the parameters in a common state vector $\boldsymbol{x} \in \mathbb{R}^n$:

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}_{\boldsymbol{t}} \\ \boldsymbol{\theta} \end{bmatrix}, \qquad n = n_t + n_\theta \tag{2.2}$$

and the system in equation (2.1) may be rewritten with respect to the state vector as:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}) + \sum_{i=1}^{m} \boldsymbol{g}_{\boldsymbol{i}}(\boldsymbol{x}) u_i \\ \boldsymbol{y} &= \boldsymbol{h}(\boldsymbol{x}) \end{aligned} \tag{2.3}$$

where

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{bmatrix} \boldsymbol{F}(\boldsymbol{x}_{\boldsymbol{t}}, \boldsymbol{\theta}) \\ \mathbf{0}_{n_\theta \times 1} \end{bmatrix}, \qquad \boldsymbol{g}_{\boldsymbol{i}}(\boldsymbol{x}) = \begin{bmatrix} \boldsymbol{G}_{\boldsymbol{i}}(\boldsymbol{x}_{\boldsymbol{t}}, \boldsymbol{\theta}) \\ \mathbf{0}_{n_\theta \times 1} \end{bmatrix} \tag{2.4}$$

The observability of the above system defined in equations (2.3) and (2.4) can be examined by the ORC. To derive the observability testing approach, it is first necessary to introduce the Lie derivative operation. The Lie derivative of a vector of scalar functions, $\boldsymbol{\lambda} = \begin{bmatrix}\lambda_1(x) & ... & \lambda_p(x)\end{bmatrix}^T$, along a vector field, $\boldsymbol{v} = \begin{bmatrix}v_1(x) & ... & v_n(x)\end{bmatrix}^T$, is defined as:

$$\boldsymbol{L_v\lambda} = \frac{\partial \boldsymbol{\lambda}}{\partial \boldsymbol{x}}\boldsymbol{v} \tag{2.5}$$

where $\frac{\partial \boldsymbol{\lambda}}{\partial \boldsymbol{x}}$ is a Jacobian matrix:

$$\frac{\partial \boldsymbol{\lambda}}{\partial \boldsymbol{x}} = \begin{bmatrix} \frac{\partial \lambda_1}{\partial x_1} & \frac{\partial \lambda_1}{\partial x_2} & \cdots & \frac{\partial \lambda_1}{\partial x_n} \\ \frac{\partial \lambda_2}{\partial x_1} & \frac{\partial \lambda_2}{\partial x_2} & \cdots & \frac{\partial \lambda_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \lambda_p}{\partial x_1} & \frac{\partial \lambda_p}{\partial x_2} & \cdots & \frac{\partial \lambda_p}{\partial x_n} \end{bmatrix} \tag{2.6}$$

Consider a piecewise-constant input function for $i = 1, ..., m$:

$$\begin{aligned} u_i(t) &= u_i^1, \ \ for \ \ t \in [0, t_1) \\ u_i(t) &= u_i^k, \ \ for \ \ t \in [t_1 + ... + t_{k-1}, t_1 + ... + t_k), \ \ k \geq 2 \end{aligned} \tag{2.7}$$

Define the vector field:

$$\boldsymbol{V_k} = \boldsymbol{f} + \sum_{i=1}^{m} \boldsymbol{g_i} u_i^k \tag{2.8}$$

and let $\boldsymbol{\Phi_t^k}$ denote the corresponding flow. The flow $\boldsymbol{\Phi_t}(\boldsymbol{x})$ of a vector field $\boldsymbol{V}$ is by definition the solution of:

$$\begin{aligned} \frac{\partial \boldsymbol{\Phi_t}(\boldsymbol{x})}{\partial t} &= \boldsymbol{V}(\boldsymbol{\Phi_t}(\boldsymbol{x})) \\ \boldsymbol{\Phi_0}(\boldsymbol{x}) &= \boldsymbol{x} \end{aligned} \tag{2.9}$$

Then, the state vector reached at time $t_1 + ... + t_k$ starting from an initial condition $\boldsymbol{x^0}$ under the piecewise-constant inputs may be expressed as:

$$\boldsymbol{x}(t_1 + ... + t_k) = \boldsymbol{\Phi_{t_k}^k} \circ ... \circ \boldsymbol{\Phi_{t_1}^1}(\boldsymbol{x^0}) \tag{2.10}$$

and the corresponding output vector $\boldsymbol{y}$ can be expressed as:

$$\boldsymbol{y}(t_1 + ... + t_k) = \boldsymbol{h}(\boldsymbol{x}(t_1 + ... + t_k)) \tag{2.11}$$

It should be noted herein that this output may be regarded as the value of a mapping:

$$\begin{aligned} F^{\boldsymbol{x^0}} &: (-\epsilon, \epsilon)^k \to \mathbb{R} \\ (t_1, ..., t_k) &\to \boldsymbol{h} \circ \boldsymbol{\Phi}^{\boldsymbol{k}}_{\boldsymbol{t_k}} \circ ... \circ \boldsymbol{\Phi}^{\boldsymbol{1}}_{\boldsymbol{t_1}}(\boldsymbol{x^0}) \end{aligned} \tag{2.12}$$

In the following, let indistinguishable states be defined as a pair of initial conditions, $\boldsymbol{x^a}$ and $\boldsymbol{x^b}$, producing two identical outputs for any possible piecewise-constant inputs. The indistinguishable states must satisfy:

$$F^{\boldsymbol{x^a}}(t_1, ..., t_k) = F^{\boldsymbol{x^b}}(t_1, ..., t_k) \tag{2.13}$$

for all possible $(t_1, ..., t_k)$ with $0 \leq t_i < \epsilon$. From this it can be deduced that:

$$\left.\frac{\partial^k F^{\boldsymbol{x^a}}}{\partial t_1 ... \partial t_k}\right|_{t_1=...=t_k=0} = \left.\frac{\partial^k F^{\boldsymbol{x^b}}}{\partial t_1 ... \partial t_k}\right|_{t_1=...=t_k=0} \tag{2.14}$$

A straightforward calculation shows that:

$$\left.\frac{\partial^k F^{\boldsymbol{x^0}}}{\partial t_1 ... \partial t_k}\right|_{t_1=...=t_k=0} = \boldsymbol{L_{V_1}} ... \boldsymbol{L_{V_k}} \boldsymbol{h}(\boldsymbol{x^0}) \tag{2.15}$$

and therefore,

$$\boldsymbol{L_{V_1}} ... \boldsymbol{L_{V_k}} \boldsymbol{h}(\boldsymbol{x^a}) = \boldsymbol{L_{V_1}} ... \boldsymbol{L_{V_k}} \boldsymbol{h}(\boldsymbol{x^b}) \tag{2.16}$$

Now recall that $\boldsymbol{V_j}$ $(j = 1, ..., k)$ depends on $(u^j_1, ..., u^j_m)$ and that the above equality must hold for all possible choices of $(u^j_1, ..., u^j_m) \in \mathbb{R}^m$. By appropriately selecting these $(u^j_1, ..., u^j_m)$ one easily arrives at an equality of the form:

$$\boldsymbol{L_{v_1}} ... \boldsymbol{L_{v_k}} \boldsymbol{h}(\boldsymbol{x^a}) = \boldsymbol{L_{v_1}} ... \boldsymbol{L_{v_k}} \boldsymbol{h}(\boldsymbol{x^b}) \tag{2.17}$$

where $\boldsymbol{v_1}, ..., \boldsymbol{v_k}$ are the vector fields belonging to the set $\{\boldsymbol{f}, \boldsymbol{g_1}, ..., \boldsymbol{g_m}\}$. Let $\boldsymbol{\gamma_2} = \boldsymbol{L_{V_2}} ... \boldsymbol{L_{V_k}} \boldsymbol{h}$, and from the equality $\boldsymbol{L_{V_1}} \boldsymbol{\gamma_2}(\boldsymbol{x}^a) = \boldsymbol{L_{V_1}} \boldsymbol{\gamma_2}(\boldsymbol{x^b})$ it can be obtained that:

$$\boldsymbol{L_f} \boldsymbol{\gamma_2}(\boldsymbol{x^a}) + \sum_{i=1}^{m} \boldsymbol{L_{g_i}} \boldsymbol{\gamma_2}(\boldsymbol{x^a}) u^1_i = \boldsymbol{L_f} \boldsymbol{\gamma_2}(\boldsymbol{x^b}) + \sum_{i=1}^{m} \boldsymbol{L_{g_i}} \boldsymbol{\gamma_2}(\boldsymbol{x^b}) u^1_i \tag{2.18}$$

This, due to the arbitrariness of the $(u_1^1, ..., u_m^1)$, implies that:

$$\boldsymbol{L_v \gamma_2}(\boldsymbol{x^a}) = \boldsymbol{L_v \gamma_2}(\boldsymbol{x^b}) \tag{2.19}$$

where $\boldsymbol{v}$ is any vector in the set $\{\boldsymbol{f}, \boldsymbol{g_1}, ..., \boldsymbol{g_m}\}$. This procedure can be iterated, by setting $\boldsymbol{\gamma_3} = \boldsymbol{L_{V_3}} ... \boldsymbol{L_{V_k}} \boldsymbol{h}$. From the above equality one gets:

$$\boldsymbol{L_v L_f \gamma_3}(\boldsymbol{x^a}) + \sum_{i=1}^{m} \boldsymbol{L_v L_{g_i} \gamma_3}(\boldsymbol{x^a}) u_i^2 = \boldsymbol{L_v L_f \gamma_3}(\boldsymbol{x^b}) + \sum_{i=1}^{m} \boldsymbol{L_v L_{g_i} \gamma_3}(\boldsymbol{x^b}) u_i^2 \tag{2.20}$$

and therefore,

$$\boldsymbol{L_{v_1} L_{v_2} \gamma_3}(\boldsymbol{x^a}) = \boldsymbol{L_{v_1} L_{v_2} \gamma_3}(\boldsymbol{x^b}) \tag{2.21}$$

for all $\boldsymbol{v_1}, \boldsymbol{v_2}$ belonging to $\{\boldsymbol{f}, \boldsymbol{g_1}, ..., \boldsymbol{g_m}\}$. Finally one is able to arrive at equation (2.17).

Equation (2.17) implies that all the possible Lie derivatives $\boldsymbol{L_{v_1}} ... \boldsymbol{L_{v_k}} \boldsymbol{h}$ for any $k = 1, 2, ...$ are evaluated equal at indistinguishable states. This derived Lie derivative property allows to establish the following rank testing approach to identify whether such pairs of indistinguishable states (locally) exist. Successively arrange those Lie derivatives, i.e. $\boldsymbol{h}$, $\boldsymbol{L_f h}$, $\boldsymbol{L_{g_i} h}$, $\boldsymbol{L_f L_f h}$, $\boldsymbol{L_{g_i} L_f h}$, etc. in a column vector $\boldsymbol{\lambda}$:

$$\boldsymbol{\lambda} = \begin{bmatrix} \boldsymbol{h} \\ \boldsymbol{L_f h} \\ \boldsymbol{L_{g_i} h} \\ \boldsymbol{L_f L_f h} \\ \boldsymbol{L_{g_i} L_f h} \\ \vdots \end{bmatrix} \tag{2.22}$$

and define the observability matrix $\boldsymbol{\Omega}$ as the gradient of $\boldsymbol{\lambda}$ with respect to $\boldsymbol{x}$:

$$\boldsymbol{\Omega} = \frac{\partial \boldsymbol{\lambda}}{\partial \boldsymbol{x}} \tag{2.23}$$

Note that it has been shown in [2, 70] that the Lie derivatives of $k = 1, ..., n-1$, where n is the dimension of the state vector, are sufficient to explore the observability.

If $\boldsymbol{\Omega}$ is a full-rank matrix at $\boldsymbol{x} = \boldsymbol{x^0}$, i.e. $rank(\boldsymbol{\Omega}(\boldsymbol{x^0})) = n$, the Inverse Function Theorem [94] ensures that the only indistinguishable state of $\boldsymbol{x^0}$, i.e. the one resulting in equal Lie derivatives as $\boldsymbol{x^0}$, within its closed neighbourhood is $\boldsymbol{x^0}$ itself. This consequently means that it is possible to distinguish and observe $\boldsymbol{x^0}$ and hence the trajectory $\boldsymbol{x}(t, \boldsymbol{x^0})$ from its neighbours using the given input-output measurements. Following the above fact, the observability testing criterion can thus be concluded: the system in equations (2.3) and (2.4) is said to satisfy the Observability Rank Condition (ORC) at $\boldsymbol{x^0}$ such that its state vector $\boldsymbol{x}(t, \boldsymbol{x^0})$ is (locally weakly) observable, if and only if $rank(\boldsymbol{\Omega}(\boldsymbol{x^0})) = n$; the system satisfies the ORC if $rank(\boldsymbol{\Omega}(\boldsymbol{x})) = n$ for all possible $\boldsymbol{x} \in \mathbb{R}^n$ (normally except for some singular points where the ranks of $\boldsymbol{\Omega}$ would be reduced).

2.1.2 Implementation

To implement the ORC for a given nonlinear system with state-space and output equations, a simple algorithm described in [26, 50] can be used to compute the Lie derivatives in a recursive way and successively arrange the gradients of these Lie derivatives to output the observability matrix. The algorithm is presented in the following:

Algorithm 2.1

Input to the algorithm: the system functions $\boldsymbol{f}, \boldsymbol{g_1}, \boldsymbol{g_2}, ..., \boldsymbol{g_m}$, the output functions $\boldsymbol{h}$ and the system states $\boldsymbol{x}$

Output from the algorithm: the observability matrix $\boldsymbol{\Omega}$

Initialization: Let $k = 0$, $\boldsymbol{\lambda_0} = \boldsymbol{h}$, $\boldsymbol{\Omega_0} = \frac{\partial \boldsymbol{\lambda_0}}{\partial \boldsymbol{x}}$

1. Compute the Lie derivatives $\boldsymbol{\lambda_{k+1}} = \begin{bmatrix} (\boldsymbol{L_f \lambda_k})^T & (\boldsymbol{L_{g_1} \lambda_k})^T & (\boldsymbol{L_{g_2} \lambda_k})^T & \dots & (\boldsymbol{L_{g_m} \lambda_k})^T \end{bmatrix}^T$

2. Compute the rows of the gradient of the Lie derivatives, and arrange them into the observability matrix $\boldsymbol{\Omega_{k+1}} = \boldsymbol{\Omega_k} \cup \frac{\partial \boldsymbol{\lambda_{k+1}}}{\partial \boldsymbol{x}}$

3. Compute the rank, if $rank(\boldsymbol{\Omega_{k+1}}) = rank(\boldsymbol{\Omega_k})$, or $rank(\boldsymbol{\Omega_{k+1}}) = n$, or $k = n-2$, end and output $\boldsymbol{\Omega} = \boldsymbol{\Omega_{k+1}}$

4. Let $k = k+1$ and go to step 1

It should be noted that the computations involved in Algorithm 2.1 are symbolic as the gradient operations are analytic and thus must be implemented by differentiating symbolically the associated expressions. Once the observability matrix is obtained, the observability of the system in equations (2.3) and (2.4) is analyzed through rank testing. The system satisfies the Observability Rank Condition so that it is observable if $\boldsymbol{\Omega}$ has a rank equal to the dimension of $\boldsymbol{x}$, n. The observability of the $l^{th}(l = 1, ..., n)$ state of $\boldsymbol{x}$ can be detected by removing the l^{th} column of $\boldsymbol{\Omega}$: if the rank of the resulting matrix is smaller than the rank of $\boldsymbol{\Omega}$, then the l^{th} state is observable; otherwise it is unobservable. If that state is further a parameter it will be correspondingly be deemed as identifiable or unidentifiable. If and only if all of the dynamic states are observable and all of the parameters are identifiable, then the system is said to be observable. Furthermore, the transcendence degree of the system, that is calculated as the rank deficiency $n - rank(\boldsymbol{\Omega})$, can be used to account for the number of unobservable states for which a modelling assumption needs to be made in order to reduce an unobservable system to become observable [26, 2].

2.2 The observability matrix of linear systems with unknown parameters

It is worth mentioning that the observability matrix of linear systems with *known* parameters has been well-explored [55]. This work focuses on linear underlying systems with *unknown* parameters $\boldsymbol{\theta}$ which are described by:

$$\begin{aligned} \dot{\boldsymbol{x}}_{\boldsymbol{t}} &= \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{x}_{\boldsymbol{t}} + \sum_{i=1}^{m} \boldsymbol{B}_{\boldsymbol{i}}(\boldsymbol{\theta})u_i, \qquad \dot{\boldsymbol{\theta}} = \mathbf{0}_{n_\theta \times 1} \\ \boldsymbol{y} &= \boldsymbol{C}(\boldsymbol{\theta})\boldsymbol{x}_{\boldsymbol{t}} \end{aligned} \tag{2.24}$$

where $\boldsymbol{A}(\boldsymbol{\theta})$, $\boldsymbol{B}_{\boldsymbol{i}}(\boldsymbol{\theta})$ and $\boldsymbol{C}(\boldsymbol{\theta})$ are the matrices with elements functions of the unknown parameters. Strictly speaking, the system in equation (2.24) is linear with respect to the dynamic states and the inputs, but overall nonlinear due to the terms of the products between $\boldsymbol{A}(\boldsymbol{\theta})$, $\boldsymbol{B}_{\boldsymbol{i}}(\boldsymbol{\theta})$, $\boldsymbol{C}(\boldsymbol{\theta})$ and $\boldsymbol{x}_{\boldsymbol{t}}$, u_i. Following the state augmentation $\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}_{\boldsymbol{t}}{}^T & \boldsymbol{\theta}^T \end{bmatrix}^T$ for the purpose of parameter estimation, the system is rewritten as the form of (2.3), and the corresponding $\boldsymbol{f}$ and $\boldsymbol{g}_{\boldsymbol{i}}$ can be obtained as:

$$\boldsymbol{f} = \begin{bmatrix} \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{x}_{\boldsymbol{t}} \\ \mathbf{0}_{n_\theta \times 1} \end{bmatrix}, \qquad \boldsymbol{g}_{\boldsymbol{i}} = \begin{bmatrix} \boldsymbol{B}_{\boldsymbol{i}}(\boldsymbol{\theta}) \\ \mathbf{0}_{n_\theta \times 1} \end{bmatrix} \tag{2.25}$$

Algorithm 2.1 is applicable to test the observability of the system in equations (2.24) and (2.25). However, as noted in the literature review, the symbolic computations involved in the algorithm are cumbersome especially when the augmented state vector $\boldsymbol{x}$ is of large size and the system functions $\boldsymbol{f}$ and $\boldsymbol{g}_{\boldsymbol{i}}$ are complicated, which appears as a substantial limitation to real-life engineering applications. In order to develop a more computationally efficient algorithm, this section aims to derive an explicit expression of the observability matrix of the system. The key idea is to separate all the Lie derivatives of the system into three subsets:

$$\{\boldsymbol{L}_{\boldsymbol{f}}^{\boldsymbol{k}}\boldsymbol{h} : k = 0, 1, ..., n-1\} \tag{2.26}$$

$$\{\boldsymbol{L_{g_i}L_f^k h} : k = 0, 1, ..., n-2\} \tag{2.27}$$

$$\{\boldsymbol{L_{v_r}...L_{v_1}L_{g_i}L_f^k h} : k = 0, 1, ..., n-3, r = 1, 2, ..., n-k-2\} \tag{2.28}$$

where $\boldsymbol{v_r}$ is any of the vector fields belonging to the set $\{\boldsymbol{f}, \boldsymbol{g_1}, \boldsymbol{g_2}, ..., \boldsymbol{g_m}\}$. $\boldsymbol{L_f^k h}$ is the k^{th} order Lie derivative, i.e. $\underbrace{\boldsymbol{L_f...L_f}}_{k}\boldsymbol{h}$, and this can be calculated recursively by $\boldsymbol{L_f^k h = L_f L_f^{k-1} h}$ with $\boldsymbol{L_f^0 h = h}$. The expression of each subset of the Lie derivatives is derived as follows.

In (2.26), the k^{th} order Lie derivative of $\boldsymbol{h}$ along $\boldsymbol{f}$ is expressed as:

$$\boldsymbol{L_f^k h = CA^k x_t} \tag{2.29}$$

A proof by induction for equation (2.29) follows: for $k = 0$, $\boldsymbol{L_f^0 h = h = Cx_t}$, which indeed satisfies (2.29). If one assumes that equation (2.29) holds for $k-1$, i.e. $\boldsymbol{L_f^{k-1} h = CA^{k-1} x_t}$, the equality will be shown to hold for k. Using equation (2.5) to compute $\boldsymbol{L_f^k h = L_f L_f^{k-1} h}$, it is necessary to compute $\frac{\partial \boldsymbol{L_f^{k-1} h}}{\partial \boldsymbol{x}} = \frac{\partial \boldsymbol{CA^{k-1} x_t}}{\partial \boldsymbol{x}}$. The gradient with respect to the state vector, $\boldsymbol{x}$, can be separated into a sub-gradient with respect to the dynamic states, $\boldsymbol{x_t}$, and a sub-gradient with respect to the parameters, $\boldsymbol{\theta}$, giving $\frac{\partial \boldsymbol{CA^{k-1} x_t}}{\partial \boldsymbol{x}} = \begin{bmatrix} \frac{\partial \boldsymbol{CA^{k-1} x_t}}{\partial \boldsymbol{x_t}} & \frac{\partial \boldsymbol{CA^{k-1} x_t}}{\partial \boldsymbol{\theta}} \end{bmatrix}$. Since $\boldsymbol{CA^{k-1}}$ is a matrix depending on $\boldsymbol{\theta}$ but not on $\boldsymbol{x_t}$, $\frac{\partial \boldsymbol{CA^{k-1} x_t}}{\partial \boldsymbol{x_t}} = \boldsymbol{CA^{k-1}}$. Thus taking the expression of $\boldsymbol{f}$ in equation (2.25) and substituting it into (2.5), $\boldsymbol{L_f^k h}$ is computed as:

$$\boldsymbol{L_f^k h = L_f L_f^{k-1} h} = \begin{bmatrix} \boldsymbol{CA^{k-1}} & \frac{\partial \boldsymbol{CA^{k-1} x_t}}{\partial \boldsymbol{\theta}} \end{bmatrix} \begin{bmatrix} \boldsymbol{Ax_t} \\ \boldsymbol{0}_{n_\theta \times 1} \end{bmatrix} = \boldsymbol{CA^k x_t} \tag{2.30}$$

which coincides with equation (2.29) for k, and therefore concludes the proof by induction of (2.29).

Having obtained the expression of $\boldsymbol{L_f^k h}$, it proceeds to derive the Lie derivative of $\boldsymbol{L_f^k h}$ along $\boldsymbol{g_i}$ using equations (2.25) and (2.5) in (2.27):

$$\boldsymbol{L_{g_i} L_f^k h} = \begin{bmatrix} \boldsymbol{CA}^k & \frac{\partial \boldsymbol{CA}^k \boldsymbol{x_t}}{\partial \boldsymbol{\theta}} \end{bmatrix} \begin{bmatrix} \boldsymbol{B_i} \\ \boldsymbol{0}_{n_\theta \times 1} \end{bmatrix} = \boldsymbol{CA}^k \boldsymbol{B_i} \tag{2.31}$$

To evaluate the remaining Lie derivatives $\boldsymbol{L_{v_r} ... L_{v_1} L_{g_i} L_f^k h}$ appearing in (2.28), $\boldsymbol{L_{v_1} L_{g_i} L_f^k h}$ corresponding to either $\boldsymbol{L_f L_{g_i} L_f^k h}$ or $\boldsymbol{L_{g_j} L_{g_i} L_f^k h}$ $(j = 1, ..., m)$ is first computed using equations (2.25) and (2.5). $\frac{\partial \boldsymbol{L_{g_i} L_f^k h}}{\partial \boldsymbol{x}} = \frac{\partial \boldsymbol{CA}^k \boldsymbol{B_i}}{\partial \boldsymbol{x}}$ based on equation (2.31) is written as $\frac{\partial \boldsymbol{L_{g_i} L_f^k h}}{\partial \boldsymbol{x}} = \begin{bmatrix} \frac{\partial \boldsymbol{CA}^k \boldsymbol{B_i}}{\partial \boldsymbol{x_t}} & \frac{\partial \boldsymbol{CA}^k \boldsymbol{B_i}}{\partial \boldsymbol{\theta}} \end{bmatrix}$. Since $\boldsymbol{CA}^k \boldsymbol{B_i}$ is a vector of parameters but not of dynamic states, $\frac{\partial \boldsymbol{CA}^k \boldsymbol{B_i}}{\partial \boldsymbol{x_t}}$ is a $p \times n_t$ zero matrix. Therefore,

$$\begin{aligned} \boldsymbol{L_f L_{g_i} L_f^k h} &= \begin{bmatrix} \boldsymbol{0}_{p \times n_t} & \frac{\partial \boldsymbol{CA}^k \boldsymbol{B_i}}{\partial \boldsymbol{\theta}} \end{bmatrix} \begin{bmatrix} \boldsymbol{Ax_t} \\ \boldsymbol{0}_{n_\theta \times 1} \end{bmatrix} = \boldsymbol{0}_{p \times 1} \\ \boldsymbol{L_{g_j} L_{g_i} L_f^k h} &= \begin{bmatrix} \boldsymbol{0}_{p \times n_t} & \frac{\partial \boldsymbol{CA}^k \boldsymbol{B_i}}{\partial \boldsymbol{\theta}} \end{bmatrix} \begin{bmatrix} \boldsymbol{B_j} \\ \boldsymbol{0}_{n_\theta \times 1} \end{bmatrix} = \boldsymbol{0}_{p \times 1} \end{aligned} \tag{2.32}$$

The zero Lie derivatives $\boldsymbol{L_{v_1} L_{g_i} L_f^k h}$ lead to zero higher order Lie derivatives, i.e. $\boldsymbol{L_{v_r} ... L_{v_1} L_{g_i} L_f^k h} = \boldsymbol{0}_{p \times 1}$, which can be ignored when construting the observability matrix since they do not deliver any observability information of the system.

Thanks to equations (2.29) and (2.31), where all the nonzero Lie derivatives have been expressed in terms of the system matrices $\boldsymbol{A}$, $\boldsymbol{B_i}$, $\boldsymbol{C}$ and the dynamic states $\boldsymbol{x_t}$, the expression of the observability matrix can be readily obtained by evaluating the gradients of the Lie derivatives. At the k^{th} iteration of the algorithm, one obtains:

$$\begin{aligned} \frac{\partial \boldsymbol{L_f^k h}}{\partial \boldsymbol{x}} &= \begin{bmatrix} \boldsymbol{CA}^k & \frac{\partial \boldsymbol{CA}^k \boldsymbol{x_t}}{\partial \boldsymbol{\theta}} \end{bmatrix} \\ \frac{\partial \boldsymbol{L_{g_i} L_f^{k-1} h}}{\partial \boldsymbol{x}} &= \begin{bmatrix} \boldsymbol{0}_{p \times n_t} & \frac{\partial \boldsymbol{CA}^{k-1} \boldsymbol{B_i}}{\partial \boldsymbol{\theta}} \end{bmatrix} \end{aligned} \tag{2.33}$$

Then for $k = 0, ..., n-1$ the full observability matrix of the system in equations (2.24)

and (2.25) attains the following form:

$$\boldsymbol{\Omega} = \begin{bmatrix} \boldsymbol{C} & \frac{\partial \boldsymbol{C}\boldsymbol{x_t}}{\partial \boldsymbol{\theta}} \\ \boldsymbol{CA} & \frac{\partial \boldsymbol{CA}\boldsymbol{x_t}}{\partial \boldsymbol{\theta}} \\ \boldsymbol{0}_{p\times n_t} & \frac{\partial \boldsymbol{CB_1}}{\partial \boldsymbol{\theta}} \\ \vdots & \vdots \\ \boldsymbol{0}_{p\times n_t} & \frac{\partial \boldsymbol{CB_m}}{\partial \boldsymbol{\theta}} \\ \vdots & \vdots \\ \boldsymbol{CA}^k & \frac{\partial \boldsymbol{CA}^k\boldsymbol{x_t}}{\partial \boldsymbol{\theta}} \\ \boldsymbol{0}_{p\times n_t} & \frac{\partial \boldsymbol{CA}^{k-1}\boldsymbol{B_1}}{\partial \boldsymbol{\theta}} \\ \vdots & \vdots \\ \boldsymbol{0}_{p\times n_t} & \frac{\partial \boldsymbol{CA}^{k-1}\boldsymbol{B_m}}{\partial \boldsymbol{\theta}} \\ \vdots & \vdots \\ \boldsymbol{CA}^{n-1} & \frac{\partial \boldsymbol{CA}^{n-1}\boldsymbol{x_t}}{\partial \boldsymbol{\theta}} \\ \boldsymbol{0}_{p\times n_t} & \frac{\partial \boldsymbol{CA}^{n-2}\boldsymbol{B_1}}{\partial \boldsymbol{\theta}} \\ \vdots & \vdots \\ \boldsymbol{0}_{p\times n_t} & \frac{\partial \boldsymbol{CA}^{n-2}\boldsymbol{B_m}}{\partial \boldsymbol{\theta}} \end{bmatrix} \tag{2.34}$$

2.3 Recursive computation

This section aims to compute the observability matrix of equation (2.34) in a recursive way. A key objective of the recursive formulation is to avoid the use of symbolic differentiations and instead compute the submatrices involved in $\boldsymbol{\Omega}$ by the operations of matrix addition and multiplication. While it is straightforward to compute the submatrix $\boldsymbol{CA}^k$ recursively as $\boldsymbol{CA}^{k-1}\boldsymbol{A}$, the recursive computations of $\frac{\partial \boldsymbol{CA}^k\boldsymbol{x_t}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{CA}^{k-1}\boldsymbol{B_i}}{\partial \boldsymbol{\theta}}$ are realized by introducing a product rule to expand the gradient of matrix multiplication, which is analogous to the well known product rule for the derivative of product of two scalar functions: $(fg)' = f'g + fg'$.

2.3.1 A product rule for the gradient of matrix multiplication

Let $\boldsymbol{U}$ be a 2-dimensional matrix with its $(i_1, i_2)^{th}$ element denoted by $U_{i_1 i_2}$, and $\boldsymbol{z}$ be a vector with its l^{th} element z_l. The gradient of $\boldsymbol{U}$ with respect to $\boldsymbol{z}$, $\boldsymbol{V} = \frac{\partial \boldsymbol{U}}{\partial \boldsymbol{z}}$, is a

3-dimensional matrix where the $(i_1, i_2, l)^{th}$ element of $\boldsymbol{V}$ is calculated by $V_{i_1 i_2 l} = \frac{\partial U_{i_1 i_2}}{\partial z_l}$.

Let $\boldsymbol{U}$ and $\boldsymbol{M}$ be matrices with dimensions $\alpha \times \beta$ and $\beta \times \gamma$ respectively, and $\boldsymbol{z}$ be a vector with dimension $\nu \times 1$. A product rule for the gradient of the matrix multiplication $\boldsymbol{UM}$ with respect to $\boldsymbol{z}$ is introduced as follows:

$$\frac{\partial \boldsymbol{UM}}{\partial \boldsymbol{z}} = (\frac{\partial \boldsymbol{U}}{\partial \boldsymbol{z}}^{T_t} \boldsymbol{M})^{T_t} + \boldsymbol{U}\frac{\partial \boldsymbol{M}}{\partial \boldsymbol{z}} \tag{2.35}$$

where $\frac{\partial \boldsymbol{UM}}{\partial \boldsymbol{z}}$ is a $\alpha \times \gamma \times \nu$ matrix, $\frac{\partial \boldsymbol{U}}{\partial \boldsymbol{z}}$ is a $\alpha \times \beta \times \nu$ matrix and $\frac{\partial \boldsymbol{M}}{\partial \boldsymbol{z}}$ is a $\beta \times \gamma \times \nu$ matrix. The notation 'T_t' over a 3-dimensional matrix is used to denote the transpose between the second dimension and the third dimension of the matrix. For example, $\boldsymbol{V}$ under the transpose, $\boldsymbol{V}^{T_t}$, is a 3-dimensional matrix with its $(i_1, l, i_2)^{th}$ element equal to the $(i_1, i_2, l)^{th}$ element of $\boldsymbol{V}$, i.e. $(V^{T_t})_{i_1 l i_2} = V_{i_1 i_2 l}$. The multi-dimensional matrix multiplication follows the rule of single contraction between the last dimension of the first matrix and the first dimension of the second matrix. For example, if $\boldsymbol{W} = \boldsymbol{U}\frac{\partial \boldsymbol{M}}{\partial \boldsymbol{z}}$, then $\boldsymbol{W}$ is a $\alpha \times \gamma \times \nu$ matrix with its $(i_1, j_2, l)^{th}$ element calculated by:

$$W_{i_1 j_2 l} = \sum_{i_2=1}^{\beta} U_{i_1 i_2} (\frac{\partial M}{\partial z})_{i_2 j_2 l} \tag{2.36}$$

Similarly, if $\boldsymbol{M}$ is a $\beta \times 1$ vector, the product rule is given as:

$$\frac{\partial \boldsymbol{UM}}{\partial \boldsymbol{z}} = \frac{\partial \boldsymbol{U}}{\partial \boldsymbol{z}}^{T_t} \boldsymbol{M} + \boldsymbol{U}\frac{\partial \boldsymbol{M}}{\partial \boldsymbol{z}} \tag{2.37}$$

where $\frac{\partial \boldsymbol{UM}}{\partial \boldsymbol{z}}$ is a $\alpha \times \nu$ Jacobian matrix and $\frac{\partial \boldsymbol{M}}{\partial \boldsymbol{z}}$ is a $\beta \times \nu$ Jacobian matrix.

Proof: Equation (2.35) is proved in the following relying on the concept of tensors. Referring to the Chapter 1 of [29], a matrix of any number of dimensions has a corresponding tensor form of the same number of orders. The matrix $\boldsymbol{U}$ may be written as a second order tensor using the Einstein summation notation: $U_{i_1 i_2} \boldsymbol{e_{i_1}} \otimes \boldsymbol{e_{i_2}} (i_1 = 1, ..., \alpha, i_2 = 1, ..., \beta)$, where $U_{i_1 i_2}$ is the $(i_1, i_2)^{th}$ element of the tensor, $\boldsymbol{e_{i_1}}$ and

$\boldsymbol{e_{i_2}}$ are directional unit vectors, $\otimes$ is the tensor product. Similarly, $\boldsymbol{M}$ and $\boldsymbol{z}$ can be written as $M_{j_1 j_2}\boldsymbol{e_{j_1}} \otimes \boldsymbol{e_{j_2}} (j_1 = 1, ..., \beta, j_2 = 1, ..., \gamma)$ and $z_l \boldsymbol{e_l} (l = 1, ..., \nu)$ respectively. The correspondence of the matrix multiplication, e.g. $\boldsymbol{UM}$, in the tensor field is namely the dot product following the rule of tensor single contraction:

$$(U_{i_1 i_2}\boldsymbol{e_{i_1}} \otimes \boldsymbol{e_{i_2}}) \cdot (M_{j_1 j_2}\boldsymbol{e_{j_1}} \otimes \boldsymbol{e_{j_2}}) = U_{i_1 i_2} M_{i_2 j_2}\boldsymbol{e_{i_1}} \otimes \boldsymbol{e_{j_2}} \tag{2.38}$$

The gradient of the resulting tensor (2.38), which is the tensor form of the left hand side of equation (2.35) i.e. $\frac{\partial \boldsymbol{UM}}{\partial \boldsymbol{z}}$, can then be written as:

$$\frac{\partial U_{i_1 i_2} M_{i_2 j_2}\boldsymbol{e_{i_1}} \otimes \boldsymbol{e_{j_2}}}{\partial z_l} \otimes \boldsymbol{e_l} = \frac{\partial U_{i_1 i_2} M_{i_2 j_2}}{\partial z_l}\boldsymbol{e_{i_1}} \otimes \boldsymbol{e_{j_2}} \otimes \boldsymbol{e_l} \tag{2.39}$$

Applying the standard product rule for the derivative of product between scalars yields:

$$\frac{\partial U_{i_1 i_2} M_{i_2 j_2}}{\partial z_l}\boldsymbol{e_{i_1}} \otimes \boldsymbol{e_{j_2}} \otimes \boldsymbol{e_l} = (\frac{\partial U_{i_1 i_2}}{\partial z_l} M_{i_2 j_2} + U_{i_1 i_2}\frac{\partial M_{i_2 j_2}}{\partial z_l})\boldsymbol{e_{i_1}} \otimes \boldsymbol{e_{j_2}} \otimes \boldsymbol{e_l} \tag{2.40}$$

Next, it aims to follow the above tensor operations to calculate the tensor form of the right hand side of equation (2.35):

$$\big((\frac{\partial U_{i_1 i_2}}{\partial z_l}\boldsymbol{e_{i_1}} \otimes \boldsymbol{e_{i_2}} \otimes \boldsymbol{e_l})^{T_t} \cdot M_{j_1 j_2}\boldsymbol{e_{j_1}} \otimes \boldsymbol{e_{j_2}}\big)^{T_t} + U_{i_1 i_2}\boldsymbol{e_{i_1}} \otimes \boldsymbol{e_{i_2}} \cdot \frac{\partial M_{j_1 j_2}}{\partial z_l}\boldsymbol{e_{j_1}} \otimes \boldsymbol{e_{j_2}} \otimes \boldsymbol{e_l} \tag{2.41}$$

The earlier defined transpose 'T_t' is used as:

$$(\frac{\partial U_{i_1 i_2}}{\partial z_l}\boldsymbol{e_{i_1}} \otimes \boldsymbol{e_{i_2}} \otimes \boldsymbol{e_l})^{T_t} = \frac{\partial U_{i_1 i_2}}{\partial z_l}\boldsymbol{e_{i_1}} \otimes \boldsymbol{e_l} \otimes \boldsymbol{e_{i_2}} \tag{2.42}$$

Thus continuing from expression (2.41), the tensor form of the right hand side of (2.35) can be simplified as:

$$\begin{aligned}
&\big(\frac{\partial U_{i_1 i_2}}{\partial z_l}\boldsymbol{e_{i_1}} \otimes \boldsymbol{e_l} \otimes \boldsymbol{e_{i_2}} \cdot M_{j_1 j_2}\boldsymbol{e_{j_1}} \otimes \boldsymbol{e_{j_2}}\big)^{T_t} + \\
&U_{i_1 i_2}\boldsymbol{e_{i_1}} \otimes \boldsymbol{e_{i_2}} \cdot \frac{\partial M_{j_1 j_2}}{\partial z_l}\boldsymbol{e_{j_1}} \otimes \boldsymbol{e_{j_2}} \otimes \boldsymbol{e_l} \\
=&(\frac{\partial U_{i_1 i_2}}{\partial z_l} M_{i_2 j_2}\boldsymbol{e_{i_1}} \otimes \boldsymbol{e_l} \otimes \boldsymbol{e_{j_2}})^{T_t} + U_{i_1 i_2}\frac{\partial M_{i_2 j_2}}{\partial z_l}\boldsymbol{e_{i_1}} \otimes \boldsymbol{e_{j_2}} \otimes \boldsymbol{e_l} \\
=&(\frac{\partial U_{i_1 i_2}}{\partial z_l} M_{i_2 j_2} + U_{i_1 i_2}\frac{\partial M_{i_2 j_2}}{\partial z_l})\boldsymbol{e_{i_1}} \otimes \boldsymbol{e_{j_2}} \otimes \boldsymbol{e_l}
\end{aligned} \tag{2.43}$$

which coincides with expression (2.40). Therefore the tensor forms of the $L.H.S$ and $R.H.S$ of equation (2.35) are equal and thus (2.35) is proved. Following the same way, equation (2.37) when $\boldsymbol{M}$ is a $\beta \times 1$ vector can also be proved to hold. The tensor form of the left hand side of (2.37) is written as:

$$\frac{\partial U_{i_1 i_2} M_{i_2}}{\partial z_l} \boldsymbol{e_{i_1}} \otimes \boldsymbol{e_l} = (\frac{\partial U_{i_1 i_2}}{\partial z_l} M_{i_2} + U_{i_1 i_2} \frac{\partial M_{i_2}}{\partial z_l}) \boldsymbol{e_{i_1}} \otimes \boldsymbol{e_l} \tag{2.44}$$

While the tensor form of the right hand side of (2.37) becomes:

$$(\frac{\partial U_{i_1 i_2}}{\partial z_l} \boldsymbol{e_{i_1}} \otimes \boldsymbol{e_{i_2}} \otimes \boldsymbol{e_l})^{T_t} \cdot M_{j_1} \boldsymbol{e_{j_1}} + U_{i_1 i_2} \boldsymbol{e_{i_1}} \otimes \boldsymbol{e_{i_2}} \cdot \frac{\partial M_{j_1}}{\partial z_l} \boldsymbol{e_{j_1}} \otimes \boldsymbol{e_l} \tag{2.45}$$

Expression (2.45) can be simplified as:

$$\begin{aligned} & \frac{\partial U_{i_1 i_2}}{\partial z_l} \boldsymbol{e_{i_1}} \otimes \boldsymbol{e_l} \otimes \boldsymbol{e_{i_2}} \cdot M_{j_1} \boldsymbol{e_{j_1}} + U_{i_1 i_2} \boldsymbol{e_{i_1}} \otimes \boldsymbol{e_{i_2}} \cdot \frac{\partial M_{j_1}}{\partial z_l} \boldsymbol{e_{j_1}} \otimes \boldsymbol{e_l} \\ =& \frac{\partial U_{i_1 i_2}}{\partial z_l} M_{i_2} \boldsymbol{e_{i_1}} \otimes \boldsymbol{e_l} + U_{i_1 i_2} \frac{\partial M_{i_2}}{\partial z_l} \boldsymbol{e_{i_1}} \otimes \boldsymbol{e_l} \\ =& (\frac{\partial U_{i_1 i_2}}{\partial z_l} M_{i_2} + U_{i_1 i_2} \frac{\partial M_{i_2}}{\partial z_l}) \boldsymbol{e_{i_1}} \otimes \boldsymbol{e_l} \end{aligned} \tag{2.46}$$

which coincides with expression (2.44). Therefore the tensor forms of the $L.H.S$ and $R.H.S$ of equation (2.37) are equal and thus (2.37) is proved.

2.3.2 Recursive computation of the observability matrix

Consider the submatrices $\frac{\partial \boldsymbol{C A^k x_t}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{C A^{k-1} B_i}}{\partial \boldsymbol{\theta}}$, since $\boldsymbol{CA}^k$ is a 2-dimensional matrix, $\boldsymbol{x_t}$ is a vector and $\boldsymbol{\theta}$ is a vector, $\frac{\partial \boldsymbol{C A^k x_t}}{\partial \boldsymbol{\theta}}$ is expanded according to equation (2.37):

$$\frac{\partial \boldsymbol{C A^k x_t}}{\partial \boldsymbol{\theta}} = (\frac{\partial \boldsymbol{C A^k}}{\partial \boldsymbol{\theta}})^{T_t} \boldsymbol{x_t} + \boldsymbol{C A^k} \frac{\partial \boldsymbol{x_t}}{\partial \boldsymbol{\theta}} \tag{2.47}$$

$\frac{\partial \boldsymbol{x_t}}{\partial \boldsymbol{\theta}}$ is a zero matrix due to the dynamic states and the parameters being independent, and therefore equation (2.47) becomes:

$$\frac{\partial \boldsymbol{C A^k x_t}}{\partial \boldsymbol{\theta}} = (\frac{\partial \boldsymbol{C A^k}}{\partial \boldsymbol{\theta}})^{T_t} \boldsymbol{x_t} \tag{2.48}$$

Expressing $\boldsymbol{CA}^k$ as $\boldsymbol{CA}^{k-1}\boldsymbol{A}$, $\frac{\partial \boldsymbol{CA}^k}{\partial \boldsymbol{\theta}}$ is expanded based on equation (2.35):

$$\frac{\partial \boldsymbol{CA}^k}{\partial \boldsymbol{\theta}} = \big((\frac{\partial \boldsymbol{CA}^{k-1}}{\partial \boldsymbol{\theta}})^{T_t}\boldsymbol{A}\big)^{T_t} + \boldsymbol{CA}^{k-1}\frac{\partial \boldsymbol{A}}{\partial \boldsymbol{\theta}} \tag{2.49}$$

Taking the transpose 'T_t' of the both sides of equation (2.49) gives:

$$(\frac{\partial \boldsymbol{CA}^k}{\partial \boldsymbol{\theta}})^{T_t} = (\frac{\partial \boldsymbol{CA}^{k-1}}{\partial \boldsymbol{\theta}})^{T_t}\boldsymbol{A} + (\boldsymbol{CA}^{k-1}\frac{\partial \boldsymbol{A}}{\partial \boldsymbol{\theta}})^{T_t} \tag{2.50}$$

Similarly, $\frac{\partial \boldsymbol{CA}^{k-1}\boldsymbol{B_i}}{\partial \boldsymbol{\theta}}$ is expanded using (2.37):

$$\frac{\partial \boldsymbol{CA}^{k-1}\boldsymbol{B_i}}{\partial \boldsymbol{\theta}} = (\frac{\partial \boldsymbol{CA}^{k-1}}{\partial \boldsymbol{\theta}})^{T_t}\boldsymbol{B_i} + \boldsymbol{CA}^{k-1}\frac{\partial \boldsymbol{B_i}}{\partial \boldsymbol{\theta}} \tag{2.51}$$

Equations (2.48), (2.50) and (2.51) can then be used to recursively update the sub-matrices $\frac{\partial \boldsymbol{CA}^k\boldsymbol{x_t}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{CA}^{k-1}\boldsymbol{B_i}}{\partial \boldsymbol{\theta}}$, i.e. $\frac{\partial \boldsymbol{CA}^k\boldsymbol{x_t}}{\partial \boldsymbol{\theta}}$ at step k is computed from $(\frac{\partial \boldsymbol{CA}^k}{\partial \boldsymbol{\theta}})^{T_t}$ at step k based on equation (2.48); $(\frac{\partial \boldsymbol{CA}^k}{\partial \boldsymbol{\theta}})^{T_t}$ at step k and $\frac{\partial \boldsymbol{CA}^{k-1}\boldsymbol{B_i}}{\partial \boldsymbol{\theta}}$ at step $k-1$ are computed from $(\frac{\partial \boldsymbol{CA}^{k-1}}{\partial \boldsymbol{\theta}})^{T_t}$ at step $k-1$ and $\boldsymbol{CA}^{k-1}$ at step $k-1$ based on equations (2.50) and (2.51) respectively. The constant terms during the recursion, including $\frac{\partial \boldsymbol{A}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{B_i}}{\partial \boldsymbol{\theta}}$, can be pre-computed at the initialization step.

For clarity of the presentation, the symbols $\boldsymbol{a_k}$, $\boldsymbol{b_k}$, $\boldsymbol{c_{k-1}}$ and $\boldsymbol{d_k}$ are introduced to denote $\boldsymbol{CA}^k$, $\frac{\partial \boldsymbol{CA}^k\boldsymbol{x_t}}{\partial \boldsymbol{\theta}}$, $\frac{\partial \boldsymbol{CA}^{k-1}\boldsymbol{B_i}}{\partial \boldsymbol{\theta}}$ and $(\frac{\partial \boldsymbol{CA}^k}{\partial \boldsymbol{\theta}})^{T_t}$ respectively. At step k, the matrices in equation (2.33) are computed by:

$$\begin{aligned} \frac{\partial \boldsymbol{L_f^k h}}{\partial \boldsymbol{x}} &= \begin{bmatrix} \boldsymbol{a_k} & \boldsymbol{b_k} \end{bmatrix} \\ \frac{\partial \boldsymbol{L_{g_i} L_f^{k-1} h}}{\partial \boldsymbol{x}} &= \begin{bmatrix} \boldsymbol{0}_{p\times n_t} & \boldsymbol{c_{k-1}} \end{bmatrix} \end{aligned} \tag{2.52}$$

where $\boldsymbol{a_k}$ is computed by:

$$\boldsymbol{a_k} = \boldsymbol{a_{k-1}}\boldsymbol{A} \tag{2.53}$$

$\boldsymbol{b_k}$ is computed by equation (2.48):

$$\boldsymbol{b_k} = \boldsymbol{d_k}\boldsymbol{x_t} \tag{2.54}$$

$\boldsymbol{d_k}$ is computed by equation (2.50):

$$\boldsymbol{d_k} = \boldsymbol{d_{k-1}}\boldsymbol{A} + (\boldsymbol{a_{k-1}}\frac{\partial \boldsymbol{A}}{\partial \boldsymbol{\theta}})^{T_t} \tag{2.55}$$

and $\boldsymbol{c_{k-1}}$ is computed by equation (2.51):

$$\boldsymbol{c_{k-1}} = \boldsymbol{d_{k-1}}\boldsymbol{B_i} + \boldsymbol{a_{k-1}}\frac{\partial \boldsymbol{B_i}}{\partial \boldsymbol{\theta}} \tag{2.56}$$

with $\boldsymbol{a_0} = C$ and $\boldsymbol{d_0} = (\frac{\partial \boldsymbol{C}}{\partial \boldsymbol{\theta}})^{T_t}$ at the initial step. A flowchart diagram is presented in Figure 2.1 showing the recursive formulation can be used as an algorithm to output the observability matrix of the system in equations (2.24) and (2.25).

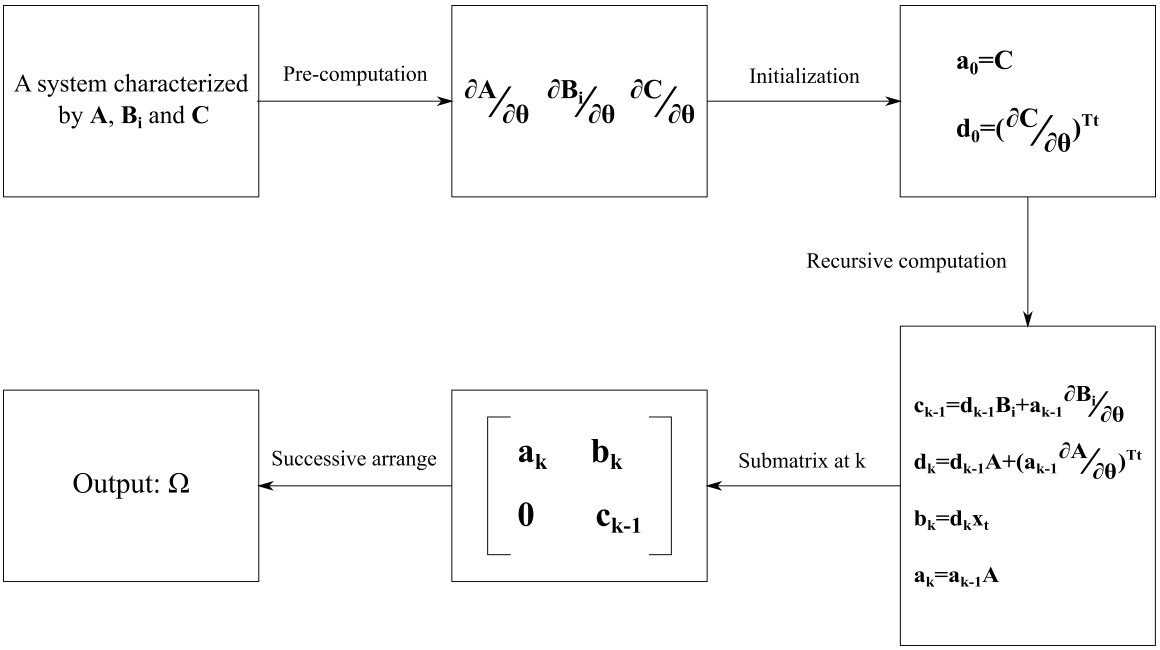

Figure 2.1: Recursive computation of the observability matrix

2.4 Robust algorithm

2.4.1 Numerical realization

The previously derived algorithm can be implemented symbolically using symbolic computing tools such as MATLAB symbolic toolbox [71], Mathematica [110] and

Maple [69]. On the other hand, an advantage of avoiding symbolic differentiations, with the exception of the pre-computation in Figure 2.1, is that numerical realization of the algorithm becomes possible. This can be achieved by specializing the dynamic states and the parameters on some randomly chosen values: $\boldsymbol{x_t} = \tilde{\boldsymbol{x}}_{\boldsymbol{t}}$, $\boldsymbol{\theta} = \tilde{\boldsymbol{\theta}}$. As a result, $\boldsymbol{A}(\tilde{\boldsymbol{\theta}})$, $\boldsymbol{B_i}(\tilde{\boldsymbol{\theta}})$, $\boldsymbol{C}(\tilde{\boldsymbol{\theta}})$, $\frac{\partial \boldsymbol{A}}{\partial \boldsymbol{\theta}}(\tilde{\boldsymbol{\theta}})$, $\frac{\partial \boldsymbol{B_i}}{\partial \boldsymbol{\theta}}(\tilde{\boldsymbol{\theta}})$ and $\frac{\partial \boldsymbol{C}}{\partial \boldsymbol{\theta}}(\tilde{\boldsymbol{\theta}})$ at the initialization are all numerical matrices, the following matrix operations are all numerical, and consequently the algorithm is expected to output a numerical observability matrix $\boldsymbol{\Omega}(\tilde{\boldsymbol{x}}_{\boldsymbol{t}}, \tilde{\boldsymbol{\theta}})$.

For observability problems, one is interested in the generic rank rather than reduced ranks of the observability matrix that might occur at certain realizations of the states corresponding to singular points. Such singular points are more relevant for applications involving controllers [67]. Given that it is highly unlikely to pick those singular points in a random process, the rank of $\boldsymbol{\Omega}(\tilde{\boldsymbol{x}}_{\boldsymbol{t}}, \tilde{\boldsymbol{\theta}})$ is expected to be the same as the generic rank of the corresponding symbolic matrix $\boldsymbol{\Omega}(\boldsymbol{x_t}, \boldsymbol{\theta})$. A similar discussion can be found in [86] for development of a semi-numerical algorithm to test the observability of nonlinear algebraic systems, where the initial conditions of dynamic states and parameters of the systems are specialized on random integers. To summarize, a complete description of the proposed algorithm with numerical realizations is presented in the following as Algorithm 2.2.

Algorithm 2.2

Input to the algorithm: the matrices $\boldsymbol{A}, \boldsymbol{B_1}, ..., \boldsymbol{B_m}$ and $\boldsymbol{C}$, the dynamic states $\boldsymbol{x_t}$ and the parameters $\boldsymbol{\theta}$ of the system in equation (2.24)

Output from the algorithm: the observability matrix $\boldsymbol{\Omega}$

Preprocessing: Compute $\frac{\partial \boldsymbol{A}}{\partial \boldsymbol{\theta}}, \frac{\partial \boldsymbol{B_1}}{\partial \boldsymbol{\theta}}, ..., \frac{\partial \boldsymbol{B_m}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{C}}{\partial \boldsymbol{\theta}}$ symbolically

Initialization: Set $k = 0$. Choose random numerical realizations for $\boldsymbol{x_t} = \tilde{\boldsymbol{x}}_{\boldsymbol{t}}$, $\boldsymbol{\theta} = \tilde{\boldsymbol{\theta}}$

where $\tilde{\boldsymbol{x}}_{\boldsymbol{t}}, \tilde{\boldsymbol{\theta}} \in \mathbb{R}$. Let $\boldsymbol{a_0} = \boldsymbol{C}$, $\boldsymbol{d_0} = (\frac{\partial \boldsymbol{C}}{\partial \boldsymbol{\theta}})^{T_t}$, $\boldsymbol{b_0} = \boldsymbol{d_0 x_t}$, $\boldsymbol{\Omega_0} = \begin{bmatrix} \boldsymbol{a_0} & \boldsymbol{b_0} \end{bmatrix}$

1. From Eq. (2.56), compute $\boldsymbol{c_{k,1}} = \boldsymbol{d_k B_1} + \boldsymbol{a_k} \frac{\partial \boldsymbol{B_1}}{\partial \boldsymbol{\theta}}$, ..., $\boldsymbol{c_{k,m}} = \boldsymbol{d_k B_m} + \boldsymbol{a_k} \frac{\partial \boldsymbol{B_m}}{\partial \boldsymbol{\theta}}$

2. From Eq. (2.55), compute $\boldsymbol{d_{k+1}} = \boldsymbol{d_k A} + (\boldsymbol{a_k} \frac{\partial \boldsymbol{A}}{\partial \boldsymbol{\theta}})^{T_t}$

3. From Eq. (2.54), compute $\boldsymbol{b_{k+1}} = \boldsymbol{d_{k+1} x_t}$

4. From Eq. (2.53), compute $\boldsymbol{a_{k+1}} = \boldsymbol{a_k A}$

5. Arrange the submatrices from steps 1, 2, 3 and 4 in the observability matrix $\boldsymbol{\Omega_{k+1}} = \boldsymbol{\Omega_k} \cup \begin{bmatrix} \boldsymbol{a_{k+1}} & \boldsymbol{b_{k+1}} \\ \boldsymbol{0}_{p \times n_t} & \boldsymbol{c_{k,1}} \\ \vdots & \vdots \\ \boldsymbol{0}_{p \times n_t} & \boldsymbol{c_{k,m}} \end{bmatrix}$

6. Compute the rank, if $rank(\boldsymbol{\Omega_{k+1}}) = rank(\boldsymbol{\Omega_k})$, or $rank(\boldsymbol{\Omega_{k+1}}) = n$, or $k = n - 2$, end and output $\boldsymbol{\Omega} = \boldsymbol{\Omega_{k+1}}$

7. Let $k = k + 1$ and go to step 1

2.4.2 Symbolic numbers vs. Floating-point numbers

In terms of the type of numerical operations in a computing system, using floating-point numbers is the most efficient way to process the algorithm. However, a drawback is that floating-points are of limited precision to represent real numbers. The errors brought by the lost precision would propagate and accumulate through the recursive computation and can eventually exert a significant impact on the numerical rank of the observability matrix. Moreover, floating-point numbers have to be limited within a range defined by the maximum and minimum representable numbers. When the algorithm is implemented for large systems, the occurring numbers during the recursion can easily exceed those limits.

Symbolic numbers: A considerable alternative is to use symbolic numbers instead of floating-point numbers as symbolic numbers are not only always exact but

also infinitely representable. Operations between symbolic numbers are substantially cheaper than operations between symbols. Nevertheless, processing of symbolic numbers is less efficient than that of floating point numbers especially for large systems.

Floating-point integers with application of modular operations: The drawbacks of using floating-point numbers, related to the issues of limited precision and number growth, can be effectively alleviated by using integers, i.e. $\tilde{\boldsymbol{x}}_{\boldsymbol{t}}, \tilde{\boldsymbol{\theta}} \in \mathbb{Z}$, with application of modular operations. The modular operation of a rational number a, i.e. an integer or a fraction two integers, using a positive prime number p is denoted by $a \, mod \, p$. The rule of modular arithmetic can be simply described as follows [90]: assuming a is an integer, $a \, mod \, p$ takes the remainder of a divided by p, and the result is a positive integer between 0 and $p-1$; if the greatest common divisor of a and p is 1, then based on the rule of modular multiplicative inverse, $a^{-1} \, mod \, p$ is a positive integer b between 1 and $p-1$ such that $ab \, mod \, p = 1$. To perform the modular operations on the algorithm, a large prime number p is selected and a set of integers between 1 and $p-1$ are randomly chosen for $\tilde{\boldsymbol{x}}_{\boldsymbol{t}}$ and $\tilde{\boldsymbol{\theta}}$. Then all the involved computations in the algorithm are operated by $mod \, p$. Let $\boldsymbol{a_{k,p}}$, $\boldsymbol{b_{k,p}}$, $\boldsymbol{c_{k,p}}$ and $\boldsymbol{d_{k,p}}$ denote the recursive terms after applying the modular operations to equations (2.56), (2.55), (2.54) and (2.53),

$$
\begin{aligned}
\boldsymbol{c_{k-1,p}} &= \big((\boldsymbol{d_{k-1,p}} \boldsymbol{B_i}) \, mod \, p + (\boldsymbol{a_{k-1,p}} \frac{\partial \boldsymbol{B_i}}{\partial \boldsymbol{\theta}}) \, mod \, p\big) \, mod \, p \\
\boldsymbol{d_{k,p}} &= \big((\boldsymbol{d_{k-1,p}} \boldsymbol{A}) \, mod \, p + (\boldsymbol{a_{k-1,p}} \frac{\partial \boldsymbol{A}}{\partial \boldsymbol{\theta}})^{T_t} \, mod \, p\big) \, mod \, p \\
\boldsymbol{b_{k,p}} &= (\boldsymbol{d_{k,p}} \boldsymbol{x_t}) \, mod \, p \\
\boldsymbol{a_{k,p}} &= (\boldsymbol{a_{k-1,p}} \boldsymbol{A}) \, mod \, p
\end{aligned}
\tag{2.57}
$$

where the modulo operations are applied to the matrices elementwise.

The following distributive law [90] is used to account for the usefulness of the

modular operations on the algorithm:

$$\begin{aligned}(E \bmod p + F \bmod p) \bmod p &= (E + F) \bmod p \\ (E \bmod p)(F \bmod p) \bmod p &= EF \bmod p\end{aligned} \tag{2.58}$$

where the law holds either when E and F are rational numbers or matrices with rational elements. Based on equation (2.58), equation (2.57) is equivalent to:

$$\begin{aligned}\boldsymbol{c_{k-1,p}} &= (\boldsymbol{d_{k-1,p}B_i} + \boldsymbol{a_{k-1,p}}\frac{\partial \boldsymbol{B_i}}{\partial \boldsymbol{\theta}}) \ \ mod\, p \\ \boldsymbol{d_{k,p}} &= (\boldsymbol{d_{k-1,p}A} + (\boldsymbol{a_{k-1,p}}\frac{\partial \boldsymbol{A}}{\partial \boldsymbol{\theta}})^{T_t}) \ \ mod\, p \\ \boldsymbol{b_{k,p}} &= (\boldsymbol{d_{k,p}x_t}) \ \ mod\, p \\ \boldsymbol{a_{k,p}} &= (\boldsymbol{a_{k-1,p}A}) \ \ mod\, p\end{aligned} \tag{2.59}$$

and consequently, it can be deduced that the occurring observability matrix from the algorithm using modular operations, denoted by $\boldsymbol{\Omega_p}$, satisfies the following property:

$$\boldsymbol{\Omega_p} = \boldsymbol{\Omega} \ \ mod\, p \tag{2.60}$$

The observability of a tested system is then determined by evaluating the rank of $\boldsymbol{\Omega_p}$ over a finite field $\mathbb{F}_p$ through applying modular operations to Gaussian elimination [86, 90, 14, 91]. As discussed in [86], if $\boldsymbol{\Omega_p}$ is a full-rank matrix, $\boldsymbol{\Omega}$ must also be full-rank, both observability matrices agreeing to imply the tested system is observable. If $\boldsymbol{\Omega_p}$ is not of full-rank, it is highly likely that $\boldsymbol{\Omega}$ is not of full-rank either. An exception where $\boldsymbol{\Omega}$ would be of full-rank despite $\boldsymbol{\Omega_p}$ not being of full-rank may occur when the nonzero determinant of the largest square submatrix of $\boldsymbol{\Omega}$ vanishes modulo p (if $\boldsymbol{\Omega}$ is a square matrix, for example, the case would be $det(\boldsymbol{\Omega}) \bmod p = 0$). [86] suggested an approach to estimate the upper bound of the probability of this case where $\boldsymbol{\Omega_p}$ and $\boldsymbol{\Omega}$ deliver different rank information, and it was shown that the likelihood of this occurring becomes practically negligible if the prime number p is selected to be large.

Therefore, $\boldsymbol{\Omega_p}$ is capable of accurately describing the observability of a tested system with a very high probability of success.

The usefulness of the modular operations allows for proposing Algorithm 2.3 which makes use of floating-point integers for numerical realizations.

Algorithm 2.3

Input to the algorithm: the matrices $\boldsymbol{A}, \boldsymbol{B_1}, ..., \boldsymbol{B_m}$ and $\boldsymbol{C}$, the dynamic states $\boldsymbol{x_t}$ and the parameters $\boldsymbol{\theta}$ of the system in equation (2.24)

Output from the algorithm: the observability matrix $\boldsymbol{\Omega_p}$

Preprocessing: Compute $\frac{\partial \boldsymbol{A}}{\partial \boldsymbol{\theta}}, \frac{\partial \boldsymbol{B_1}}{\partial \boldsymbol{\theta}}, ..., \frac{\partial \boldsymbol{B_m}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{C}}{\partial \boldsymbol{\theta}}$ symbolically

Initialization: Set $k = 0$. Select a large positive prime number p. Choose random numerical realizations for $\boldsymbol{x_t} = \tilde{\boldsymbol{x}}_{\boldsymbol{t}}$, $\boldsymbol{\theta} = \tilde{\boldsymbol{\theta}}$ where $\tilde{\boldsymbol{x}}_{\boldsymbol{t}}, \tilde{\boldsymbol{\theta}} \in \{1, ..., p-1\}$ in a floating-point format. Let $\boldsymbol{a_{0,p}} = \boldsymbol{C}$, $\boldsymbol{d_{0,p}} = (\frac{\partial \boldsymbol{C}}{\partial \boldsymbol{\theta}})^{T_t}$, $\boldsymbol{b_{0,p}} = \boldsymbol{d_{0,p}} \boldsymbol{x_t}$, $\boldsymbol{\Omega_{0,p}} = \begin{bmatrix} \boldsymbol{a_{0,p}} & \boldsymbol{b_{0,p}} \end{bmatrix}$. All the matrices $mod\, p$

1. Based on Eq.(2.57), compute

$\boldsymbol{c_{k,p,i}} = \left((\boldsymbol{d_{k,p}} \boldsymbol{B_i})\, mod\, p + (\boldsymbol{a_{k,p}} \frac{\partial \boldsymbol{B_i}}{\partial \boldsymbol{\theta}})\, mod\, p \right) mod\, p \ (i = 1, ..., m)$

2. $\boldsymbol{d_{k+1,p}} = \left((\boldsymbol{d_{k,p}} \boldsymbol{A})\, mod\, p + (\boldsymbol{a_{k,p}} \frac{\partial \boldsymbol{A}}{\partial \boldsymbol{\theta}})^{T_t}\, mod\, p \right) mod\, p$

3. $\boldsymbol{b_{k+1,p}} = (\boldsymbol{d_{k+1,p}} \boldsymbol{x_t})\, mod\, p$

4. $\boldsymbol{a_{k+1,p}} = (\boldsymbol{a_{k,p}} \boldsymbol{A})\, mod\, p$

5. Arrange the submatrices from steps 1, 2, 3 and 4 in the observability matrix $\boldsymbol{\Omega_{k+1,p}} = \boldsymbol{\Omega_{k,p}} \cup \begin{bmatrix} \boldsymbol{a_{k+1,p}} & \boldsymbol{b_{k+1,p}} \\ \boldsymbol{0}_{p \times n_t} & \boldsymbol{c_{k,p,1}} \\ \vdots & \vdots \\ \boldsymbol{0}_{p \times n_t} & \boldsymbol{c_{k,p,m}} \end{bmatrix} mod\, p$

6. Compute the rank over $\mathbb{F}_p$, if $rank(\boldsymbol{\Omega_{k+1,p}}) = rank(\boldsymbol{\Omega_{k,p}})$, or $rank(\boldsymbol{\Omega_{k+1,p}}) = n$, or $k = n - 2$, end and output $\boldsymbol{\Omega_p} = \boldsymbol{\Omega_{k+1,p}}$

7. Let $k = k + 1$ and go to step 1

Note that Algorithm 2.2 through the use of symbolic numbers is more computationally expensive than Algorithm 2.3, although the latter comes at the price of a very small probability of predicting a wrong rank for the observability matrix. This issue can be alleviated by repeating the numerical realizations using different values of $\tilde{\boldsymbol{x}}_{\boldsymbol{t}}$ and $\tilde{\boldsymbol{\theta}}$ and the selection of p. A second limitation of Algorithm 2.3 is that the elements of the matrices $\boldsymbol{A}$, $\boldsymbol{B_i}$ and $\boldsymbol{C}$ must be rational functions of $\boldsymbol{\theta}$ i.e. polynomial functions or fractions of polynomial functions, while for Algorithm 2.2 those functions can be analytic with respect to $\boldsymbol{\theta}$.

2.5 Illustrative examples

2.5.1 Example 1: N degrees of freedom mass-spring system

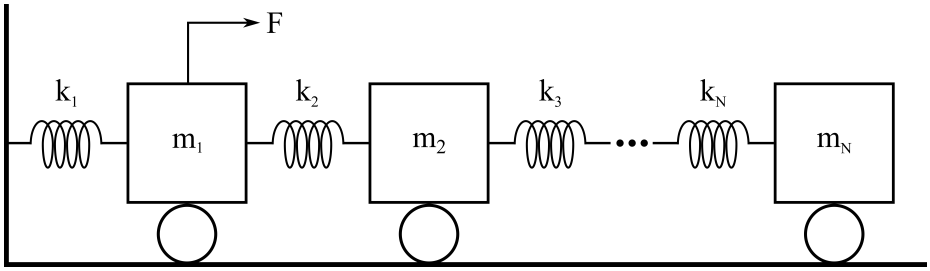

Figure 2.2: N degrees of freedom mass-spring system

The purpose of this example is to illustrate the use of the proposed Algorithm 2.2 and Algorithm 2.3, and compare their efficiency versus the standard implementation of the ORC i.e. Algorithm 2.1. The dynamical system shown in Figure 2.2 is a N degrees of freedom mass-spring system with the i^{th} mass and spring stiffness denoted by m_i and k_i respectively. The displacement and velocity of m_i are denoted by x_i and $\dot{x}_i$ respectively. The dynamical system is subject to a measured force F applied on the first mass. Suppose all of the masses and stiffness are to be identified,

and the displacements and velocities of the system are to be tracked using a set of measurements. $\boldsymbol{\theta}$ is then written as:

$$\boldsymbol{\theta} = \begin{bmatrix} k_1 & k_2 & ... & k_N & m_1 & m_2 & ... & m_N \end{bmatrix}^T \tag{2.61}$$

and $\boldsymbol{x_t}$ is:

$$\boldsymbol{x_t} = \begin{bmatrix} x_1 & x_2 & ... & x_N & \dot{x}_1 & \dot{x}_2 & ... & \dot{x}_N \end{bmatrix}^T \tag{2.62}$$

The system matrix $\boldsymbol{A}$ is of the form:

$$\boldsymbol{A} = \begin{bmatrix} \mathbf{0}_{N\times N} & \boldsymbol{I}_{N\times N} \\ -\boldsymbol{M}^{-1}\boldsymbol{K} & \mathbf{0}_{N\times N} \end{bmatrix} \tag{2.63}$$

where $\boldsymbol{M}$ is the diagonal mass matrix and $\boldsymbol{K}$ is the stiffness matrix:

$$\boldsymbol{M} = \begin{bmatrix} m_1 & 0 & \dots & 0 \\ 0 & m_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & m_N \end{bmatrix}, \boldsymbol{K} = \begin{bmatrix} k_1+k_2 & -k_2 & 0 & \dots & 0 \\ -k_2 & k_2+k_3 & -k_3 & \dots & 0 \\ 0 & -k_3 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & k_{N-1}+k_N & -k_N \\ 0 & 0 & \dots & -k_N & k_N \end{bmatrix} \tag{2.64}$$

Taking into account there is only a single force applied on m_1, the input matrix $\boldsymbol{B_i}$ with $i = 1$ is obtained as:

$$\boldsymbol{B_1} = \begin{bmatrix} \mathbf{0}_{1\times N} & \frac{1}{m_1} & \mathbf{0}_{1\times(N-1)} \end{bmatrix}^T \tag{2.65}$$

The form of the output matrix $\boldsymbol{C}$ depends on what the output measurements are. For example, if the measurement is the displacement of m_i, i.e. x_i, then $\boldsymbol{C}$ is expressed as:

$$\boldsymbol{C} = \begin{bmatrix} \mathbf{0}_{1\times(i-1)} & 1 & \mathbf{0}_{1\times(2N-i)} \end{bmatrix} \tag{2.66}$$

Three degrees of freedom system

First, the observability of a three degrees of freedom system, i.e. $N = 3$, is tested. This system has already been studied in [26] using Algorithm 2.1 and the results

are used for comparison herein. The observability of the system is investigated for three different measurement scenarios where in each the displacement of a single mass is measured, i.e. x_1, x_2 and x_3. The output results from Algorithm 2.2 and Algorithm 2.3 are presented in Table 2.1 where the observability and identifiability of the individual state is detected through rank testing after obtaining the observability matrix, as described in Section 2.1.2. The results from Algorithm 2.1, Algorithm 2.2 and Algorithm 2.3 are in full agreement and hence provide users with exactly the same observability information of the system.

	Algorithm 2.2			Algorithm 2.3		
Measurements	x_1	x_2	x_3	x_1	x_2	x_3
Rank(Ω)	12	11	10	12	11	10
Observable states	x_1, x_2, x_3, $\dot{x}_1$, $\dot{x}_2$, $\dot{x}_3$, k_1, k_2, k_3, m_1, m_2, m_3	x_2, x_3, $\dot{x}_2$, $\dot{x}_3$, k_1	x_3, $\dot{x}_3$, k_1	x_1, x_2, x_3, $\dot{x}_1$, $\dot{x}_2$, $\dot{x}_3$, k_1, k_2, k_3, m_1, m_2, m_3	x_2, x_3, $\dot{x}_2$, $\dot{x}_3$, k_1	x_3, $\dot{x}_3$, k_1
Unobservable states	-	x_1, $\dot{x}_1$, k_2, k_3, m_1, m_2, m_3	x_1, x_2, $\dot{x}_1$, $\dot{x}_2$, k_2, k_3, m_1, m_2, m_3	-	x_1, $\dot{x}_1$, k_2, k_3, m_1, m_2, m_3	x_1, x_2, $\dot{x}_1$, $\dot{x}_2$, k_2, k_3, m_1, m_2, m_3

Table 2.1: The observability of the 3 DOFs mass-spring system using Algorithm 2.2 and Algorithm 2.3

Large systems

Having built the confidence in the proposed algorithms, the observability tests are performed on the system in Figure 2.2 with an increasing number of masses: $N = 10, 20, 40, 60, 80, 100, 200, 500,$ in order to compare the capability and efficiency of Algorithm 2.1, Algorithm 2.2 and Algorithm 2.3. All the algorithms are implemented in MATLAB R2016a on a desktop computer with Core i7-6700 CPU (3.40GHz) and 16GB RAM. The elapsed time of implementation is recorded in seconds for each test, as shown in Table 2.2. The elapsed time does not count the time of preprocessing since the symbolic computations of $\frac{\partial \boldsymbol{A}}{\partial \boldsymbol{\theta}}, \frac{\partial \boldsymbol{B_1}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{C}}{\partial \boldsymbol{\theta}}$ are common for the proposed

N	Algorithm 2.1	Algorithm 2.2	Algorithm 2.3
7	5698.73	0.46	0.02
20	-	6.57	0.09
40	-	45.84	1.18
60	-	154.49	6.19
80	-	393.71	19.11
100	-	884.04	46.74
200	-	16888.09	733.56
500	-	-	23014.22

Table 2.2: The elapsed time of observability testing (seconds) for large systems

algorithms. If a test is not completed within 48 hours or the computer runs out of memory, the result is indicated by the notation '-'.

The displacement x_1 is used as the measurement for all the systems and the force is always assumed to be applied on the first mass. It is worth mentioning that Franco et al. [40] have theoretically proved that a mass-spring system with any number of DOFs is observable when i) the excitation is located at the first mass m_1 and the measurement is the displacement x_1, ii) the excitation is located at the last mass m_N and the measurement is the displacement x_N or iii) the excitation is located at any mass m_i and the measurements are the displacements x_i and x_{i-1}. The scenarios considered herein fall within case i).

The observability results of all successful implementations are that the system is observable as expected for this case. As can be seen in Table 2.2, Algorithm 2.1 successfully tests a system of 7 DOFs, but it is unable to give any results of observability for the systems with 20 and hence more DOFs within 48 hours. In comparison, the proposed algorithm either using symbolic numbers or floating-point integers is capable of dealing with a large system with several hundreds of DOFs within an acceptable period of time. When implementing Algorithm 2.2 using symbolic numbers,

the elapsed time grows significantly with the size of the system. Algorithm 2.3 is more efficient due to fast processing of floating-points in a computer, and therefore it is applicable to larger systems.

The reported efficiency of implementations might be further improved e.g. by parallelizing the algorithms, but the focus of this study is on the relative differences between the algorithms. It should be mentioned that the main limitation with Algorithm 2.1 is its high RAM requirements. The implementations of Algorithm 2.2 and Algorithm 2.3 are efficient in the usage of RAM and thus can avoid problems with exceeding the RAM available to MATLAB. This then allows Algorithm 2.2 and Algorithm 2.3 to improve their efficiency by making use of more cores in parallel in the computer. However, it is difficult for Algorithm 2.1 to improve if the RAM requirements are the critical limiting factor.

Viscously damped system

In this part, the existence of rate-proportional viscous damping forces associated with a corresponding damping matrix $\boldsymbol{C_d}$ is considered for the system in Figure 2.2. $\boldsymbol{A}$ matrix of such a mass-spring-damper system becomes:

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{0}_{N\times N} & \boldsymbol{I}_{N\times N} \\ -\boldsymbol{M}^{-1}\boldsymbol{K} & -\boldsymbol{M}^{-1}\boldsymbol{C_d} \end{bmatrix} \tag{2.67}$$

Algorithm 2.2 and Algorithm 2.3 are applied to examine the observability of the system with different forms of the damping matrix, including i) a full symmetric $\boldsymbol{C_d}$, ii) $\boldsymbol{C_d}$ corresponding to having viscous dashpots between successive masses and iii) a Rayleigh, mass and stiffness proportional damping matrix. In case i), the damping

matrix can be generally written as:

$$\boldsymbol{C_d} = \begin{bmatrix} c_{11} & \dots & c_{1N} \\ & \ddots & \vdots \\ sym. & & c_{NN} \end{bmatrix} \tag{2.68}$$

where $c_{11}, ..., c_{NN}$ are the damping parameters to be identified and thus included in the vector of parameters $\boldsymbol{\theta}$. Case ii) results in:

$$\boldsymbol{C_d} = \begin{bmatrix} c_1 + c_2 & -c_2 & 0 & \dots & 0 \\ -c_2 & c_2 + c_3 & -c_3 & \dots & 0 \\ 0 & -c_3 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & c_{N-1} + c_N & -c_N \\ 0 & 0 & \dots & -c_N & c_N \end{bmatrix} \tag{2.69}$$

where the coefficients of the viscous dashpots $c_1, .., c_N$ are the parameters to be identified. The last case focuses on the form:

$$\boldsymbol{C_d} = \alpha \boldsymbol{M} + \beta \boldsymbol{K} \tag{2.70}$$

where α and β are scalar parameters to be identified.

If the system has 100 masses, i.e. $N = 100$, the occurring 100 DOFs system can be viewed as the model of a 100-floor shear building. For simplicity, it is still assumed that a single force is applied on the first floor. Note that because of the inclusion of a damping matrix whose parameters are to be identified, the conclusions on the observability of the system in [40] are not applicable anymore. Suppose ten sensors are installed at the floors following two different sensor configurations where in each five displacements and five accelerations are measured: configuration 1) the sensors are installed at the top ten floors measuring $x_{100}, x_{99}, x_{98}, x_{97}, x_{96}, \ddot{x}_{95}, \ddot{x}_{94}, \ddot{x}_{93}, \ddot{x}_{92}, \ddot{x}_{91}$ and configuration 2) the sensors are installed at every ten floors measuring $x_{100}, \ddot{x}_{90}, x_{80}, \ddot{x}_{70}, x_{60}, \ddot{x}_{50}, x_{40}, \ddot{x}_{30}, x_{20}, \ddot{x}_{10}$. Using the two sensor setups the observability of the system is examined under the three different considerations of damping. As is often the case in practice, there is also a

consideration of whether the masses of the system are deemed as known or unknown. The observability results of all the scenarios are presented in Table 2.3, where a cell gives the observability property of the system under one of the sensor configurations, one of the assumed damping matrices and either the masses being considered as known or unknown.

As shown in Table 2.3, the system with case i) a full symmetric damping matrix is highly unobservable regardless of whether the masses are assumed to be known or unknown and under any of the two sensor setups. It implies that a method that aims to identify the general matrices of mass, stiffness and damping, as for example the subspace identification method would attempt to do [106], would not succeed under certain sensor setups and applied measured force(s). On the contrary, case ii) the damping matrix makes the system observable for all the scenarios. Interestingly, this is typically the form of damping that one would assume when using for example the methods of nonlinear Kalman filters [25]. The system with case iii) the Rayleigh damping matrix is observable when the masses are known but unobservable when the masses are unknown given the sensor configuration 1. A further observability analysis of this case shows that, when the mass parameters are unknown, both the Rayleigh damping coefficients α and β are identifiable but the stiffness and the masses are

	Sensor configuration 1		Sensor configuration 2	
Damping matrix	Mass parameters	Observability	Mass parameters	Observability
Case i)	known	unobservable	known	unobservable
	unknown	unobservable	unknown	unobservable
Case ii)	known	observable	known	observable
	unknown	observable	unknown	observable
Case iii)	known	observable	known	observable
	unknown	unobservable	unknown	observable

Table 2.3: The observability of a 100 DOFs mass-spring-damper system

not separately identifiable. This is an indication that while one would not able to identify those parameters, the combined $\boldsymbol{M^{-1}K}$ and hence the frequencies of the system might be identified successfully. Unlike the sensor configuration 1, the Rayleigh damped system is observable for both mass scenarios under the sensor configuration 2 where the locations of sensors are distributed evenly along the building.

This example demonstrates that for a given sensor setup the assumptions on the form of damping used may have significant effects on the observability of the system. Interestingly, the most general assumption of case i) would lead to inability of identifying the system for the examined sensor setups. While using a specific damping as in cases ii) and iii) improves the observability of the system, one should also realize that the results obtained are possible only as a consequence of introducing those assumptions. Moreover, using a different sensor configuration or introducing more sensors would also help improve observability. The observability tools proposed in this work allow users to achieve such investigations without concerning about the limitations associated with Algorithm 2.1.

2.5.2 Example 2: 3D finite element model of a truss-beam bridge

Application of Algorithm 2.2 and Algorithm 2.3 to a 3D finite element (FE) model of a truss-beam bridge whose geometry is inspired by Tokyo Gate Bridge is demonstrated in this example. The model under investigation contains more than two thousand states, and its size does not allow for using Algorithm 2.1 in practice due to the associated physical memory requirements. Tokyo Gate Bridge is a recently built (2011) steel bridge located across the Tokyo Bay in Tokyo. The main bridge is a three-span composite structure of trusses and a steel girder. A schematic diagram to

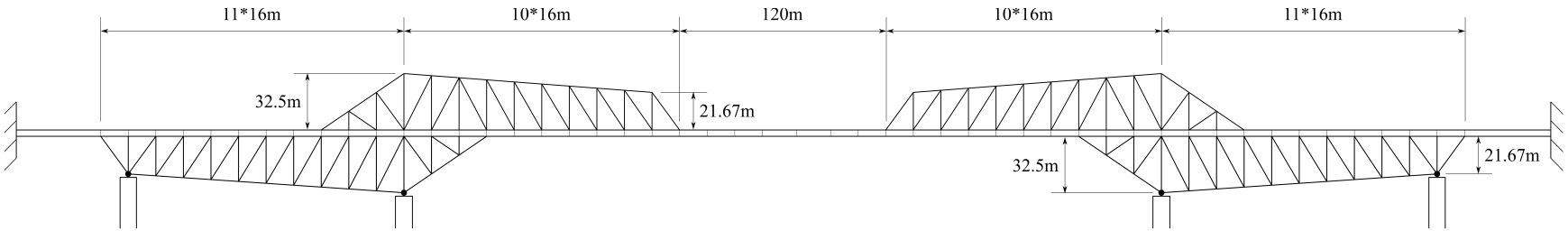

Figure 2.3: The geometry of the bridge model under study

show the geometry of the bridge model under investigation is in Figure 2.3, while the detailed information about the construction of Tokyo Gate Bridge can be found in [112]. This example is chosen to illustrate that the proposed observability algorithms can be implemented in a realistic scenario where a researcher would be investigating the viability of various sensor patterns for estimating the properties of the bridge.

The bridge is modelled using linear frame-type macro-elements. The vector of dynamic states $\boldsymbol{x_t}$ of the model includes the displacements and the corresponding velocities at the nodes of the discretization shown in Figure 2.4. The system matrix $\boldsymbol{A}$ is of the form (2.63), where $\boldsymbol{M}$ and $\boldsymbol{K}$ are now the mass and stiffness matrices of the model that are obtained from assembling the elemental stiffness and mass matrices. The steel deck of the bridge is modelled using beam elements, while the other pin-jointed members are modelled as truss elements. For truss elements, each truss node has 3 DOFs, i.e. the translational motions along x, y and z axes. Each truss element between nodes i and j would contribute to the translational components of the mass matrix at the nodes. In the following the nodal mass from all truss elements lumped at the node i is defined as m_i. The axial stiffness of the truss element between nodes i and j is k_{ij}, and given its rotation relative to horizon, the stiffness matrix of the element can thus be derived in a global coordinate system. The deck is treated as a geometrically uniform Euler-Bernoulli beam. The 792m long beam is discretized into 49 segments including 47 segments of 16m and 2 segments of 20m as illustrated in Figures 2.3 and 2.4. Each beam node has 5 DOFs, as shown in Figure 2.5(a), assuming the axial deformations are neglected. The consistent mass matrix and the

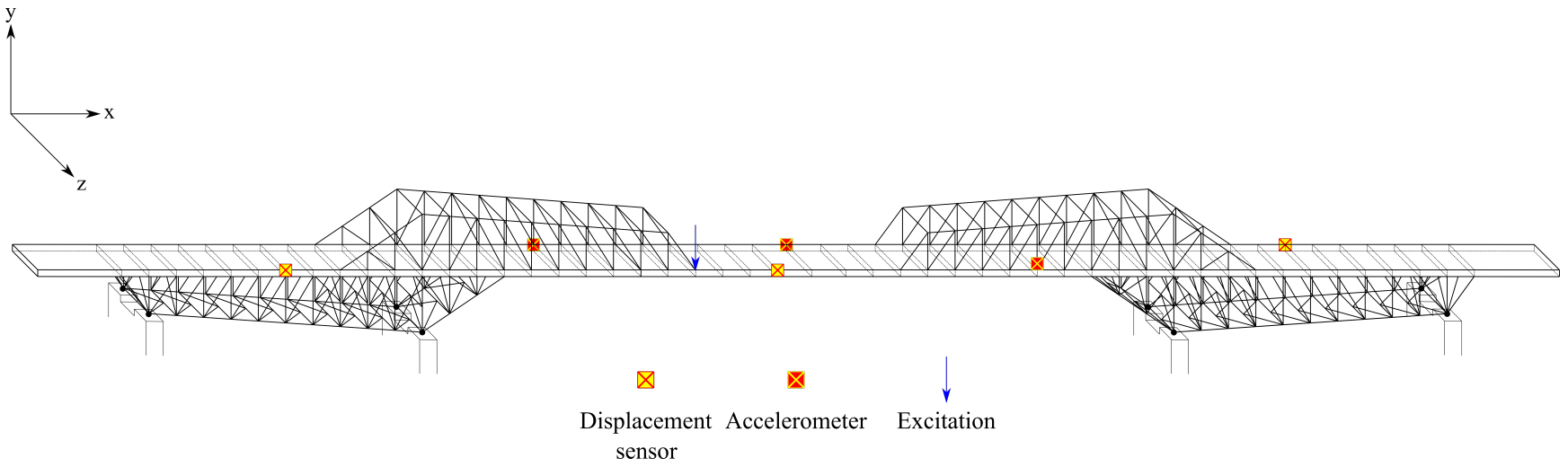

Figure 2.4: The 3D FE model of the bridge with excitation-sensor setups

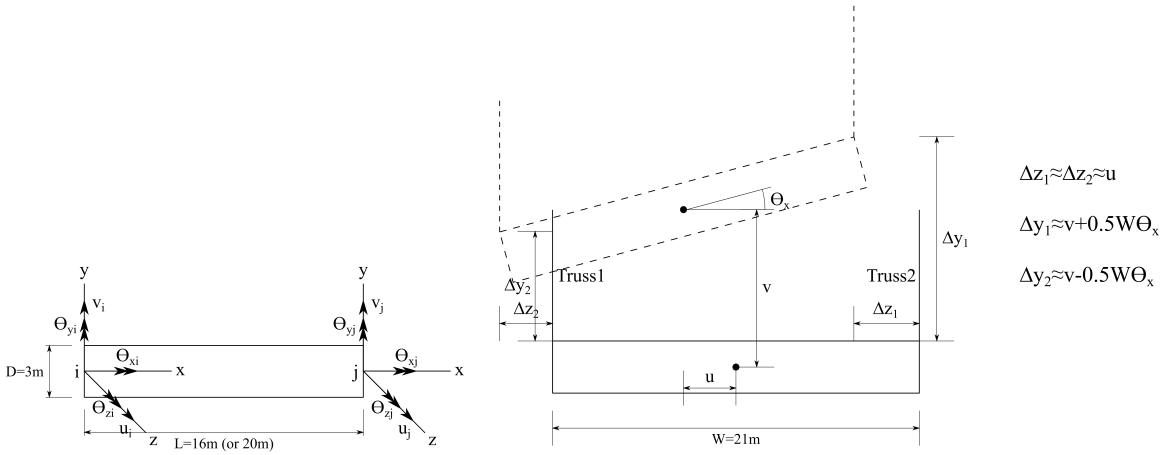

Figure 2.5: (a) Beam segment with 2 nodes having 10 DOFs (b) The translational and rotational motions of a rigid beam cross-section with truss elements connected at two sides

stiffness matrix of a beam segment are given as:

$$\boldsymbol{M}_{beam} = \frac{\rho AL}{420}\begin{bmatrix} 156 & 0 & 0 & 0 & 22L & 54 & 0 & 0 & 0 & -13L \\ 0 & 156 & 0 & -22L & 0 & 0 & 54 & 0 & 13L & 0 \\ 0 & 0 & \frac{140I_x}{A} & 0 & 0 & 0 & 0 & \frac{70I_x}{A} & 0 & 0 \\ 0 & -22L & 0 & 4L^2 & 0 & 0 & -13L & 0 & -3L^2 & 0 \\ 22L & 0 & 0 & 0 & 4L^2 & 13L & 0 & 0 & 0 & -3L^2 \\ 54 & 0 & 0 & 0 & 13L & 156 & 0 & 0 & 0 & -22L \\ 0 & 54 & 0 & -13L & 0 & 0 & 156 & 0 & 22L & 0 \\ 0 & 0 & \frac{70I_x}{A} & 0 & 0 & 0 & 0 & \frac{140I_x}{A} & 0 & 0 \\ 0 & 13L & 0 & -3L^2 & 0 & 0 & 22L & 0 & 4L^2 & 0 \\ -13L & 0 & 0 & 0 & -3L^2 & -22L & 0 & 0 & 0 & 4L^2 \end{bmatrix}$$

$$\boldsymbol{K}_{beam} = \begin{bmatrix} \frac{12EI_z}{L^3} & 0 & 0 & 0 & \frac{6EI_z}{L^2} & \frac{-12EI_z}{L^3} & 0 & 0 & 0 & \frac{6EI_z}{L^2} \\ 0 & \frac{12EI_y}{L^3} & 0 & \frac{-6EI_y}{L^2} & 0 & 0 & \frac{-12EI_y}{L^3} & 0 & \frac{-6EI_y}{L^2} & 0 \\ 0 & 0 & \frac{GJ}{L} & 0 & 0 & 0 & 0 & \frac{-GJ}{L} & 0 & 0 \\ 0 & \frac{-6EI_y}{L^2} & 0 & \frac{4EI_y}{L} & 0 & 0 & \frac{6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & 0 \\ \frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{4EI_z}{L} & \frac{-6EI_z}{L^2} & 0 & 0 & 0 & \frac{2EI_z}{L} \\ \frac{-12EI_z}{L^3} & 0 & 0 & 0 & \frac{-6EI_z}{L^2} & \frac{12EI_z}{L^3} & 0 & 0 & 0 & \frac{-6EI_z}{L^2} \\ 0 & \frac{-12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & 0 & 0 & \frac{12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & 0 \\ 0 & 0 & \frac{-GJ}{L} & 0 & 0 & 0 & 0 & \frac{GJ}{L} & 0 & 0 \\ 0 & \frac{-6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & 0 & 0 & \frac{6EI_y}{L^2} & 0 & \frac{4EI_y}{L} & 0 \\ \frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{2EI_z}{L} & \frac{-6EI_z}{L^2} & 0 & 0 & 0 & \frac{4EI_z}{L} \end{bmatrix} \tag{2.71}$$

where A is the cross section area and L is the length of the beam segment. I_y and I_z are the second moments of area of the cross section about y and z axes respectively, $I_x = I_y + I_z$ is the polar moment of inertia and J is the torsion constant. E and G are the Young's modulus and the shear modulus respectively of the beam segment, and ρ is the density. Assume the Young's modulus and the shear modulus of the beam

segment between nodes i and j are E_{ij} and G_{ij} respectively, and all the segments share the common density ρ. As beam cross section remains plane after deformation, at the nodes of contact between the truss elements and the deck, a rigid element transformation relates the truss nodal DOFs to the beam nodal DOFs as shown in Figure 2.5(b). The rigidity transformations are applied on the elemental mass and stiffness matrices to express the corresponding truss DOFs at those nodes in terms of the beam nodal DOFs. The assembly of the elemental matrices can be found in [30]. The truss elements are connected with moment-free joints to the deck and the piers; the ends of the deck are assumed fixed. The following properties are considered unknown: 508 axial stiffness of the truss elements k_{ij}, 148 lumped masses of the truss nodes m_i, 49 Young's modulus E_{ij} and 49 shear modulus G_{ij} of the beam segments and the density of the deck ρ; the model totally contains 755 unknown parameters to be identified and 1340 dynamic states corresponding to 670 DOFs.

Figure 2.4 gives an overview of the 3D FE model of the bridge with the proposed excitation-sensor setups. The bridge is excited by a load applied vertically at one side of the deck. In practice, such a load may be provided by a hammer or an actuator. Three displacement sensors and three accelerometers are installed on the sides of the deck to record its vertical and transversal displacement and acceleration responses. The observability results obtained from the implementations using Algorithm 2.2 and Algorithm 2.3 are in absolute agreement: there are 36 unidentifiable parameters and 240 unobservable dynamic states. The detailed results are schematically presented in Figure 2.6 with the unobservable dynamic states and the unidentifiable parameters highlighted. Interestingly, the observability of the states related to the left half of the bridge is the same as that of the right half, while it should be noted that the

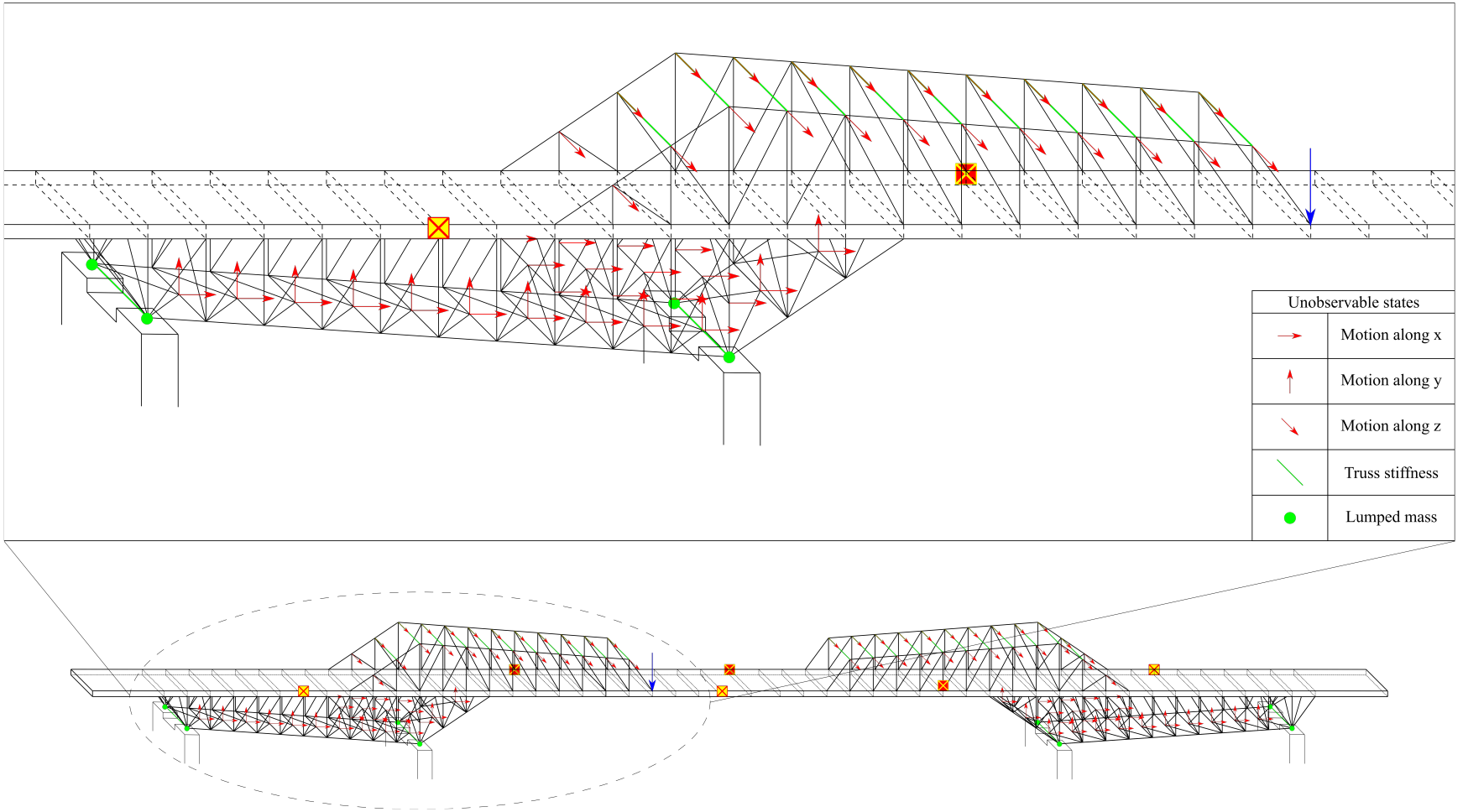

Figure 2.6: The observability of the FE model with the unobservable dynamic states and parameters highlighted by red arrows and green elements respectively

geometry of the bridge is symmetric but the locations of the excitation and sensors are not. As can be seen, the axial stiffness of the lateral truss elements on the top of the bridge are unidentifiable, and their motions (displacement and velocity) toward z direction are unobservable. A further test shows that if extra sensors are installed on the top to measure the lateral vibrations of the truss nodes under e.g. wind load, those dynamic states and parameters become observable. Despite the observability of the truss elements, the parameters and dynamic states related to the beam segments are all observable, and therefore it can be concluded that, under the proposed excitation-measurement scheme, it is feasible to estimate the properties of the deck and simultaneously track its vibrations using system identification methods. In practice, more sensors are often needed to install in order to minimize the influence of measurement noises on the identification quality, that is, to enhance the so called practical observability. The optimal number and locations of additional sensors can be determined by using e.g. information theory based methods, as in [80], which are based on finding the largest norm (determinant or trace) of the Fisher information matrix (FIM). However, the lack of observability of the truss elements might affect the ability to properly identify some of the overall structural modes without introducing further assumptions.

2.6 Conclusions

In this chapter, a robust algorithm is developed to implement the Observability Rank Condition (ORC) for the observability of the dynamic states and the identifiability of the parameters of linear systems. To derive the framework of the algorithm, an explicit expression of the observability matrix of linear systems with unknown pa-

rameters is obtained, and a product rule for the gradient of matrix multiplication is introduced to expand the elements involved in the matrix. This allows for obtaining a recursive evaluation of those elements removing the need to use symbolic differentiations after the initialization of the algorithm. Additionally, it opens up the route of efficient numerical implementations, and the introduction of modular operations allows for obtaining a floating-point implementation that can handle robustly very large linear systems. Applications and superior performance of the algorithm are shown by successfully testing several examples of large engineering systems, including the 100-floor high-rise shear building and the 3D FE model of a large truss-beam bridge. To the best of author's knowledge, none of the existing observability methods have the ability to deal with the problems of such sizes and complexities. The observability results provided by the algorithm are practically useful to give guidance for engineers on sensor placement and model adjustment in system identification campaigns.

Chapter 3

An Efficient Algorithm to Test the Observability of Rational Nonlinear Systems with Unmeasured Inputs

This chapter proposes a computationally efficient algorithm to test the observability of rational nonlinear systems in the presence of unmeasured inputs. The underlying theory behind the algorithm is based on an extension of the extended Observability Rank Condition EORC-DF [67]. Extending the initial work in Sedoglavic [86] to account for unmeasured inputs, the computational framework of the algorithm is developed by making use of an auxiliary variational system of differential equations and the method of Newton's iteration to construct the observability matrix. This framework allows for the use of random numerical realizations of variables and modular operations, resulting in a power series-based, semi-numerical implementation. The algorithm also possesses the functionality to test the observability of rational nonlinear systems with fully measured inputs and without direct feedthrough, which are the subcases of the systems considered in this work. Superior performance and applications of the proposed algorithm are illustrated using several suitably chosen examples of engineering models.

3.1 The extended Observability Rank Condition

In practice, it is often too difficult or expensive, or even impossible, to measure loads, extensions or inputs applied to a structural system. The Observability Rank Condition algorithm (ORC) described in Section 2.1, however, is not able to handle such situations where parts or potentially all of the inputs of a system are unmeasured or unknown. To alleviate this constraint, Martinelli [70] extended the ORC to assess the observability of input-affine nonlinear systems with unknown inputs. Maes et al. [67] provided a further extension, namely the EORC-DF, where the existence of direct feedthrough in output measurements was taken into consideration. The situation of output measurements affected by direct feedthrough is often encountered in practice, as for example the case of acceleration measurement in structural dynamics. This section gives a brief review of the EORC-DF, while the mathematical derivation of the approach can be found in [67] for details.

Consider input-affine nonlinear dynamical systems with unmeasured inputs and direct feedthrough, which are written in the following state-space representation:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x},\boldsymbol{\theta}) + \sum_{i=1}^{m} \boldsymbol{g_{ui}}(\boldsymbol{x},\boldsymbol{\theta})u_i + \sum_{i=1}^{r} \boldsymbol{g_{wi}}(\boldsymbol{x},\boldsymbol{\theta})w_i, \;\; \dot{\boldsymbol{\theta}} = \boldsymbol{0} \\ \boldsymbol{y} &= \boldsymbol{h}(\boldsymbol{x},\boldsymbol{\theta}) + \sum_{i=1}^{m} \boldsymbol{h_{ui}}(\boldsymbol{x},\boldsymbol{\theta})u_i + \sum_{i=1}^{r} \boldsymbol{h_{wi}}(\boldsymbol{x},\boldsymbol{\theta})w_i \end{aligned} \tag{3.1}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is the vector of dynamic states, $\boldsymbol{\theta} \in \mathbb{R}^l$ is the vector of unknown time-invariant parameters, $\boldsymbol{u} = [u_1, ..., u_m]^T$ is the vector of measured and hence known inputs, $\boldsymbol{w} = [w_1, ..., w_r]^T$ is the vector of unmeasured and hence unknown inputs and $\boldsymbol{y} = [y_1, ..., y_p]^T$ is the output measurement vector. $\boldsymbol{f}$ and $\boldsymbol{h}$ are vectors of analytic functions that are infinitely differentiable. Generally speaking, the EORC-DF is used to determine whether it is theoretically feasible to estimate the unknown parameters

$\boldsymbol{\theta}$, and track the dynamic states $\boldsymbol{x}$ and the unmeasured inputs $\boldsymbol{w}$ over time given the input-output $\boldsymbol{u} - \boldsymbol{y}$ measurements.

To derive the EORC-DF approach, the system in equation (3.1) is first augmented by including the dynamic states, the parameters, the unmeasured inputs and their time derivatives up to a chosen order k in a common state vector. It is then proven that the Lie derivatives up to order k of the output functions along the augmented system vector and affine-in-input vectors are equal when evaluated for indistinguishable states. Indistinguishable states are a pair of initial conditions producing identical system outputs for any measured inputs under in general two different vectors of unknown inputs. The procedure of proof is similar but an extension of what was described in Section 2.1.1. This property of Lie derivatives allows for construction of a matrix which consists of the gradients of those Lie derivatives up to order k. If there exists an order k such that the matrix is full rank, then the indistinguishable states that can produce identical outputs do not exist within a closed neighborhood. This implies that the augmented state vector, which includes the dynamic states, parameters and the unmeasured inputs, is (locally weakly) observable based on the input-output measurements.

The following procedure summarizes the steps of the EORC-DF to examine the observability of the system in equation (3.1)

1. Starting point: $k = 0$, $\boldsymbol{x^0} = [\boldsymbol{x}^T, \boldsymbol{\theta}^T, \boldsymbol{w}^T]^T$;
2. Compose $\Delta\boldsymbol{\Omega_0} = [(\boldsymbol{h} + \sum_{i=1}^{r} \boldsymbol{h_{wi}} w_i)^T, \boldsymbol{h_{u1}}^T, ..., \boldsymbol{h_{um}}^T]^T$ and $\boldsymbol{\Omega_0} = \Delta\boldsymbol{\Omega_0}$;
3. Calculate $d^0\boldsymbol{\Omega_0} = \frac{\partial \boldsymbol{\Omega_0}}{\partial \boldsymbol{x^o}}$;
4. $k = k + 1$;
5. Set $\boldsymbol{f^k} = [(\boldsymbol{f} + \sum_{i=1}^{r} \boldsymbol{g_{wi}} w_i)^T, \boldsymbol{0}, \dot{\boldsymbol{w}}^T, ..., \boldsymbol{w}^{(k)T}]^T$, $\boldsymbol{g_{ui}^k} = [\boldsymbol{g_{ui}}^T, \boldsymbol{0}]^T$;

6. Calculate $\Delta\boldsymbol{\Omega_k} = [\boldsymbol{L_{f^k}}(\boldsymbol{\Delta\Omega_{k-1}})^T, \boldsymbol{L_{g^k_{u1}}}(\boldsymbol{\Delta\Omega_{k-1}})^T, ..., \boldsymbol{L_{g^k_{um}}}(\boldsymbol{\Delta\Omega_{k-1}})^T]^T$;

7. Compose $\boldsymbol{x^k} = [\boldsymbol{x^{k-1}}^T, \boldsymbol{w}^{(k)T}]^T$;

8. Calculate $d^k\Delta\boldsymbol{\Omega_k} = \frac{\partial\Delta\boldsymbol{\Omega_k}}{\partial\boldsymbol{x^k}}$;

9. Calculate $d^k\boldsymbol{\Omega_k} = [d^{k-1}\boldsymbol{\Omega_{k-1}}, \boldsymbol{0}] \cup d^k\Delta\boldsymbol{\Omega_k}$;

10. Optional step: eliminate dependent rows from $d^k\boldsymbol{\Omega_k}$;

11. If $rank(d^k\boldsymbol{\Omega_k}) = n + l + (k+1)r$, end. $\boldsymbol{x}$, $\boldsymbol{\theta}$ and $\boldsymbol{w}$ are observable.

12. Investigate the partial observability for the j^{th} component of $\boldsymbol{x^k}$:

    a. Starting point: $j = 1$;

    b. If the j^{th} component is $(k-1)$-row observable, it is also k-row observable. Go to step f;

    c. Compose $d^k\boldsymbol{\Omega^j_k}$ by removing the j^{th} column from $d^k\boldsymbol{\Omega_k}$;

    d. The j^{th} component is observable if and only if $rank(d^k\boldsymbol{\Omega^j_k}) < rank(d^k\boldsymbol{\Omega_k})$;

    e. If $j = n + l + (k+1)r$, end.

    f. $j = j + 1$ and go to step b;

13. If $\boldsymbol{x}$, $\boldsymbol{\theta}$ and $\boldsymbol{w}$ are observable, end.

14. Go to step 4;

3.2 The extended Observability Rank Condition for rational nonlinear systems with unmeasured inputs and direct feedthrough

This section provides an extension of the EORC-DF to examine the observability of rational nonlinear systems with unmeasured inputs and direct feedthrough. Without loss of generality, the considered systems herein can be written in the following state-

space representation:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{u}, \boldsymbol{w}), \;\; \dot{\boldsymbol{\theta}} = \boldsymbol{0} \\ \boldsymbol{y} &= \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{u}, \boldsymbol{w}) \end{aligned} \tag{3.2}$$

where $\boldsymbol{x} = [x_1, ..., x_n]^T$ is the vector of dynamic states, $\boldsymbol{\theta} = [\theta_1, ..., \theta_l]^T$ is the vector of unknown time-invariant parameters, $\boldsymbol{u} = [u_1, ..., u_m]^T$ is the vector of measured and hence known inputs, $\boldsymbol{w} = [w_1, ..., w_r]^T$ is the vector of unmeasured and hence unknown inputs and $\boldsymbol{y} = [y_1, ..., y_p]^T$ is the output measurement vector. The output measurements can be affected by direct feedthrough, that is, they are functions of both the measured and unmeasured inputs. $\boldsymbol{f}$ and $\boldsymbol{h}$ are vectors of rational nonlinear functions, where a rational function is any function defined by a fraction such that both its numerator and denominator are polynomial functions. In the system described by equation (3.2) it is of interest to determine whether it is theoretically feasible to estimate the unknown parameters $\boldsymbol{\theta}$, and track the dynamic states $\boldsymbol{x}$ and the unmeasured inputs $\boldsymbol{w}$ over time given the input-output $\boldsymbol{u} - \boldsymbol{y}$ measurements.

The basic idea for achieving the derivation of the observability criterion is to augment the considered system by treating the unmeasured inputs and their time derivatives as additional states [67], which then allows for discussion of the Lie derivative property of the augmented system leading to the use of the rank condition from [44]. Following what was done in [67], the system in equation (3.2) can be augmented by including the dynamic states $\boldsymbol{x}$, the parameters $\boldsymbol{\theta}$, the unmeasured inputs $\boldsymbol{w}$ and their time derivatives up to order k in a common state vector, $\boldsymbol{x^k}$:

$$\boldsymbol{x^k} = [\boldsymbol{x}^T, \boldsymbol{\theta}^T, \boldsymbol{w}^T, \dot{\boldsymbol{w}}^T, ..., \boldsymbol{w}^{(k)T}]^T \tag{3.3}$$

such that the state-space and measurement equations of the system with respect to

$\boldsymbol{x^k}$ become:

$$\dot{\boldsymbol{x}}^{\boldsymbol{k}} = \begin{bmatrix} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{u}, \boldsymbol{w}) \\ \mathbf{0}_{l\times 1} \\ \dot{\boldsymbol{w}} \\ \vdots \\ \boldsymbol{w}^{(k)} \\ \mathbf{0}_{r\times 1} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{n\times 1} \\ \mathbf{0}_{l\times 1} \\ \mathbf{0}_{r\times 1} \\ \vdots \\ \mathbf{0}_{r\times 1} \\ \boldsymbol{w}^{(k+1)} \end{bmatrix} = \boldsymbol{F}^{\boldsymbol{k}}(\boldsymbol{x}^{\boldsymbol{k}}, \boldsymbol{u}, \boldsymbol{w}^{(k+1)}) \tag{3.4}$$
$$\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{u}, \boldsymbol{w})$$

where $\boldsymbol{w}^{(k)} = \frac{d^k \boldsymbol{w}}{dt^k}$. It should be noted that the augmented system in equation (3.4) still contains r unmeasured inputs which now coincide with the $(k+1)^{th}$ order time derivatives of the original unmeasured inputs, i.e. $\boldsymbol{w}^{(k+1)}$.

To proceed with the derivation, it is necessary to introduce the Lie derivatives, also referred to as the extended Lie derivatives as in [56], of the output function of the augmented system using the following formulation:

$$\boldsymbol{L_f^i h} = \frac{\partial \boldsymbol{L_f^{i-1} h}}{\partial \boldsymbol{x}} \boldsymbol{f} + \sum_{j=1}^{i} \frac{\partial \boldsymbol{L_f^{i-1} h}}{\partial \boldsymbol{w}^{(j-1)}} \boldsymbol{w}^{(j)} + \sum_{j=1}^{i} \frac{\partial \boldsymbol{L_f^{i-1} h}}{\partial \boldsymbol{u}^{(j-1)}} \boldsymbol{u}^{(j)} \tag{3.5}$$

where $\boldsymbol{L_f^i h}$ is defined as the i^{th} order (extended) Lie derivative of $\boldsymbol{h}$ associated with the vector field $\boldsymbol{f}$, which can be calculated recursively from the previous order using equation (3.5) given the zero-order $\boldsymbol{L_f^0 h} = \boldsymbol{h}$. Throughout this chapter, the operation $\frac{\partial \boldsymbol{V_1}}{\partial \boldsymbol{V_2}}$ for a column vector $\boldsymbol{V_1} = [v_{1,1}, v_{1,2}..., v_{1,n_{V_1}}]^T$ with respect to a column or row vector $\boldsymbol{V_2} = [v_{2,1}, v_{2,2}..., v_{2,n_{V_2}}]$ refers to the Jacobian:

$$\frac{\partial \boldsymbol{V_1}}{\partial \boldsymbol{V_2}} = \begin{bmatrix} \frac{\partial v_{1,1}}{\partial v_{2,1}} & \frac{\partial v_{1,1}}{\partial v_{2,2}} & \cdots & \frac{\partial v_{1,1}}{\partial v_{2,n_{V_2}}} \\ \frac{\partial v_{1,2}}{\partial v_{2,1}} & \frac{\partial v_{1,2}}{\partial v_{2,2}} & \cdots & \frac{\partial v_{1,2}}{\partial v_{2,n_{V_2}}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial v_{1,n_{V_1}}}{\partial v_{2,1}} & \frac{\partial v_{1,n_{V_1}}}{\partial v_{2,2}} & \cdots & \frac{\partial v_{1,n_{V_1}}}{\partial v_{2,n_{V_2}}} \end{bmatrix} \tag{3.6}$$

Unlike the Lie derivative operation used in [67] for piecewise-constant assumption of the measured inputs $\boldsymbol{u}$, the Lie derivatives herein are introduced for smooth $\boldsymbol{u}$ and thus involve the time derivatives of $\boldsymbol{u}$ within their expressions. It should be noted that

the relationship and equivalence between using the piecewise constant and smooth assumptions of $\boldsymbol{u}$ have been discussed in [93] and reviewed in [2], indicating that the assumptions would lead to different forms of computation but eventually the same observability result. A proof for the equivalence of smooth inputs and piecewise-constant inputs, based on the uniform convergence property and the fact that the outputs depend continuously on the inputs, was provided in [96].

A straightforward calculation based on the chain rule shows that the i^{th} order Lie derivative is equivalent to:

$$\boldsymbol{L_f^i h} = \frac{d\boldsymbol{L_f^{i-1} h}}{dt} = \frac{d^i \boldsymbol{h}}{dt^i} \tag{3.7}$$

Now let $\boldsymbol{x_0^k} = [\boldsymbol{x_0}^T, \boldsymbol{\theta}^T, \boldsymbol{w_0}^T, \dot{\boldsymbol{w}}_{\boldsymbol{0}}^T, ..., \boldsymbol{w_0}^{(k)T}]^T$ denote the initial condition of the augmented state vector $\boldsymbol{x^k}$ at $t = 0$, where $\boldsymbol{x_0}$ and $\boldsymbol{w_0}^{(k)}$ are respectively the initial conditions of $\boldsymbol{x}$ and $\boldsymbol{w}^{(k)}$. The flow $\boldsymbol{\Phi}(t, \boldsymbol{x_0^k}, \boldsymbol{u}, \boldsymbol{w}^{(k+1)})$ of the vector field $\boldsymbol{F^k}$ is used to describe the trajectory of $\boldsymbol{x^k}$ over time, which satisfies:

$$\frac{d\boldsymbol{\Phi}}{dt} = \boldsymbol{F^k}(\boldsymbol{\Phi}, \boldsymbol{u}, \boldsymbol{w}^{(k+1)}), \;\; \boldsymbol{\Phi}(0, \boldsymbol{x_0^k}, \boldsymbol{u}, \boldsymbol{w}^{(k+1)}) = \boldsymbol{x_0^k} \tag{3.8}$$

$\boldsymbol{\Phi_x}(t, \boldsymbol{x_0^k}, \boldsymbol{u}, \boldsymbol{w}^{(k+1)})$ and $\boldsymbol{\Phi_w}(t, \boldsymbol{x_0^k}, \boldsymbol{u}, \boldsymbol{w}^{(k+1)})$ are the components of $\boldsymbol{\Phi}$ defined respectively as the flows of $\boldsymbol{x}$ and $\boldsymbol{w}$ with $\boldsymbol{\Phi_x}(0, \boldsymbol{x_0^k}, \boldsymbol{u}, \boldsymbol{w}^{(k+1)}) = \boldsymbol{x_0}$ and $\boldsymbol{\Phi_w}(0, \boldsymbol{x_0^k}, \boldsymbol{u}, \boldsymbol{w}^{(k+1)}) = \boldsymbol{w_0}$. Then, $\boldsymbol{x}$ and $\boldsymbol{w}$ reached at time t starting from the initial condition may be expressed as:

$$\begin{aligned} \boldsymbol{x}(t; \boldsymbol{x_0^k}) &= \boldsymbol{\Phi_x}(t, \boldsymbol{x_0^k}, \boldsymbol{u}, \boldsymbol{w}^{(k+1)}) \\ \boldsymbol{w}(t; \boldsymbol{x_0^k}) &= \boldsymbol{\Phi_w}(t, \boldsymbol{x_0^k}, \boldsymbol{u}, \boldsymbol{w}^{(k+1)}) \end{aligned} \tag{3.9}$$

Considering the time interval $t \in [0, t_1 + t_2 + ... + t_i]$ which consists of i small time intervals $0 \leq t_i \leq \epsilon$, the system outputs at $t = t_1 + t_2 + ... + t_i$ can thus be expressed as:

$$\boldsymbol{y}(t_1 + ... + t_i) = \boldsymbol{h}(\boldsymbol{x}(t_1 + ... + t_i; \boldsymbol{x_0^k}), \boldsymbol{\theta}, \boldsymbol{u}(t_1 + ... + t_i), \boldsymbol{w}(t_1 + ... + t_i; \boldsymbol{x_0^k})) \tag{3.10}$$

Indistinguishable states in the presence of unknown inputs are defined in [70] as a pair of initial conditions, $\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0,a}}$ and $\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0,b}}$, producing identical system outputs for any measured inputs $\boldsymbol{u}$, under in general two different vectors of unknown inputs. For any i, the indistinguishable states must satisfy:

$$\begin{aligned}&\boldsymbol{h}\big(\boldsymbol{x}(t_1+...+t_i;\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0,a}}),\boldsymbol{\theta_a},\boldsymbol{u}(t_1+...+t_i),\boldsymbol{w}(t_1+...+t_i;\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0,a}})\big)\\ =&\boldsymbol{h}(\boldsymbol{x}\big(t_1+...+t_i;\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0,b}}),\boldsymbol{\theta_b},\boldsymbol{u}(t_1+...+t_i),\boldsymbol{w}(t_1+...+t_i;\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0,b}})\big)\end{aligned} \tag{3.11}$$

for all possible $t_1, ..., t_i$. From this, the following equality can be shown to hold:

$$\begin{aligned}&\frac{d^i\boldsymbol{h}\big(\boldsymbol{x}(t_1+...+t_i;\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0,a}}),\boldsymbol{\theta_a},\boldsymbol{u}(t_1+...+t_i),\boldsymbol{w}(t_1+...+t_i;\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0,a}})\big)}{dt_1dt_2...dt_i}\Big|_{t_1,...,t_i\to 0}\\ =&\frac{d^i\boldsymbol{h}\big(\boldsymbol{x}(t_1+...+t_i;\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0,b}}),\boldsymbol{\theta_b},\boldsymbol{u}(t_1+...+t_i),\boldsymbol{w}(t_1+...+t_i;\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0,b}})\big)}{dt_1dt_2...dt_i}\Big|_{t_1,...,t_i\to 0}\end{aligned} \tag{3.12}$$

Through successive applications of the chain rule, the following equation is obtained for $i = 0, ..., k$:

$$\begin{aligned}&\frac{d^i\boldsymbol{h}\big(\boldsymbol{x}(t_1+...+t_i;\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0}}),\boldsymbol{\theta},\boldsymbol{u}(t_1+...+t_i),\boldsymbol{w}(t_1+...+t_i;\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0}})\big)}{dt_1dt_2...dt_i}\Big|_{t_1,...,t_i\to 0}\\ =&\boldsymbol{L}^{\boldsymbol{i}}_{\boldsymbol{f}}\boldsymbol{h}\Big|_{\boldsymbol{x}^{\boldsymbol{k}}=\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0}},\boldsymbol{u},...,\boldsymbol{u}^{(k)}=\boldsymbol{u_0},...,\boldsymbol{u_0}^{(k)}}\end{aligned} \tag{3.13}$$

where $\boldsymbol{u_0}^{(k)}$ is the initial condition of $\boldsymbol{u}^{(k)}$ at $t = 0$. It should be noted that the Lie derivatives up to order k are independent of the unmeasured inputs of the augmented system i.e. $\boldsymbol{w}^{(k+1)}$ and their higher order time derivatives, as can be easily deduced from equation (3.5). Equation (3.12) therefore becomes:

$$\begin{aligned}&\boldsymbol{L}^{\boldsymbol{i}}_{\boldsymbol{f}}\boldsymbol{h}\Big|_{\boldsymbol{x}^{\boldsymbol{k}}=\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0,a}},\boldsymbol{u},...,\boldsymbol{u}^{(k)}=\boldsymbol{u_0},...,\boldsymbol{u_0}^{(k)}}\\ =&\boldsymbol{L}^{\boldsymbol{i}}_{\boldsymbol{f}}\boldsymbol{h}\Big|_{\boldsymbol{x}^{\boldsymbol{k}}=\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0,b}},\boldsymbol{u},...,\boldsymbol{u}^{(k)}=\boldsymbol{u_0},...,\boldsymbol{u_0}^{(k)}}\end{aligned} \tag{3.14}$$

for $i = 0, ..., k$. For the sake of brevity, equation (3.14) may be written as $\boldsymbol{L}^{\boldsymbol{i}}_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0,a}}) = \boldsymbol{L}^{\boldsymbol{i}}_{\boldsymbol{f}}\boldsymbol{h}(\boldsymbol{x}^{\boldsymbol{k}}_{\boldsymbol{0,b}})$, which implies that the Lie derivatives of the augmented system up to order k are equal when evaluated for indistinguishable states.

Similar as in [67], the afore-shown Lie derivative property allows for applying the rank condition from [44] (see Theorem 3.1 and Lemma 3.2) to establish the observability criterion for the augmented system in equation (3.4). Arrange the Lie derivatives for $i = 0, ..., k$ in a column vector:

$$\boldsymbol{\Omega^k} = \begin{bmatrix} \boldsymbol{L_f^0 h}(\boldsymbol{x_0^k}) \\ \boldsymbol{L_f h}(\boldsymbol{x_0^k}) \\ \vdots \\ \boldsymbol{L_f^k h}(\boldsymbol{x_0^k}) \end{bmatrix} \tag{3.15}$$

The Jacobian of $\boldsymbol{\Omega^k}$ with respect to $\boldsymbol{x_0^k}$ results in the so called k-row observability matrix $d\boldsymbol{\Omega^k}$:

$$d\boldsymbol{\Omega^k} = \frac{\partial \boldsymbol{\Omega^k}}{\partial \boldsymbol{x_0^k}} = [d\boldsymbol{\Omega_x^k}, d\boldsymbol{\Omega_\theta^k}, d\boldsymbol{\Omega_{w_1}^k}, ..., d\boldsymbol{\Omega_{w_r}^k}] \tag{3.16}$$

where

$$\begin{aligned} d\boldsymbol{\Omega_x^k} &= \frac{\partial \boldsymbol{\Omega^k}}{\partial \boldsymbol{x_0}} = \begin{bmatrix} \frac{\partial \boldsymbol{L_f^0 h}}{\partial x_{1,0}} & \cdots & \frac{\partial \boldsymbol{L_f^0 h}}{\partial x_{n,0}} \\ \vdots & \ddots & \vdots \\ \frac{\partial \boldsymbol{L_f^k h}}{\partial x_{1,0}} & \cdots & \frac{\partial \boldsymbol{L_f^k h}}{\partial x_{n,0}} \end{bmatrix}, \quad d\boldsymbol{\Omega_\theta^k} = \frac{\partial \boldsymbol{\Omega^k}}{\partial \boldsymbol{\theta}} = \begin{bmatrix} \frac{\partial \boldsymbol{L_f^0 h}}{\partial \theta_1} & \cdots & \frac{\partial \boldsymbol{L_f^0 h}}{\partial \theta_l} \\ \vdots & \ddots & \vdots \\ \frac{\partial \boldsymbol{L_f^k h}}{\partial \theta_1} & \cdots & \frac{\partial \boldsymbol{L_f^k h}}{\partial \theta_l} \end{bmatrix} \\ d\boldsymbol{\Omega_{w_j}^k} &= \begin{bmatrix} \frac{\partial \boldsymbol{L_f^0 h}}{\partial w_{j,0}} & 0 & \cdots & 0 \\ \frac{\partial \boldsymbol{L_f h}}{\partial w_{j,0}} & \frac{\partial \boldsymbol{L_f h}}{\partial \dot{w}_{j,0}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \boldsymbol{L_f^k h}}{\partial w_{j,0}} & \frac{\partial \boldsymbol{L_f^k h}}{\partial \dot{w}_{j,0}} & \cdots & \frac{\partial \boldsymbol{L_f^k h}}{\partial w_{j,0}^{(k)}} \end{bmatrix} \quad (j = 1, ..., r) \end{aligned} \tag{3.17}$$

If $d\boldsymbol{\Omega^k}$ is a full-rank matrix, i.e. $rank(d\boldsymbol{\Omega^k}) = n + l + (k+1)r$, the Inverse Function Theorem [94] ensures that the only indistinguishable state of $\boldsymbol{x_0^k}$, i.e. the one resulting in the same Lie derivatives as $\boldsymbol{x_0^k}$, within its closed neighborhood is $\boldsymbol{x_0^k}$ itself. This consequently means that it is possible to distinguish and observe $\boldsymbol{x_0^k}$ from its neighbors using the given input-output measurements. Following the above fact, the observability criterion can be concluded: the augmented system in equation (3.4), or equivalently the original considered system in (3.2), is said to satisfy the extended Observability Rank Condition EORC-DF such that the augmented state

vector $\boldsymbol{x^k}$, or equivalently $\boldsymbol{x}$, $\boldsymbol{\theta}$ and $\boldsymbol{w}$ as the components of $\boldsymbol{x^k}$, are (locally weakly) observable, if and only if there exists a k such that $d\boldsymbol{\Omega^k}$ is full-ranked for all possible $\boldsymbol{x_0^k} \in \mathbb{R}^{n+l+(k+1)r}$ (normally except for those singular realizations of $\boldsymbol{x_0^k}$ where rank of the matrix would be reduced). If the considered system in (3.2) satisfies the extended Observability Rank Condition EORC-DF, it is then theoretically feasible to distinguish and observe the initial conditions of $\boldsymbol{x}$, $\boldsymbol{\theta}$ and $\boldsymbol{w}$ and their trajectories over time from their neighbors using the given input-output measurements. From a practical standpoint and for the purposes of this work, further distinctions between the concepts of locally weakly observable, locally observable, weakly observable and globally observable, as in [44], are not considered herein.

If the k-row observability matrix for a chosen $k = k_0$ is not a full-rank matrix, i.e. $rank(d\boldsymbol{\Omega^k}) < n + l + (k + 1)r$, it is then important to detect the observability property of each individual component of $\boldsymbol{x^k}$. This can be achieved by removing the i^{th} $(i = 1, ..., n+l+(k+1)r)$ column of $d\boldsymbol{\Omega^k}$: if the rank of the occurring matrix is smaller than the rank of $d\boldsymbol{\Omega^k}$, then the i^{th} state (dynamic state/parameter/unmeasured input) of $\boldsymbol{x^k}$ is observable; otherwise, the state is so-called k-row unobservable as defined in [67]. It should be noted that a k-row unobservable state might be detected observable or remain k-row unobservable for a larger $k > k_0$, while an (k-row) observable state can be concluded immediately. For this reason, the suggested implementation of the EORC-DF involves iterative increment of k starting from k_0 ($k_0 = 0$ in [67]) with repeated rank evaluation and observability detection of the corresponding observability matrix at each increment of k. In theory, such increment is continued until the detected observability properties of all the components of $\boldsymbol{x^k}$ remain invariant, i.e. converge, for any larger k and thus conclude the observability results.

The following algorithm summarizes the procedure of observability testing for the rational nonlinear system in equation (3.2).

Algorithm 3.1

Input to the algorithm: the state-space and measurement equations of the system

Output from the algorithm: the observability of the system

Initialization: Set $k = 0$, $\boldsymbol{x_0^k} = [\boldsymbol{x_0}^T, \boldsymbol{\theta}^T, \boldsymbol{w_0}^T]^T$, $\boldsymbol{L_f^k h}(\boldsymbol{x_0^k}) = \boldsymbol{h}(\boldsymbol{x_0}, \boldsymbol{\theta}, \boldsymbol{u_0}, \boldsymbol{w_0})$, $d\boldsymbol{\Omega^k} = \frac{\partial \boldsymbol{L_f^k h}(\boldsymbol{x_0^k})}{\partial \boldsymbol{x_0^k}}$

1. Set $k = k + 1$;

2. Set $\boldsymbol{x_0^k} = [\boldsymbol{x_0^{k-1}}^T, \boldsymbol{w_0}^{(k)T}]^T$;

3. Compute $\boldsymbol{L_f^k h} = \frac{\partial \boldsymbol{L_f^{k-1} h}}{\partial \boldsymbol{x}} \boldsymbol{f} + \sum_{j=1}^{k} \frac{\partial \boldsymbol{L_f^{k-1} h}}{\partial \boldsymbol{w}^{(j-1)}} \boldsymbol{w}^{(j)} + \sum_{j=1}^{k} \frac{\partial \boldsymbol{L_f^{k-1} h}}{\partial \boldsymbol{u}^{(j-1)}} \boldsymbol{u}^{(j)}$;

4. Compute and arrange $d\boldsymbol{\Omega^k} = \begin{bmatrix} d\boldsymbol{\Omega^{k-1}} & \boldsymbol{0} \\ \frac{\partial \boldsymbol{L_f^k h}(\boldsymbol{x_0^k})}{\partial \boldsymbol{x_0^k}} \end{bmatrix}$;

5. Compute the rank of $d\boldsymbol{\Omega^k}$, and if $rank(d\boldsymbol{\Omega^k}) < n + l + (k+1)r$, detect the observability of $\boldsymbol{x}$, $\boldsymbol{\theta}$ and $\boldsymbol{w}$;

6. End if $rank(d\boldsymbol{\Omega^k}) = n + l + (k+1)r$, or $\boldsymbol{x}$, $\boldsymbol{\theta}$ and $\boldsymbol{w}$ are observable, or the observability of $\boldsymbol{x}$, $\boldsymbol{\theta}$ and $\boldsymbol{w}$ has been convergent;

7. Go to step **1**;

The above algorithm should be implemented symbolically, i.e. symbols are used as generic representations of the involved variables in the algorithm. To compute $d\boldsymbol{\Omega^k}$, the involved variables can include at most $\boldsymbol{x_0}$, $\boldsymbol{\theta}$, $\boldsymbol{w_0}$, ..., $\boldsymbol{w_0}^{(k)}$ and $\boldsymbol{u_0}$, ..., $\boldsymbol{u_0}^{(k)}$. From an application point of view, however, such a symbolic implementation is computationally cumbersome and suffers from the issues related to high physical memory requirements and low processing speed, especially when used for large systems. Even if a tested system is of modest size, the size of the augmented form of the system for a large k might still result in an intractable computation problem on a standard com-

puter. Therefore in order to maximise the applicability of the observability testing approach to real-world engineering systems that are often large and complex, a more computationally efficient framework for implementing the approach, more specifically for calculating the observability matrix, is developed and presented in the following section.

3.3 Efficient algorithm

The computation and rank evaluation of the k-row observability matrix $d\boldsymbol{\Omega}^{\boldsymbol{k}}$ could be implemented numerically by using random values as the generic representations of the involved variables, which would significantly save computational resources. This can be done by making:

$$\boldsymbol{x_0} = \tilde{\boldsymbol{x}}_{\boldsymbol{0}},\ \ \boldsymbol{\theta} = \tilde{\boldsymbol{\theta}},\ \ \boldsymbol{w_0}, ..., \boldsymbol{w_0}^{(k)} = \tilde{\boldsymbol{w}}_{\boldsymbol{0}}, ..., \tilde{\boldsymbol{w}}_{\boldsymbol{0}}{}^{(k)},\ \ \boldsymbol{u_0}, ..., \boldsymbol{u_0}^{(k)} = \tilde{\boldsymbol{u}}_{\boldsymbol{0}}, ..., \tilde{\boldsymbol{u}}_{\boldsymbol{0}}{}^{(k)} \quad (3.18)$$

where the variables with $\tilde{}$ stand for a set of positive integers. These integers are randomly chosen in order to avoid the aforementioned singular realizations of $\boldsymbol{x_0^k}$. This section presents a power series-based framework which allows for using such random numerical realizations to calculate the elements of $d\boldsymbol{\Omega}^{\boldsymbol{k}}$ without the need to perform the repeated gradient and Jacobian operations.

Making use of equation (3.7) that $\boldsymbol{L_f^i h} = \frac{d^i \boldsymbol{h}}{dt^i}$, consider the following power series expansion of the output function truncated at order k:

$$\boldsymbol{h} = \boldsymbol{L_f^0 h}(\boldsymbol{x_0^k}) + \boldsymbol{L_f h}(\boldsymbol{x_0^k})t + ... + \boldsymbol{L_f^k h}(\boldsymbol{x_0^k})\frac{t^k}{k!} + O(t^{k+1}) \quad (3.19)$$

Let $\boldsymbol{w_{j,0}^k}$ denote the initial condition of the vector $\boldsymbol{w_j^k} = [w_j, ..., w_j^{(k)}]$, and taking derivatives of both sides of equation (3.19) with respect to $\boldsymbol{x_0}$, $\boldsymbol{\theta}$ and $\boldsymbol{w_{j,0}^k}$ respectively

yields:

$$\begin{aligned}
\frac{d\boldsymbol{h}}{d\boldsymbol{x_0}} &= \frac{\partial \boldsymbol{L_f^0 h}(\boldsymbol{x_0^k})}{\partial \boldsymbol{x_0}} + \frac{\partial \boldsymbol{L_f h}(\boldsymbol{x_0^k})}{\partial \boldsymbol{x_0}} t + ... + \frac{\partial \boldsymbol{L_f^k h}(\boldsymbol{x_0^k})}{\partial \boldsymbol{x_0}} \frac{t^k}{k!} + O(t^{k+1}) \\
\frac{d\boldsymbol{h}}{d\boldsymbol{\theta}} &= \frac{\partial \boldsymbol{L_f^0 h}(\boldsymbol{x_0^k})}{\partial \boldsymbol{\theta}} + \frac{\partial \boldsymbol{L_f h}(\boldsymbol{x_0^k})}{\partial \boldsymbol{\theta}} t + ... + \frac{\partial \boldsymbol{L_f^k h}(\boldsymbol{x_0^k})}{\partial \boldsymbol{\theta}} \frac{t^k}{k!} + O(t^{k+1}) \\
\frac{d\boldsymbol{h}}{d\boldsymbol{w_{j,0}^k}} &= \frac{\partial \boldsymbol{L_f^0 h}(\boldsymbol{x_0^k})}{\partial \boldsymbol{w_{j,0}^k}} + \frac{\partial \boldsymbol{L_f h}(\boldsymbol{x_0^k})}{\partial \boldsymbol{w_{j,0}^k}} t + ... + \frac{\partial \boldsymbol{L_f^k h}(\boldsymbol{x_0^k})}{\partial \boldsymbol{w_{j,0}^k}} \frac{t^k}{k!} + O(t^{k+1})
\end{aligned} \tag{3.20}$$

It should be noted that the three power series in equation (3.20) can also be viewed as the Taylor series expansions of $\frac{d\boldsymbol{h}}{d\boldsymbol{x_0}}$, $\frac{d\boldsymbol{h}}{d\boldsymbol{\theta}}$ and $\frac{d\boldsymbol{h}}{d\boldsymbol{w_{j,0}^k}}$ at $t = 0$ up to order k. Comparing equation (3.20) to equations (3.16) and (3.17), it is immediately observed that the coefficients of the power series correspond to the elements of $d\boldsymbol{\Omega^k}$. In particular, the coefficients of the power series of $\frac{d\boldsymbol{h}}{d\boldsymbol{x_0}}$ correspond to the elements of $d\boldsymbol{\Omega_x^k}$, those of $\frac{d\boldsymbol{h}}{d\boldsymbol{\theta}}$ correspond to the elements of $d\boldsymbol{\Omega_\theta^k}$ and those of $\frac{d\boldsymbol{h}}{d\boldsymbol{w_{j,0}^k}}$ correspond to the elements of $d\boldsymbol{\Omega_{w_j}^k}$. This leads directly to the idea that the computations can be turned toward calculating the power series in equation (3.20), and consequently their coefficients can be extracted to build the observability matrix. For the sake of brevity, in the rest of the chapter the power series expansion of any variable z, with respect to t and truncated at order k, will be denoted by the notation $z[t, k]$, i.e.

$$z[t, k] = z_0 + z_1 t + ... + z_k t^k + O(t^{k+1}) \tag{3.21}$$

where $z_0, ..., z_k$ are constant coefficients, and $z_0, z_1 t, ..., z_k t^k$ are the terms of the power series.

Applying the chain rule to the power series expansions of $\frac{d\boldsymbol{h}}{d\boldsymbol{x_0}}$, $\frac{d\boldsymbol{h}}{d\boldsymbol{\theta}}$ and $\frac{d\boldsymbol{h}}{d\boldsymbol{w_{j,0}^k}}$ in equation (3.20), which are now denoted by $\frac{d\boldsymbol{h}}{d\boldsymbol{x_0}}[t, k]$, $\frac{d\boldsymbol{h}}{d\boldsymbol{\theta}}[t, k]$ and $\frac{d\boldsymbol{h}}{d\boldsymbol{w_{j,0}^k}}[t, k]$, the following

equation is obtained:

$$\begin{aligned}
\frac{d\boldsymbol{h}}{d\boldsymbol{x_0}}[t,k] &= (\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}})[t,k] \\
&= \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}[t,k]\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}[t,k] \ \ (mod\ t^{k+1}) \\
\frac{d\boldsymbol{h}}{d\boldsymbol{\theta}}[t,k] &= (\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}} + \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}})[t,k] \\
&= \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}[t,k]\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}[t,k] + \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}[t,k] \ \ (mod\ t^{k+1}) \\
\frac{d\boldsymbol{h}}{d\boldsymbol{w^k_{j,0}}}[t,k] &= (\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w^k_{j,0}}} + \frac{\partial \boldsymbol{h}}{\partial w_j}\frac{\partial w_j}{\partial \boldsymbol{w^k_{j,0}}})[t,k] \\
&= \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}[t,k]\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w^k_{j,0}}}[t,k] + \frac{\partial \boldsymbol{h}}{\partial w_j}[t,k]\frac{\partial w_j}{\partial \boldsymbol{w^k_{j,0}}}[t,k] \ \ (mod\ t^{k+1})
\end{aligned} \tag{3.22}$$

where $mod\ t^{k+1}$ indicates the truncation of the corresponding power series at order k. Equation (3.22) can then be used to calculate $\frac{d\boldsymbol{h}}{d\boldsymbol{x_0}}[t,k]$, $\frac{d\boldsymbol{h}}{d\boldsymbol{\theta}}[t,k]$ and $\frac{d\boldsymbol{h}}{d\boldsymbol{w^k_{j,0}}}[t,k]$ through obtaining $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}[t,k]$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}[t,k]$, $\frac{\partial \boldsymbol{h}}{\partial w_j}[t,k]$, $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}[t,k]$, $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}[t,k]$, $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w^k_{j,0}}}[t,k]$ and $\frac{\partial w_j}{\partial \boldsymbol{w^k_{j,0}}}[t,k]$. $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{h}}{\partial w_j}$ are generally expressions of $\boldsymbol{x}$, $\boldsymbol{\theta}$, $\boldsymbol{u}$ and $\boldsymbol{w}$ and can be calculated symbolically given the expression of $\boldsymbol{h}$. Their power series expansions $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}[t,k]$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}[t,k]$ and $\frac{\partial \boldsymbol{h}}{\partial w_j}[t,k]$, under the random numerical realizations as described at the beginning, can be obtained by substituting $\boldsymbol{x} = \boldsymbol{x}[t,k]$, $\boldsymbol{\theta} = \tilde{\boldsymbol{\theta}}$, $\boldsymbol{u} = \boldsymbol{u}[t,k]$ and $\boldsymbol{w} = \boldsymbol{w}[t,k]$ and truncating at order k. It should be noted that if $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{h}}{\partial w_j}$ are polynomial expressions, such substitution results directly in $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}[t,k]$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}[t,k]$ and $\frac{\partial \boldsymbol{h}}{\partial w_j}[t,k]$. If they are rational expressions, obtaining $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}[t,k]$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}[t,k]$ and $\frac{\partial \boldsymbol{h}}{\partial w_j}[t,k]$ may require additional Taylor series expansion operations or the use of Newton's iteration, which will be introduced later, after the substitution. Amongst all the power series needed to calculate $\frac{d\boldsymbol{h}}{d\boldsymbol{x_0}}[t,k]$, $\frac{d\boldsymbol{h}}{d\boldsymbol{\theta}}[t,k]$ and $\frac{d\boldsymbol{h}}{d\boldsymbol{w^k_{j,0}}}[t,k]$ in equation (3.22), $\boldsymbol{u}[t,k]$, $\boldsymbol{w}[t,k]$ and $\frac{\partial w_j}{\partial \boldsymbol{w^k_{j,0}}}[t,k]$ are readily obtained as:

$$\begin{aligned}
\boldsymbol{u}[t,k] = \tilde{\boldsymbol{u}}_{\boldsymbol{0}} + ... + \tilde{\boldsymbol{u}}_{\boldsymbol{0}}{}^{(k)}\frac{t^k}{k!}, \ \ \boldsymbol{w}[t,k] = \tilde{\boldsymbol{w}}_{\boldsymbol{0}} + ... + \tilde{\boldsymbol{w}}_{\boldsymbol{0}}{}^{(k)}\frac{t^k}{k!} \\
\frac{\partial w_j}{\partial \boldsymbol{w^k_{j,0}}}[t,k] = [1, t, ..., \frac{t^k}{k!}]
\end{aligned} \tag{3.23}$$

The remaining unknowns $\boldsymbol{x}[t,k]$, $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}[t,k]$, $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}[t,k]$ and $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}}[t,k]$ can be calculated indirectly by making use of an auxiliary system of ordinary differential equations.

3.3.1 Variational system

This subsection describes an auxiliary system of ordinary differential equations whose power series expansions of the solutions give $\boldsymbol{x}[t,k]$, $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}[t,k]$, $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}[t,k]$ and $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}}[t,k]$. The system of ODEs, so-called the variational system, is an extension of the equations derived in [86] for rational nonlinear systems with fully measured inputs, i.e. the number of unmeasured inputs $r = 0$. Similar to what was done in [86], express the rational system functions $\boldsymbol{f}$ as fractions with their numerators and denominators polynomial functions:

$$\boldsymbol{f}(\boldsymbol{x},\boldsymbol{\theta},\boldsymbol{u},\boldsymbol{w}) = \begin{bmatrix} \frac{f_{nu,1}}{f_{de,1}} \\ \vdots \\ \frac{f_{nu,n}}{f_{de,n}} \end{bmatrix} = \begin{bmatrix} \frac{1}{f_{de,1}} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{f_{de,n}} \end{bmatrix} \begin{bmatrix} f_{nu,1} \\ \vdots \\ f_{nu,n} \end{bmatrix} \tag{3.24}$$

where $f_{nu,1}$, ..., $f_{nu,n}$ and $f_{de,1}$, ..., $f_{de,n}$ are all polynomial functions of $\boldsymbol{x}$, $\boldsymbol{\theta}$, $\boldsymbol{u}$ and $\boldsymbol{w}$. The state-space equations of the system in (3.2) can then be written as a set of polynomial ordinary differential equations, denoted by $\boldsymbol{P} = \boldsymbol{0}$, in the following form:

$$\boldsymbol{P}(\boldsymbol{x},\dot{\boldsymbol{x}},\boldsymbol{\theta},\boldsymbol{u},\boldsymbol{w}) = \begin{bmatrix} f_{de,1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & f_{de,n} \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix} - \begin{bmatrix} f_{nu,1} \\ \vdots \\ f_{nu,n} \end{bmatrix} = \boldsymbol{0}_{n\times 1} \tag{3.25}$$

Taking derivatives of $\boldsymbol{P} = \boldsymbol{0}$ with respect to $\boldsymbol{x_0}$, $\boldsymbol{\theta}$ and $\boldsymbol{w_{j,0}^k}$ respectively yields:

$$\begin{aligned} \frac{d\boldsymbol{P}}{d\boldsymbol{x_0}} &= \frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}\frac{\partial \dot{\boldsymbol{x}}}{\partial \boldsymbol{x_0}} + \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}} = \boldsymbol{0}_{n\times n} \\ \frac{d\boldsymbol{P}}{d\boldsymbol{\theta}} &= \frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}\frac{\partial \dot{\boldsymbol{x}}}{\partial \boldsymbol{\theta}} + \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}} + \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\theta}} = \boldsymbol{0}_{n\times l} \\ \frac{d\boldsymbol{P}}{d\boldsymbol{w_{j,0}^k}} &= \frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}\frac{\partial \dot{\boldsymbol{x}}}{\partial \boldsymbol{w_{j,0}^k}} + \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}} + \frac{\partial \boldsymbol{P}}{\partial w_j}\frac{\partial w_j}{\partial \boldsymbol{w_{j,0}^k}} = \boldsymbol{0}_{n\times k} \end{aligned} \tag{3.26}$$

Using equations (3.25) and (3.26), the variational system, $\nabla \boldsymbol{P}$, is defined as the following set of differential equations:

$$\nabla \boldsymbol{P}(\boldsymbol{x}, \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}, \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}, \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}}, \boldsymbol{\theta}, \boldsymbol{u}, \boldsymbol{w}) : \begin{cases} \boldsymbol{P}(\boldsymbol{x}, \dot{\boldsymbol{x}}, \boldsymbol{\theta}, \boldsymbol{u}, \boldsymbol{w}) = \boldsymbol{0}_{n\times 1} \\ \frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}} \frac{\partial \dot{\boldsymbol{x}}}{\partial \boldsymbol{x_0}} + \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}} \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}} = \boldsymbol{0}_{n\times n} \\ \frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}} \frac{\partial \dot{\boldsymbol{x}}}{\partial \boldsymbol{\theta}} + \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}} \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}} + \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\theta}} = \boldsymbol{0}_{n\times l} \\ \frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}} \frac{\partial \dot{\boldsymbol{x}}}{\partial \boldsymbol{w_{j,0}^k}} + \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}} \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}} + \frac{\partial \boldsymbol{P}}{\partial w_j} \frac{\partial w_j}{\partial \boldsymbol{w_{j,0}^k}} = \boldsymbol{0}_{n\times k} \end{cases} \tag{3.27}$$

where $\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}$, $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{P}}{\partial w_j}$ are generally expressions of $\boldsymbol{x}$, $\dot{\boldsymbol{x}}$, $\boldsymbol{\theta}$, $\boldsymbol{u}$ and $\boldsymbol{w}$ and can be calculated symbolically after obtaining the expression of $\boldsymbol{P}$. $\nabla \boldsymbol{P}$ is a system of first order polynomial ODEs with dependent variables $\boldsymbol{x}$, $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}$, $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}}$ and independent variable t. The associated initial conditions are:

$$\boldsymbol{x}\big|_{t=0} = \boldsymbol{x_0}, \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}\bigg|_{t=0} = \boldsymbol{I}_{n\times n}, \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}\bigg|_{t=0} = \boldsymbol{0}_{n\times l}, \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}}\bigg|_{t=0} = \boldsymbol{0}_{n\times k} \tag{3.28}$$

Using the random numerical realizations and letting $\boldsymbol{\theta} = \tilde{\boldsymbol{\theta}}$, $\boldsymbol{u} = \boldsymbol{u}[t,k]$, $\boldsymbol{w} = \boldsymbol{w}[t,k]$ and $\frac{\partial w_j}{\partial \boldsymbol{w_{j,0}^k}} = \frac{\partial w_j}{\partial \boldsymbol{w_{j,0}^k}}[t,k]$, the occurring polynomial differential system $\nabla \boldsymbol{P}(\boldsymbol{x}, \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}, \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}, \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}}, \tilde{\boldsymbol{\theta}}, \boldsymbol{u}[t,k], \boldsymbol{w}[t,k])$ then allows for the use of Newton's iteration method to obtain the power series solutions of its dependent variables up to order k, i.e. $\boldsymbol{x}[t,k]$, $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}[t,k]$, $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}[t,k]$ and $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}}[t,k]$.

3.3.2 Newton's iteration

The details about the use of Newton's iteration for finding power series solutions of polynomial ordinary differential equations up to an order of interest can be seen in Chapter 4 of [82], Section 5 of [16] and Section 3 of [86].

This subsection, first of all, places the attention on the first equation of $\nabla \boldsymbol{P}(\boldsymbol{x}, \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}, \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}, \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}}, \tilde{\boldsymbol{\theta}}, \tilde{\boldsymbol{u}}[t,k], \tilde{\boldsymbol{w}}[t,k])$, i.e.

$$\boldsymbol{P}(\boldsymbol{x}, \dot{\boldsymbol{x}}, \tilde{\boldsymbol{\theta}}, \boldsymbol{u}[t,k], \boldsymbol{w}[t,k]) = \boldsymbol{0}_{n\times 1} \tag{3.29}$$

The objective of the Newton's iteration is to solve equation (3.29) for the power series expansion of $\boldsymbol{x}$ up to order k, i.e. $\boldsymbol{x}[t,k]$, through a number of iterative computations. The iterations start from the initial condition $\boldsymbol{x_0}$ which is the first term of $\boldsymbol{x}[t,k]$. At each iteration, the power series expansion of $\boldsymbol{x}$ is computed up to a certain order smaller than k, and it is then used for the next iteration to compute the power series up to a higher order closer to k. The iterations are terminated when order k is reached. Let $\boldsymbol{x_q}$ denote the power series computed at the q^{th} iteration of the Newton's method. The formula of Newton's iteration for equation (3.29) is obtained by means of linearization of the equation around the point $\boldsymbol{x} = \boldsymbol{x_{q-1}}$, yielding:

$$\left.\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}\right|_{\boldsymbol{x}=\boldsymbol{x_{q-1}}} (\dot{\boldsymbol{x}}_{\boldsymbol{q}} - \dot{\boldsymbol{x}}_{\boldsymbol{q-1}}) + \left.\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}\right|_{\dot{\boldsymbol{x}}=\dot{\boldsymbol{x}}_{\boldsymbol{q-1}},\boldsymbol{x}=\boldsymbol{x_{q-1}}} (\boldsymbol{x_q} - \boldsymbol{x_{q-1}}) + \left.\boldsymbol{P}\right|_{\dot{\boldsymbol{x}}=\dot{\boldsymbol{x}}_{\boldsymbol{q-1}},\boldsymbol{x}=\boldsymbol{x_{q-1}}} = \boldsymbol{0}_{n\times 1} \tag{3.30}$$

where $\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}$, $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}$ and $\boldsymbol{P}$ are matrices with power series elements after evaluating at $\boldsymbol{x} = \boldsymbol{x_{q-1}}$. Equation (3.30) is thus a first order linear ODE of the form:

$$\boldsymbol{C_1}(t)\dot{\boldsymbol{v}} + \boldsymbol{C_2}(t)\boldsymbol{v} + \boldsymbol{C_3}(t) = \boldsymbol{0} \tag{3.31}$$

Equation (3.31) can be solved analytically using the method of integrating factor, giving the solution:

$$\boldsymbol{v}(t) = -e^{-\int \boldsymbol{C_1}^{-1}\boldsymbol{C_2}dt} \int e^{\int \boldsymbol{C_1}^{-1}\boldsymbol{C_2}dt}(\boldsymbol{C_1}^{-1}\boldsymbol{C_3})dt + \boldsymbol{C_4}e^{-\int \boldsymbol{C_1}^{-1}\boldsymbol{C_2}dt} \tag{3.32}$$

where $\boldsymbol{C_1}$ is an invertible matrix, and $\boldsymbol{C_4}$ is a matrix of constants which is determined by knowing the initial condition $\boldsymbol{v}(0)$. Following equation (3.32), the correction term $\boldsymbol{E_q}(t)$ defined as

$$\boldsymbol{E_q}(t) = \boldsymbol{x_q} - \boldsymbol{x_{q-1}}, \;\; \boldsymbol{E_q}(0) = \boldsymbol{0} \tag{3.33}$$

is determined as an analytical solution to equation (3.30), and $\boldsymbol{x_q}$ can then be ex-

pressed in terms of $\boldsymbol{x_{q-1}}$ as $\boldsymbol{x_q} = \boldsymbol{x_{q-1}} + \boldsymbol{E_q}(t)$:

$$\boldsymbol{x_q} = \boldsymbol{x_{q-1}} - e^{-\int (\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}})^{-1} \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}} dt} \int e^{\int (\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}})^{-1} \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}} dt} (\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}^{-1} \boldsymbol{P}) dt \tag{3.34}$$

Within equation (3.34), the formal integrations and differentiations are allowed to perform over power series, and the power series of the matrix exponentials and inverses involved can be computed efficiently by using a nested Newton's iteration (see details in [86] and [16]). For example, if $\boldsymbol{A_g}$ and $\boldsymbol{B_g}$ represent respectively the power series expansions of $e^{-\int (\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}})^{-1} \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}} dt}$ and its inverse $e^{\int (\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}})^{-1} \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}} dt}$ computed at the g^{th} iteration of the Newton's iteration, the formula for recursively updating $\boldsymbol{A_g}$ and $\boldsymbol{B_g}$ is given by:

$$\begin{aligned} \boldsymbol{A_g} &= \boldsymbol{A_{g-1}} + \boldsymbol{A_{g-1}} \int \boldsymbol{B_{g-1}} (-(\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}})^{-1} \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}) \boldsymbol{A_{g-1}} - \boldsymbol{B_{g-1}} \dot{\boldsymbol{A}}_{\boldsymbol{g-1}} dt \\ \boldsymbol{B_g} &= 2\boldsymbol{B_{g-1}} - \boldsymbol{B_{g-1}} \boldsymbol{A_g} \boldsymbol{B_{g-1}} \end{aligned} \tag{3.35}$$

with the initial conditions $\boldsymbol{A_0} = \boldsymbol{B_0} = \boldsymbol{I}_{n \times n}$. Equation (3.34) therefore becomes:

$$\boldsymbol{x_q} = \boldsymbol{x_{q-1}} - \boldsymbol{A_q} \int \boldsymbol{B_q} (\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}^{-1} \boldsymbol{P}) dt \tag{3.36}$$

Starting from $\boldsymbol{x_0} = \tilde{\boldsymbol{x}}_{\boldsymbol{0}}$ under the numerical realizations, equation (3.34) can readily be used to update $\boldsymbol{x_q}$ from $\boldsymbol{x_{q-1}}$, where $\boldsymbol{x_q}$ is the power series expansion of $\boldsymbol{x}$ containing more correct terms than $\boldsymbol{x_{q-1}}$, i.e. is correct up to a higher order than $\boldsymbol{x_{q-1}}$. It has been proven that the convergence of Newton's iteration is quadratic, which means the number of additional correct terms computed doubles at each iteration [16]. At the q^{th} iteration, the first 2^q terms of $\boldsymbol{x_q}$ are correct, and it is therefore truncated at order $2^q - 1$, i.e. $\boldsymbol{x_q} = \boldsymbol{x}[t, 2^q - 1]$, to remove the incorrect terms of larger orders. The iterative process is continued until the q_0^{th} iteration when the maximum order of correct terms $2^{q_0} - 1 \geq k$. The computed power series $\boldsymbol{x_{q_0}} = \boldsymbol{x}[t, 2^{q_0} - 1]$ is then truncated at order k leading to the result of $\boldsymbol{x}[t, k]$.

Having obtained $\boldsymbol{x}[t,k]$ from the first equation of $\nabla \boldsymbol{P}(\boldsymbol{x}, \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}, \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}, \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}}, \tilde{\boldsymbol{\theta}}, \tilde{\boldsymbol{u}}[t,k], \tilde{\boldsymbol{w}}[t,k])$, it proceeds to compute $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}[t,k]$, $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}[t,k]$ and $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}}[t,k]$ based on the later three equations, i.e.

$$\begin{aligned}
\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}(\cdot)\frac{\partial \dot{\boldsymbol{x}}}{\partial \boldsymbol{x_0}} + \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}(\cdot)\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}} &= \mathbf{0}_{n\times n} \\
\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}(\cdot)\frac{\partial \dot{\boldsymbol{x}}}{\partial \boldsymbol{\theta}} + \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}(\cdot)\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}} + \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\theta}}(\cdot) &= \mathbf{0}_{n\times l} \\
\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}(\cdot)\frac{\partial \dot{\boldsymbol{x}}}{\partial \boldsymbol{w_{j,0}^k}} + \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}(\cdot)\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}} + \frac{\partial \boldsymbol{P}}{\partial w_j}(\cdot)\frac{\partial w_j}{\partial \boldsymbol{w_{j,0}^k}}[t,k] &= \mathbf{0}_{n\times k}
\end{aligned} \tag{3.37}$$

where $(\cdot)$ denotes $(\boldsymbol{x}, \dot{\boldsymbol{x}}, \tilde{\boldsymbol{\theta}}, \tilde{\boldsymbol{u}}[t,k], \tilde{\boldsymbol{w}}[t,k])$ for brevity. Since the power series solutions of equation (3.37) up to the accuracy of order k are only considered, substitute $\boldsymbol{x}[t,k]$ into the equations,

$$\begin{aligned}
\left.\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}\right|_{\boldsymbol{x}=\boldsymbol{x}[t,k]}\frac{\partial \dot{\boldsymbol{x}}}{\partial \boldsymbol{x_0}} + \left.\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}[t,k],\dot{\boldsymbol{x}}=\dot{\boldsymbol{x}}[t,k]}\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}} &= \mathbf{0}_{n\times n} \\
\left.\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}\right|_{\boldsymbol{x}=\boldsymbol{x}[t,k]}\frac{\partial \dot{\boldsymbol{x}}}{\partial \boldsymbol{\theta}} + \left.\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}[t,k],\dot{\boldsymbol{x}}=\dot{\boldsymbol{x}}[t,k]}\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}} + \left.\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\theta}}\right|_{\boldsymbol{x}=\boldsymbol{x}[t,k],\dot{\boldsymbol{x}}=\dot{\boldsymbol{x}}[t,k]} &= \mathbf{0}_{n\times l} \\
\left.\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}\right|_{\boldsymbol{x}=\boldsymbol{x}[t,k]}\frac{\partial \dot{\boldsymbol{x}}}{\partial \boldsymbol{w_{j,0}^k}} + \left.\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}[t,k],\dot{\boldsymbol{x}}=\dot{\boldsymbol{x}}[t,k]}\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}} + \left.\frac{\partial \boldsymbol{P}}{\partial w_j}\right|_{\boldsymbol{x}=\boldsymbol{x}[t,k],\dot{\boldsymbol{x}}=\dot{\boldsymbol{x}}[t,k]}\frac{\partial w_j}{\partial \boldsymbol{w_{j,0}^k}}[t,k] &= \mathbf{0}_{n\times k}
\end{aligned} \tag{3.38}$$

All the three equations in (3.38) are first order linear ODEs with given initial conditions in equation (3.28), and are thus solvable using equation (3.32). The following solutions are obtained:

$$\begin{aligned}
\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}} &= e^{-\int (\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}})^{-1}\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}dt} \\
\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}} &= -e^{-\int (\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}})^{-1}\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}dt}\int e^{\int (\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}})^{-1}\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}dt}(\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}^{-1}\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\theta}})dt \\
\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}} &= -e^{-\int (\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}})^{-1}\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}dt}\int e^{\int (\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}})^{-1}\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}dt}(\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}^{-1}\frac{\partial \boldsymbol{P}}{\partial w_j}\frac{\partial w_j}{\partial \boldsymbol{w_{j,0}^k}}[t,k])dt
\end{aligned} \tag{3.39}$$

With the power series manipulations of integration, differentiation, matrix exponential and inverse as previously described, the power series forms of $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}$, $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}}$ can be obtained from equation (3.39). These power series contain correct terms up to order k, leading to the results of $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}[t,k]$, $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}[t,k]$ and $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}}[t,k]$ through truncation.

3.3.3 The algorithm and modular operations

It is now sufficient to summarize the procedure of computing the k-row observability matrix in equations (3.16) and (3.17) based on the power series methods described in the previous subsections. The algorithm is presented in the following:

Algorithm 3.2

Input to the algorithm: the state-space and measurement equations of the system in equation (3.2), and the choice of the maximum order of derivatives of the unmeasured inputs taken into account $k = k_0$

Output from the algorithm: the k-row observability matrix

Preprocessing: Compute $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{h}}{\partial w_j}$ $(j = 1, ..., r)$ symbolically. Convert the state-space equations to $\boldsymbol{P} = \boldsymbol{0}$ using equation (3.25), and compute $\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}$, $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{P}}{\partial w_j}$ symbolically

Initialization: Choose the sets of positive integers randomly $\tilde{\boldsymbol{x}}_{\boldsymbol{0}}$, $\tilde{\boldsymbol{\theta}}$, $\tilde{\boldsymbol{w}}_{\boldsymbol{0}}, ..., \tilde{\boldsymbol{w}}_{\boldsymbol{0}}{}^{(k)}$ and $\tilde{\boldsymbol{u}}_{\boldsymbol{0}}, ..., \tilde{\boldsymbol{u}}_{\boldsymbol{0}}{}^{(k)}$. Let $\boldsymbol{u}[t,k] = \tilde{\boldsymbol{u}}_{\boldsymbol{0}} + ... + \tilde{\boldsymbol{u}}_{\boldsymbol{0}}{}^{(k)}\frac{t^k}{k!}$, $\boldsymbol{w}[t,k] = \tilde{\boldsymbol{w}}_{\boldsymbol{0}} + ... + \tilde{\boldsymbol{w}}_{\boldsymbol{0}}{}^{(k)}\frac{t^k}{k!}$, and evaluate $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}$, $\frac{\partial \boldsymbol{h}}{\partial w_j}$, $\boldsymbol{P}$, $\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}$, $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{P}}{\partial w_j}$ at $\boldsymbol{\theta} = \tilde{\boldsymbol{\theta}}, \boldsymbol{u} = \boldsymbol{u}[t,k], \boldsymbol{w} = \boldsymbol{w}[t,k]$. Set $\boldsymbol{x}_{\boldsymbol{0}} = \tilde{\boldsymbol{x}}_{\boldsymbol{0}}$ and $q = 0$

1. Compute $\boldsymbol{x}[t,k]$ through Newton's iteration:

a. Set $q = q + 1$;

b. Evaluate $\boldsymbol{P}$, $\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}$ and $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}$ at $\dot{\boldsymbol{x}} = \dot{\boldsymbol{x}}_{\boldsymbol{q-1}}$, $\boldsymbol{x} = \boldsymbol{x}_{\boldsymbol{q-1}}$, truncated at order $2^q - 1$;

c. Set $g = 0$ and $\boldsymbol{A}_{\boldsymbol{0}} = \boldsymbol{B}_{\boldsymbol{0}} = \boldsymbol{I}_{n\times n}$, and compute $\boldsymbol{A}_{\boldsymbol{q}}$ and $\boldsymbol{B}_{\boldsymbol{q}}$:

i. Set $g = g + 1$;

ii. Compute $\boldsymbol{A}_{\boldsymbol{g}} = \boldsymbol{A}_{\boldsymbol{g-1}} + \boldsymbol{A}_{\boldsymbol{g-1}} \int \boldsymbol{B}_{\boldsymbol{g-1}}(-(\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}})^{-1}\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}})\boldsymbol{A}_{\boldsymbol{g-1}} - \boldsymbol{B}_{\boldsymbol{g-1}}\dot{\boldsymbol{A}}_{\boldsymbol{g-1}}dt$

truncated at order $2^g - 1$;

iii. Compute $\boldsymbol{B_g} = 2\boldsymbol{B_{g-1}} - \boldsymbol{B_{g-1}}\boldsymbol{A_g}\boldsymbol{B_{g-1}}$ truncated at order $2^g - 1$;

iv. Go to step **d** if $g = q$. Otherwise go to step **i**;

d. Compute $\boldsymbol{x_q} = \boldsymbol{x_{q-1}} - \boldsymbol{A_q}\int \boldsymbol{B_q}(\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}^{-1}\boldsymbol{P})dt$ truncated at order $2^q - 1$;

e. If $2^q - 1 \geq k$, $\boldsymbol{x}[t,k] = \boldsymbol{x_q}$ truncated at order k and go to step **2**.

Otherwise go to step **a**;

2. Evaluate $\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}$, $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{P}}{\partial w_j}$ at $\dot{\boldsymbol{x}} = \dot{\boldsymbol{x}}[t,k], \boldsymbol{x} = \boldsymbol{x}[t,k]$, truncated at order k;

3. Set $g = 0$ and $\boldsymbol{A_0} = \boldsymbol{B_0} = \boldsymbol{I}_{n\times n}$, and compute $\boldsymbol{A_q}$ and $\boldsymbol{B_q}$:

a. Set $g = g + 1$;

b. Compute $\boldsymbol{A_g} = \boldsymbol{A_{g-1}} + \boldsymbol{A_{g-1}}\int \boldsymbol{B_{g-1}}(-(\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}})^{-1}\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}})\boldsymbol{A_{g-1}} - \boldsymbol{B_{g-1}}\dot{\boldsymbol{A}}_{\boldsymbol{g-1}}dt$

truncated at order $2^g - 1$;

c. Compute $\boldsymbol{B_g} = 2\boldsymbol{B_{g-1}} - \boldsymbol{B_{g-1}}\boldsymbol{A_g}\boldsymbol{B_{g-1}}$ truncated at order $2^g - 1$;

d. If $g = q$, truncate $\boldsymbol{A_q}$ and $\boldsymbol{B_q}$ at order k and go to step **4**. Otherwise

go to step **a**;

4. Compute $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}[t,k] = \boldsymbol{A_q}$ truncated at order k;

5. Compute $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}[t,k] = -\boldsymbol{A_q}\int \boldsymbol{B_q}(\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}^{-1}\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\theta}})dt$ truncated at order k;

6. Set $\frac{\partial w_j}{\partial \boldsymbol{w}^{\boldsymbol{k}}_{\boldsymbol{j,0}}}[t,k] = [1, t, ..., \frac{t^k}{k!}]$, and compute $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w}^{\boldsymbol{k}}_{\boldsymbol{j,0}}}[t,k] = -\boldsymbol{A_q}\int \boldsymbol{B_q}(\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}^{-1}\frac{\partial \boldsymbol{P}}{\partial w_j}\frac{\partial w_j}{\partial \boldsymbol{w}^{\boldsymbol{k}}_{\boldsymbol{j,0}}}[t,k])dt$

truncated at order k;

7. Compute $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}[t,k]$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}[t,k]$ and $\frac{\partial \boldsymbol{h}}{\partial w_j}[t,k]$ by evaluating $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{h}}{\partial w_j}$ at $\boldsymbol{x} = \boldsymbol{x}[t,k]$,

truncated at order k. Take Taylor series expansions at $t = 0$ up to order k if $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}$

and $\frac{\partial \boldsymbol{h}}{\partial w_j}$ are rational expressions;

8. Compute $\frac{d\boldsymbol{h}}{d\boldsymbol{x_0}}[t,k] = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}[t,k]\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}[t,k]$ truncated at order k;

9. Compute $\frac{d\boldsymbol{h}}{d\boldsymbol{\theta}}[t,k] = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}[t,k]\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}[t,k] + \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}[t,k]$ truncated at order k;

10. Compute $\frac{d\boldsymbol{h}}{d\boldsymbol{w}^{\boldsymbol{k}}_{\boldsymbol{j,0}}}[t,k] = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}[t,k]\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w}^{\boldsymbol{k}}_{\boldsymbol{j,0}}}[t,k] + \frac{\partial \boldsymbol{h}}{\partial w_j}[t,k]\frac{\partial w_j}{\partial \boldsymbol{w}^{\boldsymbol{k}}_{\boldsymbol{j,0}}}[t,k]$ truncated at order k;

11. Extract the coefficients of $\frac{d\boldsymbol{h}}{d\boldsymbol{x_0}}[t,k]$, $\frac{d\boldsymbol{h}}{d\boldsymbol{\theta}}[t,k]$ and $\frac{d\boldsymbol{h}}{d\boldsymbol{w}^{\boldsymbol{k}}_{\boldsymbol{j,0}}}[t,k]$ to respectively construct $d\boldsymbol{\Omega}^{\boldsymbol{k}}_{\boldsymbol{x}}$, $d\boldsymbol{\Omega}^{\boldsymbol{k}}_{\boldsymbol{\theta}}$ and $d\boldsymbol{\Omega}^{\boldsymbol{k}}_{\boldsymbol{w_j}}$ based on the expressions in equations (3.17) and (3.20);

12. Arrange the observability matrix $d\boldsymbol{\Omega}^{\boldsymbol{k}} = [d\boldsymbol{\Omega}^{\boldsymbol{k}}_{\boldsymbol{x}}, d\boldsymbol{\Omega}^{\boldsymbol{k}}_{\boldsymbol{\theta}}, d\boldsymbol{\Omega}^{\boldsymbol{k}}_{\boldsymbol{w_1}}, ..., d\boldsymbol{\Omega}^{\boldsymbol{k}}_{\boldsymbol{w_r}}]$.

Similarly to what was discussed in [86] for the case of all inputs being measured, in Algorithm 3.2 all the computations between power series can be done with the application of modular operations so as to further improve the efficiency. The means of achieving so is through selecting a large positive prime number p at the initialization step and choosing the random integers within the range 1 to $p-1$ for numerical realizations; all the occurring power series throughout the algorithm are then treated with their coefficients *modulo* p. The arithmetic of modular operations can be simply described as follows [90]: a positive integer a *modulo* p calculates the remainder of a divided by p, resulting in an integer between 0 and $p-1$; a fraction of two positive integers b/c *modulo* p, based on the rule of modular multiplicative inverse, also gives an integer between 0 and $p-1$. As a consequence of applying the modular operations, all the occurring coefficients of power series throughout the algorithm are restricted within the range 1 to $p-1$. The problem of number growth is thus effectively controlled to prevent expensive computations between very large numbers.

Taking into account the modular operations, the output from the proposed algorithm is equivalent to the following matrix:

$$d\boldsymbol{\Omega}^{\boldsymbol{k}} \ \ modulo \ \ p \tag{3.40}$$

where the modulo operations are used elementwise. The observability of the tested system is then determined by evaluating the rank of the matrix over a finite field $\mathbb{F}_p$ through applying the modular operations to Gaussian elimination [86, 90, 14, 91].

As discussed in [86], $d\boldsymbol{\Omega^k}$ and the matrix in (3.40) contain almost identical rank information over the corresponding fields and can hence deliver agreed observability results with a very high probability of success. The probability of disagreement, when the observability results obtained with and without the modular operations do not coincide, is vanishingly small if p is selected to be large as shown in [86]. This probability can even become practically negligible if the algorithm is repeated using different choices of p and numerical realizations.

A proper choice of $k = k_0$ such that Algorithm 3.2 computes the corresponding $d\boldsymbol{\Omega^k}$ from which the user can extract convergent observability information might not be straightforward for some large systems. In the following, a recursive version of Algorithm 3.2, termed as Algorithm 3.3, is presented. In the algorithm, k is incrementally changed to match the order of the leading term of the power series produced from Newton's iteration, i.e. $k = 2^q - 1$. At each increment of k, the corresponding $d\boldsymbol{\Omega^k}$ is computed and the (k-row) observability properties of the components of $\boldsymbol{x^k}$ are detected following the procedure described in Section 3.2. In the case of encountering observable systems, the iterations terminate when it reaches a k such that $rank(d\boldsymbol{\Omega^k}) = n + l + (k+1)r$ or $\boldsymbol{x}$, $\boldsymbol{\theta}$ and $\boldsymbol{w}$ are found to be observable.

Algorithm 3.3

Input to the algorithm: the state-space and measurement equations of the system in equation (3.2)

Output from the algorithm: the (k-row) observability of the tested system

Preprocessing: Compute $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{h}}{\partial w_j}$ $(j = 1, ..., r)$ symbolically. Convert the state-space equations to $\boldsymbol{P} = \boldsymbol{0}$ using equation (3.25), and compute $\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}$, $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{P}}{\partial w_j}$ symbolically

Initialization: Choose the sets of positive integers randomly $\tilde{\boldsymbol{x}}_{\mathbf{0}}$, $\tilde{\boldsymbol{\theta}}$, $\tilde{\boldsymbol{w}}_{\mathbf{0}}, ..., \tilde{\boldsymbol{w}}_{\mathbf{0}}{}^{(k)}$ and $\tilde{\boldsymbol{u}}_{\mathbf{0}}, ..., \tilde{\boldsymbol{u}}_{\mathbf{0}}{}^{(k)}$. Let $\boldsymbol{u}[t,k] = \tilde{\boldsymbol{u}}_{\mathbf{0}} + ... + \tilde{\boldsymbol{u}}_{\mathbf{0}}{}^{(k)}\frac{t^k}{k!}$, $\boldsymbol{w}[t,k] = \tilde{\boldsymbol{w}}_{\mathbf{0}} + ... + \tilde{\boldsymbol{w}}_{\mathbf{0}}{}^{(k)}\frac{t^k}{k!}$, and evaluate $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}$, $\frac{\partial \boldsymbol{h}}{\partial w_j}$, $\boldsymbol{P}$, $\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}$, $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{P}}{\partial w_j}$ at $\boldsymbol{\theta} = \tilde{\boldsymbol{\theta}}, \boldsymbol{u} = \boldsymbol{u}[t,k], \boldsymbol{w} = \boldsymbol{w}[t,k]$. Set $\boldsymbol{x}_{\mathbf{0}} = \tilde{\boldsymbol{x}}_{\mathbf{0}}$ and $q = 0$

1. Set $q = q + 1$;

2. Evaluate $\boldsymbol{P}$, $\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}$ and $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}$ at $\dot{\boldsymbol{x}} = \dot{\boldsymbol{x}}_{\boldsymbol{q-1}}$, $\boldsymbol{x} = \boldsymbol{x}_{\boldsymbol{q-1}}$, truncated at order $2^q - 1$;

3. Set $g = 0$ and $\boldsymbol{A}_{\mathbf{0}} = \boldsymbol{B}_{\mathbf{0}} = \boldsymbol{I}_{n\times n}$, and compute $\boldsymbol{A}_{\boldsymbol{q}}$ and $\boldsymbol{B}_{\boldsymbol{q}}$:

a. Set $g = g + 1$;

b. Compute $\boldsymbol{A}_{\boldsymbol{g}} = \boldsymbol{A}_{\boldsymbol{g-1}} + \boldsymbol{A}_{\boldsymbol{g-1}} \int \boldsymbol{B}_{\boldsymbol{g-1}}(-(\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}})^{-1}\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}})\boldsymbol{A}_{\boldsymbol{g-1}} - \boldsymbol{B}_{\boldsymbol{g-1}}\dot{\boldsymbol{A}}_{\boldsymbol{g-1}}dt$ truncated at order $2^g - 1$;

c. Compute $\boldsymbol{B}_{\boldsymbol{g}} = 2\boldsymbol{B}_{\boldsymbol{g-1}} - \boldsymbol{B}_{\boldsymbol{g-1}}\boldsymbol{A}_{\boldsymbol{g}}\boldsymbol{B}_{\boldsymbol{g-1}}$ truncated at order $2^g - 1$;

d. Go to step **4** if $g = q$. Otherwise go to step **a**;

4. Compute $\boldsymbol{x}_{\boldsymbol{q}} = \boldsymbol{x}_{\boldsymbol{q-1}} - \boldsymbol{A}_{\boldsymbol{q}} \int \boldsymbol{B}_{\boldsymbol{q}}(\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}^{-1}\boldsymbol{P})dt$ truncated at order $2^q - 1$;

5. Set $\boldsymbol{x}[t,k] = \boldsymbol{x}_{\boldsymbol{q}}$;

6. Evaluate $\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}$, $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{P}}{\partial w_j}$ at $\dot{\boldsymbol{x}} = \dot{\boldsymbol{x}}[t,k], \boldsymbol{x} = \boldsymbol{x}[t,k]$, truncated at order k;

7. Set $g = 0$ and $\boldsymbol{A}_{\mathbf{0}} = \boldsymbol{B}_{\mathbf{0}} = \boldsymbol{I}_{n\times n}$, and compute $\boldsymbol{A}_{\boldsymbol{q}}$ and $\boldsymbol{B}_{\boldsymbol{q}}$:

a. Set $g = g + 1$;

b. Compute $\boldsymbol{A}_{\boldsymbol{g}} = \boldsymbol{A}_{\boldsymbol{g-1}} + \boldsymbol{A}_{\boldsymbol{g-1}} \int \boldsymbol{B}_{\boldsymbol{g-1}}(-(\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}})^{-1}\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{x}})\boldsymbol{A}_{\boldsymbol{g-1}} - \boldsymbol{B}_{\boldsymbol{g-1}}\dot{\boldsymbol{A}}_{\boldsymbol{g-1}}dt$ truncated at order $2^g - 1$;

c. Compute $\boldsymbol{B}_{\boldsymbol{g}} = 2\boldsymbol{B}_{\boldsymbol{g-1}} - \boldsymbol{B}_{\boldsymbol{g-1}}\boldsymbol{A}_{\boldsymbol{g}}\boldsymbol{B}_{\boldsymbol{g-1}}$ truncated at order $2^g - 1$;

d. If $g = q$, truncate $\boldsymbol{A}_{\boldsymbol{q}}$ and $\boldsymbol{B}_{\boldsymbol{q}}$ at order k and go to step **8**. Otherwise go to step **a**;

8. Compute $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x}_{\mathbf{0}}}[t,k] = \boldsymbol{A}_{\boldsymbol{q}}$ truncated at order k;

9. Compute $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}[t,k] = -\boldsymbol{A_q} \int \boldsymbol{B_q}(\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}^{-1} \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\theta}})dt$ truncated at order k;

10. Set $\frac{\partial w_j}{\partial \boldsymbol{w_{j,0}^k}}[t,k] = [1, t, ..., \frac{t^k}{k!}]$, and compute $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}}[t,k] = -\boldsymbol{A_q} \int \boldsymbol{B_q}(\frac{\partial \boldsymbol{P}}{\partial \dot{\boldsymbol{x}}}^{-1} \frac{\partial \boldsymbol{P}}{\partial w_j} \frac{\partial w_j}{\partial \boldsymbol{w_{j,0}^k}}[t,k])dt$ truncated at order k;

11. Compute $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}[t,k]$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}[t,k]$ and $\frac{\partial \boldsymbol{h}}{\partial w_j}[t,k]$ by evaluating $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{h}}{\partial w_j}$ at $\boldsymbol{x} = \boldsymbol{x}[t,k]$, truncated at order k. Take Taylor series expansions at $t = 0$ up to order k if $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}$ and $\frac{\partial \boldsymbol{h}}{\partial w_j}$ are rational expressions;

12. Compute $\frac{d\boldsymbol{h}}{d\boldsymbol{x_0}}[t,k] = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}[t,k] \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x_0}}[t,k]$ truncated at order k;

13. Compute $\frac{d\boldsymbol{h}}{d\boldsymbol{\theta}}[t,k] = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}[t,k] \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\theta}}[t,k] + \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\theta}}[t,k]$ truncated at order k;

14. Compute $\frac{d\boldsymbol{h}}{d\boldsymbol{w_{j,0}^k}}[t,k] = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}[t,k] \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{w_{j,0}^k}}[t,k] + \frac{\partial \boldsymbol{h}}{\partial w_j}[t,k] \frac{\partial w_j}{\partial \boldsymbol{w_{j,0}^k}}[t,k]$ truncated at order k;

15. Extract the coefficients of $\frac{d\boldsymbol{h}}{d\boldsymbol{x_0}}[t,k]$, $\frac{d\boldsymbol{h}}{d\boldsymbol{\theta}}[t,k]$ and $\frac{d\boldsymbol{h}}{d\boldsymbol{w_{j,0}^k}}[t,k]$ to respectively construct $d\boldsymbol{\Omega_x^k}$, $d\boldsymbol{\Omega_\theta^k}$ and $d\boldsymbol{\Omega_{w_j}^k}$ based on the expressions in equations (3.17) and (3.20);

16. Arrange the observability matrix $d\boldsymbol{\Omega^k} = [d\boldsymbol{\Omega_x^k}, d\boldsymbol{\Omega_\theta^k}, d\boldsymbol{\Omega_{w_1}^k}, ..., d\boldsymbol{\Omega_{w_r}^k}]$;

17. End if $rank(d\boldsymbol{\Omega^k}) = n + l + (k+1)r$, and $\boldsymbol{x}$, $\boldsymbol{\theta}$ and $\boldsymbol{w}$ are detected observable;

18. Set $i = 0$. Detect the (k-row) observability properties of the components of $\boldsymbol{x^k}$:

a. Set $i = i + 1$;

b. Remove the i^{th} column of $d\boldsymbol{\Omega^k}$;

c. If the rank of the occurring matrix$< rank(d\boldsymbol{\Omega^k})$, the i^{th} state is observable. Otherwise the state is k-row unobservable;

d. Go to step **19** if $i = n + l + (k+1)r$. Otherwise go to step **a**;

19. End if $\boldsymbol{x}$, $\boldsymbol{\theta}$ and $\boldsymbol{w}$ are detected observable. Otherwise go to step **1**;

Algorithm 3.3 succeeds in identifying an observable system or the observable quantities (dynamic states, parameters or unmeasured inputs) of an unobservable system. Concluding the observability of those k-row unobservable quantities in a rigorous mathematical manner, or in other words proving the convergence of the k-row un-

observable quantities, however, has not yet been feasible. In the examples of this chapter, a practical guideline is provided on how to conclude the unobservable quantities with sufficient confidence by making use of the submatrices of the observability matrix of a large k obtained from Algorithm 3.2 or 3.3. Theoretically addressing this convergence problem will be the focus of future extensions of this work.

3.3.4 Remarks

The proposed algorithm in this work, the version of Algorithm 3.2 or 3.3, can be implemented efficiently in a variety of software with symbolic computing tools, such as MATLAB with symbolic toolbox [71], Maple [69] and Mathematica [110]. Compared to the purely symbolic implementation of Algorithm 3.1 presented in Section 3.2, several advantages of using the proposed algorithm for observability testing are remarked in the following. Firstly, the algorithm uses numerical realizations of the involved variables and the technique of modular operations to control number growth. With the exception of the preprocessing step, gradients and differentiations of symbolic expressions are completely avoided; the arithmetics of power series with numerical coefficients throughout the algorithm are substantially cheaper than those computations between symbolic expressions. The second advantage is due to the utilization of the Newton's iteration method. Newton's iteration allows to compute a number of terms of power series at each iteration twice as many as the previous iteration, and the number of terms of power series is in fact equivalent to the number of the Lie derivatives of corresponding orders. Due to this quadratic convergence property of Newton's iteration, a large number of the Lie derivatives can therefore be computed through only a small number of iterations. In comparison, Algorithm 3.1 uses formulation (3.5) to compute one order of Lie derivative only each time. Benefiting from

the above advantages, Algorithm 3.2 allows to efficiently compute the observability matrix of a large and complicated system for the choice of a very large k, where it should be noted that the observability matrix of a large k is more likely to contain convergent observability information than the matrix of a smaller k. Similarly in Algorithm 3.3, it is able to operate up to a very large k for a system of large size and high complexity.

3.4 Illustrative examples

3.4.1 A 2 degrees of freedom (DOFs) nonlinear system

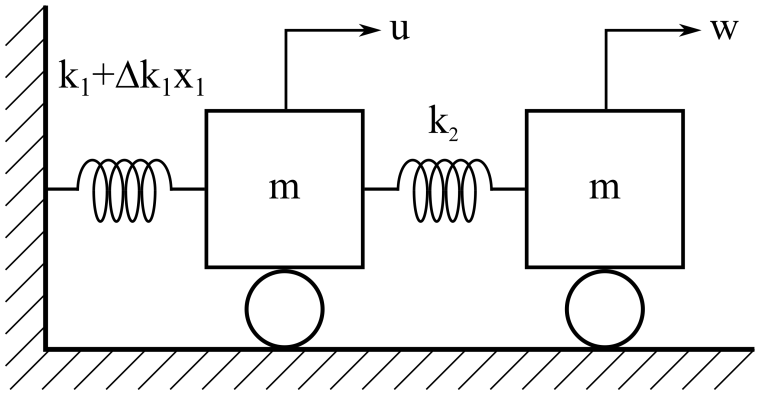

Figure 3.1: A 2 DOFs nonlinear system driven by measured and unmeasured inputs

This example demonstrates the use of the proposed algorithm to examine the observability of a 2 DOFs nonlinear system driven by both measured and unmeasured inputs as shown in Figure 3.1. Let x_1, x_2, v_1, v_2 and $\dot{v}_1$, $\dot{v}_2$ denote the displacements, velocities and accelerations of the two masses of the system respectively. A measured input u is applied at the first mass and an unmeasured input w is applied at the second mass. The stiffness of the nonlinear spring that connects the first mass to the fixed support is displacement-proportional and given by $k_1 + \Delta k_1 x_1$. The linear spring connecting the two masses has the stiffness of k_2. The dynamic states of the

system are x_1, x_2, v_1 and v_2, and the unknown parameters to be identified are k_1, Δk_1, k_2 and m. The state-space equations of the dynamical system can therefore be written as:

$$\dot{\boldsymbol{x}} = \frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ \frac{-(k_1+\Delta k_1 x_1)x_1+k_2(x_2-x_1)+u}{m} \\ \frac{k_2(x_1-x_2)+w}{m} \end{bmatrix}, \quad \dot{\boldsymbol{\theta}} = \frac{d}{dt}\begin{bmatrix} k_1 \\ \Delta k_1 \\ k_2 \\ m \end{bmatrix} = \mathbf{0}_{4\times 1} \tag{3.41}$$

Two sensor setups are considered with the measurements taken denoted as $\boldsymbol{y_1}$ and $\boldsymbol{y_2}$ respectively:

$$\begin{aligned} \boldsymbol{y_1} &= \begin{bmatrix} x_1 \\ \dot{v}_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ \frac{k_2(x_1-x_2)+w}{m} \end{bmatrix} \\ \boldsymbol{y_2} &= \begin{bmatrix} \dot{v}_1 \\ \dot{v}_2 \end{bmatrix} = \begin{bmatrix} \frac{-(k_1+\Delta k_1 x_1)x_1+k_2(x_2-x_1)+u}{m} \\ \frac{k_2(x_1-x_2)+w}{m} \end{bmatrix} \end{aligned} \tag{3.42}$$

where $\boldsymbol{y_1}$ consists of the displacement of the first mass and the acceleration of the second, while in $\boldsymbol{y_2}$, the displacement is replaced by the acceleration of the first mass.

Using the measurement of the first sensor setup $\boldsymbol{y_1}$, Algorithm 3.3 terminates at $k = 7$ reporting that the dynamical system is fully observable. The observability result is in agreement with those reported from the symbolic Algorithm 3.1 and the EORC-DF algorithm presented in [67]. It can hence be concluded that it is theoretically possible to track x_1, x_2, v_1 and v_2 over time and in the meantime identify the values of k_1, Δk_1, k_2 and m based on the $u - \boldsymbol{y_1}$ measurements. With respect to the measurement of the second sensor setup $\boldsymbol{y_2}$, however, the system is not detected observable. To analyse the (k-row) observability properties of $\boldsymbol{x}$, $\boldsymbol{\theta}$ and u and their convergence, Algorithm 3.2 is used to calculate the observability matrix $d\boldsymbol{\Omega^k}$ with a choice of $k = 15$. The resulting $d\boldsymbol{\Omega^{15}}$ then allows for obtaining $d\boldsymbol{\Omega^0}$, ..., $d\boldsymbol{\Omega^{14}}$ by making use of its submatrices. For example, one may obtain $d\boldsymbol{\Omega^{14}}$ from $d\boldsymbol{\Omega^{15}}$ by removing the column corresponding to $w^{(15)}$ and the rows corresponding to $\boldsymbol{L_f^{15}h}$. Figure 3.2 plots the ranks of $d\boldsymbol{\Omega^k}$ in (a) and the k-row observability properties de-

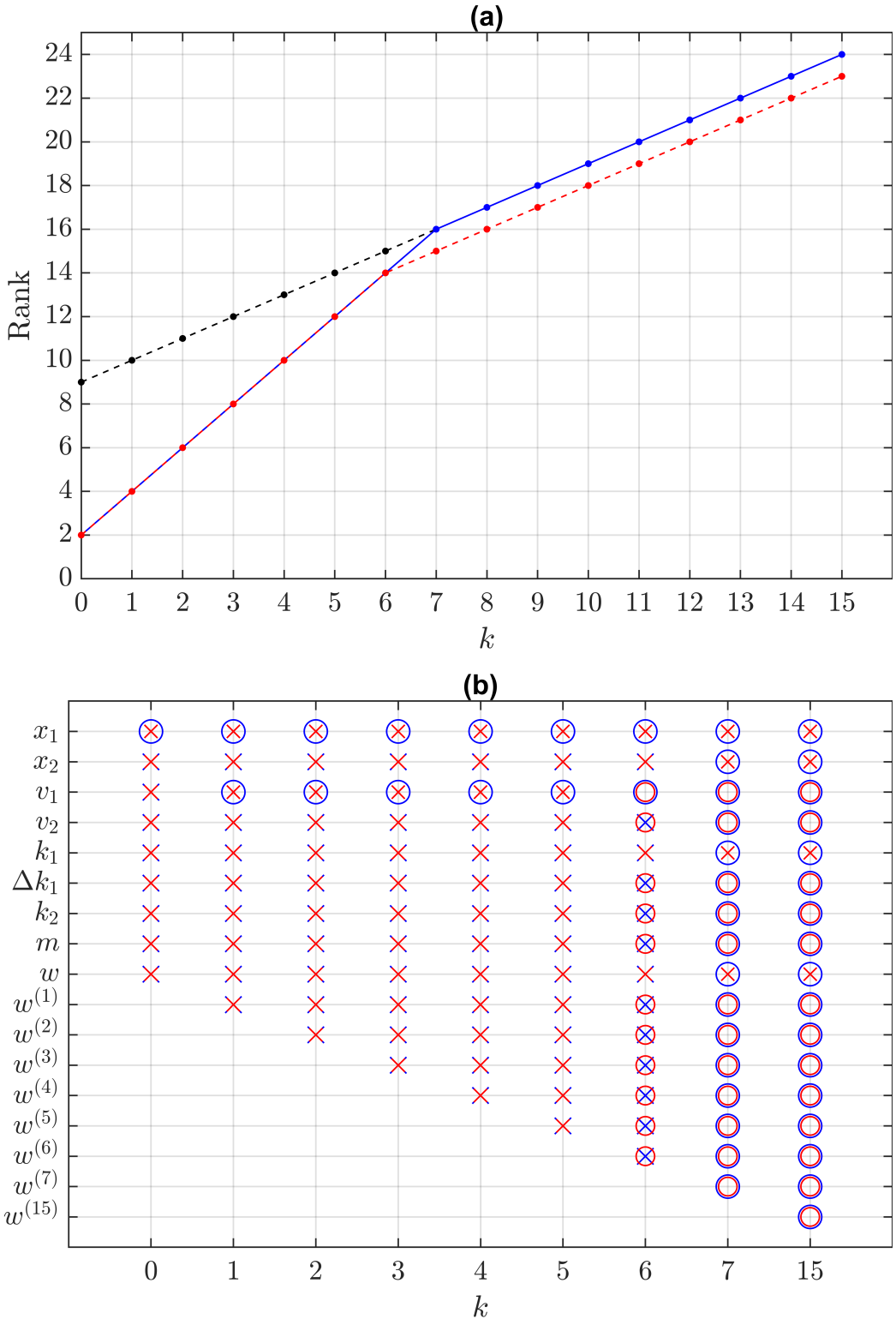

Figure 3.2: (a) The ranks of $d\boldsymbol{\Omega}^{\boldsymbol{k}}$ for the case of $\boldsymbol{y_1}$ measurement (blue line) and the case of $\boldsymbol{y_2}$ measurement (red-dashed line), with comparison to the rank condition $n + l + (k + 1)r$ (black-dashed line) (b) the observable quantities (circles) and k-row unobservable quantities (x-marks) for the case of $\boldsymbol{y_1}$ measurement (blue) and the case of $\boldsymbol{y_2}$ measurement (red)

tected from $d\boldsymbol{\Omega}^{\boldsymbol{k}}$ in (b) for $k = 0, ..., 15$. As can be seen for the measurement $\boldsymbol{y_1}$, $rank(d\boldsymbol{\Omega}^{\boldsymbol{k}}) = n + l + (k+1)r$ occurs and all the $\boldsymbol{x}$, $\boldsymbol{\theta}$ and u correspondingly become observable at $k = 7$. In the case of measuring $\boldsymbol{y_2}$, v_1, v_2, Δk_1, k_2 and m become observable at $k = 6$, but x_1, x_2, k_1 and w remain k-row unobservable throughout all the k considered. From $k = 6$ to 15, $rank(d\boldsymbol{\Omega}^{\boldsymbol{k}})$ (red-dashed line) and the rank condition $n + l + (k+1)r$ (black-dashed line) increase at the same rate, leading to two parallel lines. The occurrence of the parallel lines is in indication that it is highly likely that x_1, x_2, k_1 and w would remain k-row unobservable for larger $k \to \infty$, which implies that these quantities are in fact unobservable. An advantage of using the proposed algorithm is that $d\boldsymbol{\Omega}^{\boldsymbol{k}}$ of a larger k (e.g. $k = 100$) can be calculated to further investigate the ranks and the k-row observability properties (for e.g. $k = 16$ to 100), such that the observability of those k-row unobservable quantities can be concluded with more sufficient confidence.

The afore-described graphical method for determining unobservable quantities with a high level of confidence is suggested in [67]. This example illustrates an alternative way to conclude that x_1, x_2, k_1 and w are indeed unobservable with respect to the measurement $\boldsymbol{y_2}$. Assuming $w^{(k+1)}$ is known, the corresponding augmented form (3.4) of the considered system can be treated as a system with fully known or measured inputs. For such a system, Algorithm 3.3 can be used as the regular ORC algorithm to examine its observability. When the algorithm operates up to a convergent point where the number of the Lie derivatives calculated equals to the number of quantities to be identified, the k-row unobservable quantities detected can be immediately concluded to be unobservable (see [26, 44, 86] for the details of the ORC). Figure 3.3 summarizes the results of observability with assuming $w^{(k+1)}$ known

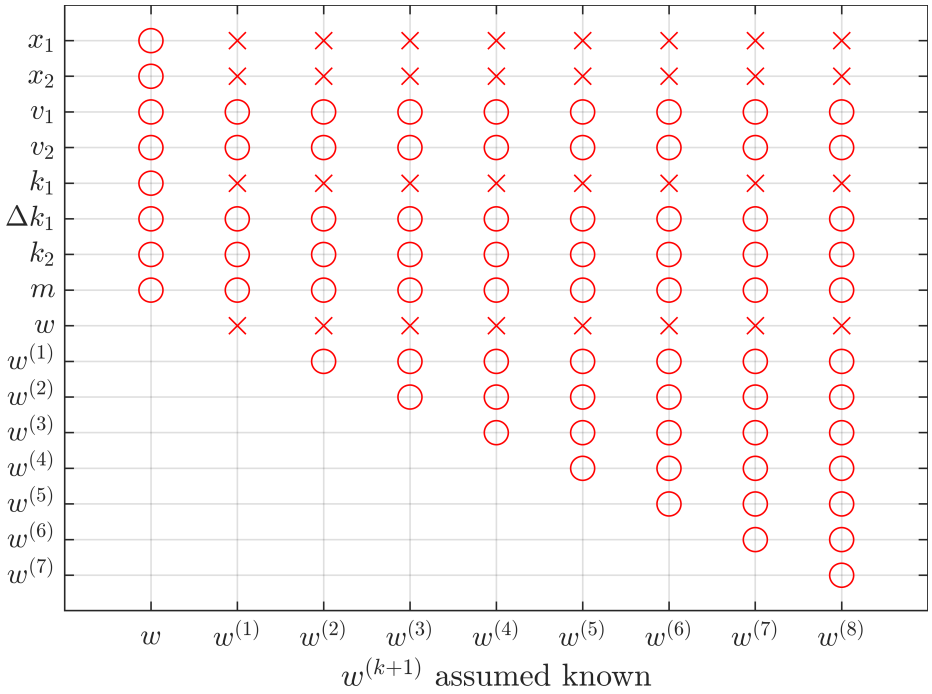

Figure 3.3: The observability of the augmented form of the system with $w^{(k+1)}$ assumed known for the case of $\boldsymbol{y_2}$ measurement

for $k+1 = 0, ..., 8$. As shown in the figure, x_1, x_2, k_1 and w are detected unobservable at $k+1 = 1$ and remain unobservable further. If a quantity is unobservable for a known $w^{(1)}$, it is guaranteed to be unobservable for an unknown $w^{(1)}$ which is the case of this example. Therefore it can be concluded that x_1, x_2, k_1 and w are certainly unobservable, while the remaining quantities are observable as detected in Figure 3.2. It is not possible to properly estimate x_1, x_2, k_1 and w based on the $u - \boldsymbol{y_1}$ measurements using any system identification method.

3.4.2 A 101 DOFs nonlinear system with a tuned mass damper

The proposed algorithm is now applied to test the observability of a large dynamical system. Increasing the number of masses of the system in Example 3.4.1 to 101, the occurring 101 DOFs nonlinear system can be viewed as the model of a 101-storey shear building. In this example, the effects of damping on the dynamics of the

shear building are taken into consideration. As shown in Figure 3.4(a), linear viscous dampers are connected successively between the floors of the model; a pendulum tuned mass damper (TMD), with its mass, stiffness and damping parameters m_d, k_d and c_d respectively, is suspended from the top floor. The modelling of this example is inspired by the design of the well-known Taipei 101 skyscraper in Taiwan. It is further assumed that unmeasured wind load with linearly increased magnitude is applied along the building. The maximum magnitude of the wind, w, appears at the level of top floor.

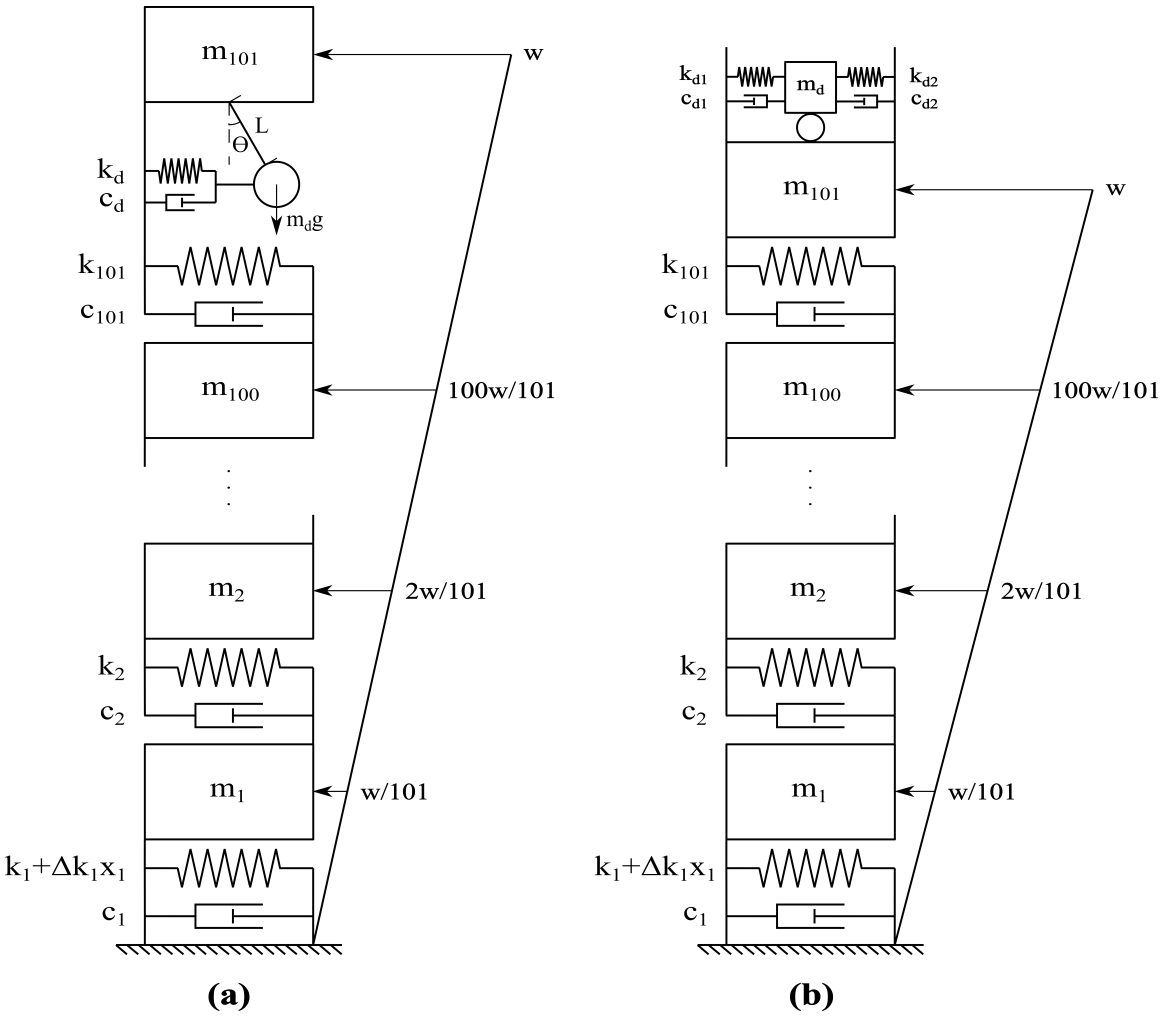

Figure 3.4: (a) A 101-storey shear building with a pendulum TMD and (b) the same shear building with a translational TMD

Let x_i and v_i denote the displacement and velocity of the mass m_i respectively. θ and v_θ denote respectively the sway angle and angular velocity of the pendulum TMD. The dynamic states of the considered system are therefore:

$$\boldsymbol{x} = [x_1, ..., x_{101}, \theta, v_1, ..., v_{101}, v_\theta]^T \tag{3.43}$$

The unknown parameters to be identified are:

$$\boldsymbol{\theta} = [k_1, ..., k_{101}, \Delta k_1, c_1, ..., c_{101}, m_d, k_d, c_d]^T \tag{3.44}$$

where k_i and c_i are respectively the coefficients of the linear spring and damper between m_{i-1} and m_i. The stiffness of the nonlinear spring at the bottom is again given by $k_1 + \Delta k_1 x_1$. The equations used to describe the dynamics of the pendulum TMD are given by:

$$\begin{aligned}
\dot{\theta} =& v_\theta \\
\dot{v}_\theta =& \frac{k_{101}(x_{101} - x_{100})}{m_{101}L} + \frac{c_{101}(v_{101} - v_{100})}{m_{101}L} - \frac{m_{101} + m_d}{m_{101}m_d}(k_d + \frac{m_d g}{L})\theta \\
& - \frac{m_{101} + m_d}{m_{101}m_d} c_d v_\theta - \frac{w}{m_{101}L}
\end{aligned} \tag{3.45}$$

The following equations describe the dynamics of m_{101}:

$$\begin{aligned}
\dot{x}_{101} =& v_{101} \\
\dot{v}_{101} =& -\frac{k_{101}(x_{101} - x_{100})}{m_{101}} - \frac{c_{101}(v_{101} - v_{100})}{m_{101}} + \frac{(k_d L + m_d g)\theta}{m_{101}} \\
& + \frac{c_d L v_\theta}{m_{101}} + \frac{w}{m_{101}}
\end{aligned} \tag{3.46}$$

while the state-space equations of the remaining parts of the system can be written referring to the standard modelling of shear buildings [30]. For the purpose of comparison, the same shear building with a translational TMD, as shown in Figure 3.4(b), is also tested. In this case, the displacement and velocity of the TMD are denoted by x_d and v_d respectively, and the dynamical equations of x_d, v_d, x_{101} and

v_{101} are written as:

$$\begin{aligned}
\dot{x}_d =& v_d \\
\dot{v}_d =& -\frac{k_d(x_d - x_{101})}{m_d} - \frac{c_d(v_d - v_{101})}{m_d} \\
\dot{x}_{101} =& v_{101} \\
\dot{v}_{101} =& -\frac{k_{101}(x_{101} - x_{100})}{m_{101}} - \frac{c_{101}(v_{101} - v_{100})}{m_{101}} + \frac{(k_{d1} + k_{d2})(x_d - x_{101})}{m_{101}} \\
&+ \frac{(c_{d1} + c_{d2})(v_d - v_{101})}{m_{101}} + \frac{w}{m_{101}}
\end{aligned} \tag{3.47}$$

where k_{d1}, k_{d2}, c_{d1} and c_{d2} are the unknown TMD stiffness and damping parameters to be identified.

For both the cases of pendulum TMD and translational TMD, ten sensors are installed at every ten floors along the building to measure 5 displacements and 5 accelerations. Specifically, the measured outputs are:

$$\boldsymbol{y} = [\dot{v}_{101}, x_{91}, \dot{v}_{81}, x_{71}, \dot{v}_{61}, x_{51}, \dot{v}_{41}, x_{31}, \dot{v}_{21}, x_{11}]^T \tag{3.48}$$

Algorithm 3.3 concludes that the shear building with the pendulum TMD is fully observable. All of its dynamic states, unknown parameters and the unmeasured wind load are observable based on the $\boldsymbol{y}$ measurement. The shear building with the translational TMD is however detected unobservable. An analysis of the observability matrix $d\boldsymbol{\Omega}^{\mathbf{511}}$ obtained from Algorithm 3.2 shows that the parameters k_{d1}, k_{d2}, c_{d1} and c_{d2} are not individually identifiable, while the remaining dynamic states, parameters and unmeasured input of the system are detected observable.

It should be noted that the symbolic Algorithm 3.1 and the EORC-DF algorithm presented in [67] are not capable of handling the large shear building systems tested in this example due to their limitation of high memory requirements. In comparison, the proposed algorithm is efficient in the usage of physical memory and is implemented

relatively fast to test these large systems which contains more than a total of 410 dynamic states and unknown parameters. The efficiency of the algorithm could be further improved through implementing in a high-performance computer and making use of more computer cores. This allows for applications of the proposed algorithm to systems of large size and high complexity that are often encountered in engineering practice.

3.4.3 A 2D finite element (FE) model of wind turbine

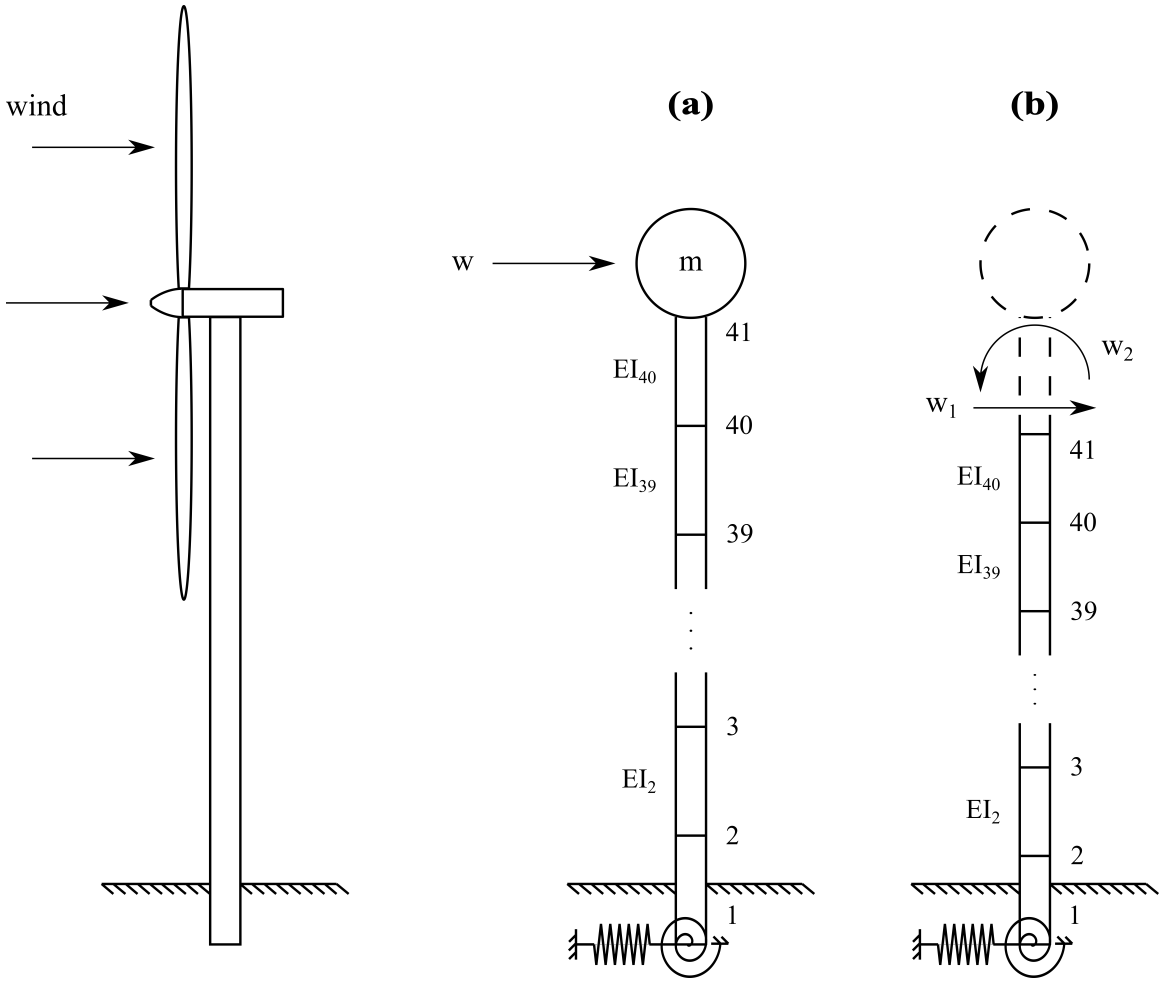

Figure 3.5: (a) A 2D FE model of wind turbine and (b) a substructural model of the wind turbine

In this example, the observability of a wind turbine model as shown in Figure

3.5 is examined using the proposed algorithm. Figure 5(a) demonstrates a simplified 2D FE model of the wind turbine, where the rotor blades and nacelle are lumped at the top of the tower as a mass. The lumped mass m is assumed to be known, and its moment of inertia, denoted by I_m, is an unknown parameter to be identified. The mass is subject to a thrust force w which is unmeasured. It is assumed that the torsional stiffness of the wind turbine is infinitely large and the torsional vibration is not taken into consideration in the modelling. The tower is discretized into 40 uniform Euler-Bernoulli beam elements. The stiffness and mass matrices of the i^{th} element between the nodes i and $i+1$ are given by:

$$\begin{aligned} \boldsymbol{K_e} &= \frac{EI_i}{L^3}\begin{bmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12 & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{bmatrix} \\ \boldsymbol{M_e} &= \frac{\rho A_i L}{420}\begin{bmatrix} 156 & 22L & 54 & -13L \\ 22L & 4L^2 & 13L & -3L^2 \\ 54 & 13L & 156 & -22L \\ -13L & -3L^2 & -22L & 4L^2 \end{bmatrix} \end{aligned} \tag{3.49}$$

where EI_i is the unknown stiffness parameter of the element, and ρ, A_i and L are the known density, cross-section area and length respectively. The foundation of the wind turbine is modelled as the following macro-element:

$$\begin{bmatrix} F_{spr} \\ M_{spr} \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} \\ k_{12} & k_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ \theta_1 \end{bmatrix} \tag{3.50}$$

where F_{spr} and M_{spr} are the resistant force and moment of the springs respectively, and x_1 and θ_1 are respectively the displacement and rotation of node 1. The stiffness matrix is assumed to be generally symmetric with the unknown coefficients k_{11}, k_{12} and k_{22} to be identified.

Let x_i and θ_i denote the displacement and rotation of node i respectively, and v_i and $v_{\theta i}$ the translational and rotational velocities respectively. The dynamic state

vector of the model is a collection of all the nodal displacements, rotations and the corresponding velocities:

$$\boldsymbol{x} = [x_1, \theta_1, ..., x_{41}, \theta_{41}, v_1, v_{\theta 1}, ..., v_{41}, v_{\theta 41}]^T \tag{3.51}$$

The unknown parameter vector is:

$$\boldsymbol{\theta} = [I_m, EI_1, ..., EI_{40}, k_{11}, k_{12}, k_{22}]^T \tag{3.52}$$

The state-space equations of the model are written in the following form:

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}w, \;\; \dot{\boldsymbol{\theta}} = \boldsymbol{0} \tag{3.53}$$

where

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{0}_{82\times 82} & \boldsymbol{I}_{82\times 82} \\ -\boldsymbol{M}^{-1}\boldsymbol{K} & \boldsymbol{0}_{82\times 82} \end{bmatrix}, \;\; \boldsymbol{B} = \begin{bmatrix} \boldsymbol{0}_{82\times 1} \\ \boldsymbol{M}^{-1} \begin{bmatrix} \boldsymbol{0}_{80\times 1} \\ 1 \\ 0 \end{bmatrix} \end{bmatrix} \tag{3.54}$$

where $\boldsymbol{K}$ and $\boldsymbol{M}$ are respectively the model stiffness and mass matrices constructed from the assembly [30] of the element stiffness and mass matrices of the lumped mass, the beam elements of the tower and the springs of the foundation.

Figure 3.5(b) shows a substructural model of the wind turbine which consists of a lower part of the tower and the foundation. A shear force w_1 and a bending moment w_2, which are unmeasured, are applied at node 41 to account for the dynamic effects of the upper-structure (the lumped mass and the upper part of the tower) on the model. This model is used to investigate whether it is theoretically possible to identify the substructure of interest using only measurements obtained from it. The feasibility of substructural identification would lead to significant reduction of identification problem size and more efficient usage of available sensors, and hence more accurate identification results. In this case, the state-space equations of the substructural

model are written as:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B_1} w_1 + \boldsymbol{B_2} w_2 \\ \dot{\boldsymbol{\theta}} &= \frac{d}{dt}\left[EI_1, ..., EI_{40}, k_{11}, k_{12}, k_{22}\right]^T = \mathbf{0} \end{aligned} \tag{3.55}$$

where

$$\boldsymbol{B_1} = \begin{bmatrix} \mathbf{0}_{82\times 1} \\ \boldsymbol{M}^{-1}\begin{bmatrix} \mathbf{0}_{80\times 1} \\ 1 \\ 0 \end{bmatrix} \end{bmatrix}, \quad \boldsymbol{B_2} = \begin{bmatrix} \mathbf{0}_{82\times 1} \\ \boldsymbol{M}^{-1}\begin{bmatrix} \mathbf{0}_{80\times 1} \\ 0 \\ 1 \end{bmatrix} \end{bmatrix} \tag{3.56}$$

For both the models in Figure 3.5(a) and Figure 3.5(b), 4 sensors are installed at every 10 nodes along the tower. 4 sensor setups are examined with the output measurements:

$$\begin{aligned} \boldsymbol{y_1} &= [\dot{v}_{41}, \dot{v}_{31}, \dot{v}_{21}, \dot{v}_{11}], \quad \boldsymbol{y_2} = [\theta_{41}, \dot{v}_{31}, \dot{v}_{21}, \dot{v}_{11}] \\ \boldsymbol{y_3} &= [\theta_{41}, \theta_{31}, \dot{v}_{21}, \dot{v}_{11}], \quad \boldsymbol{y_4} = [\theta_{40}, \theta_{31}, \dot{v}_{21}, \dot{v}_{11}] \end{aligned} \tag{3.57}$$

In the first setup $\boldsymbol{y_1}$, all the sensors are accelerometers measuring lateral accelerations, while in the second setup $\boldsymbol{y_2}$ an accelerometer at the top of the tower is replaced by an inclinometer. 2 inclinometers are used in $\boldsymbol{y_3}$. In $\boldsymbol{y_4}$ a sensor is moved from the top to the node below.

The observability results obtained from the proposed algorithm are summarized in Table 3.1. When all the 4 sensors are accelerometers as in the setup $\boldsymbol{y_1}$, both the full and substructural models are detected unobservable: neither their dynamic states nor the unmeasured forces applied are observable, but all the unknown parameters are identifiable. If inclinometers are used as in the setups $\boldsymbol{y_2}$ and $\boldsymbol{y_3}$, it then becomes possible to track those dynamic states and unmeasured forces given the measurements; the more unmeasured forces exist the more inclinometers are needed to achieve so. Using the measurement $\boldsymbol{y_4}$ for the substructural model, 4 dynamic states, 1 parameter and the 2 unmeasured forces are found highly likely unobservable by means of the graphical method described in Example 3.4.1. A comparison between the setups $\boldsymbol{y_3}$

Measurements		$\boldsymbol{y_1}$	$\boldsymbol{y_2}$	$\boldsymbol{y_3}$	$\boldsymbol{y_4}$
(a)	Observable	$\boldsymbol{\theta}$	$\boldsymbol{x}$, $\boldsymbol{\theta}$ and w	$\boldsymbol{x}$, $\boldsymbol{\theta}$ and w	$\boldsymbol{x}$, $\boldsymbol{\theta}$ and w
	Unobservable	$\boldsymbol{x}$ and w	-	-	-
(b)	Observable	$\boldsymbol{\theta}$	$\boldsymbol{\theta}$, θ_{41} and $v_{\theta 41}$	$\boldsymbol{x}$, $\boldsymbol{\theta}$, w_1 and w_2	the remaining
	Unobservable	$\boldsymbol{x}$, w_1 and w_2	the remaining	-	-
	Highly likely unobservable	-	-	-	EI_{40}, x_{41}, v_{41}, θ_{41} $v_{\theta 41}$, w_1 and w_2

Table 3.1: The observability of the full model in Figure 3.5(a) and the substructural model in Figure 3.5(b) with respect to the measurement scenarios $\boldsymbol{y_1}$, $\boldsymbol{y_2}$, $\boldsymbol{y_3}$ and $\boldsymbol{y_4}$

and $\boldsymbol{y_4}$ implies that, a sensor should be considered to be placed at where the structure is 'split' and the unmeasured forces are applied i.e. node 41 in this example, in order to achieve feasible substructural identification.

3.4.4 A 2D FE model of a truss-beam bridge

Application of the proposed algorithm to a 2D FE model of a truss-beam bridge whose geometry is inspired by the design of Tokyo Gate Bridge [112] is demonstrated in this example. The bridge is a three-span composite structure of steel bars in the arrangements of trusses and a steel box girder as shown in Figure 3.6(a). The investigated model is a substructure of the bridge, as shown in Figure 3.6(b), which consists of the left 320m part of the girder, including the left span and half of the central span, and the associated trusses. An unmeasured shear force w_1 and an unmeasured bending moment w_2 are applied at the right end of the girder to account for the dynamic effects of the remaining bridge structure on the model. A measured excitation u is applied vertically at the marked position on the deck, which in practice could be provided by a hammer or an actuator.

The substructure of the bridge is modelled using linear frame-type macro-elements: the steel girder is discretized into 20 uniform Euler-Bernoulli beam elements, and the other pin-jointed members are treated as truss elements. Each of the truss nodes

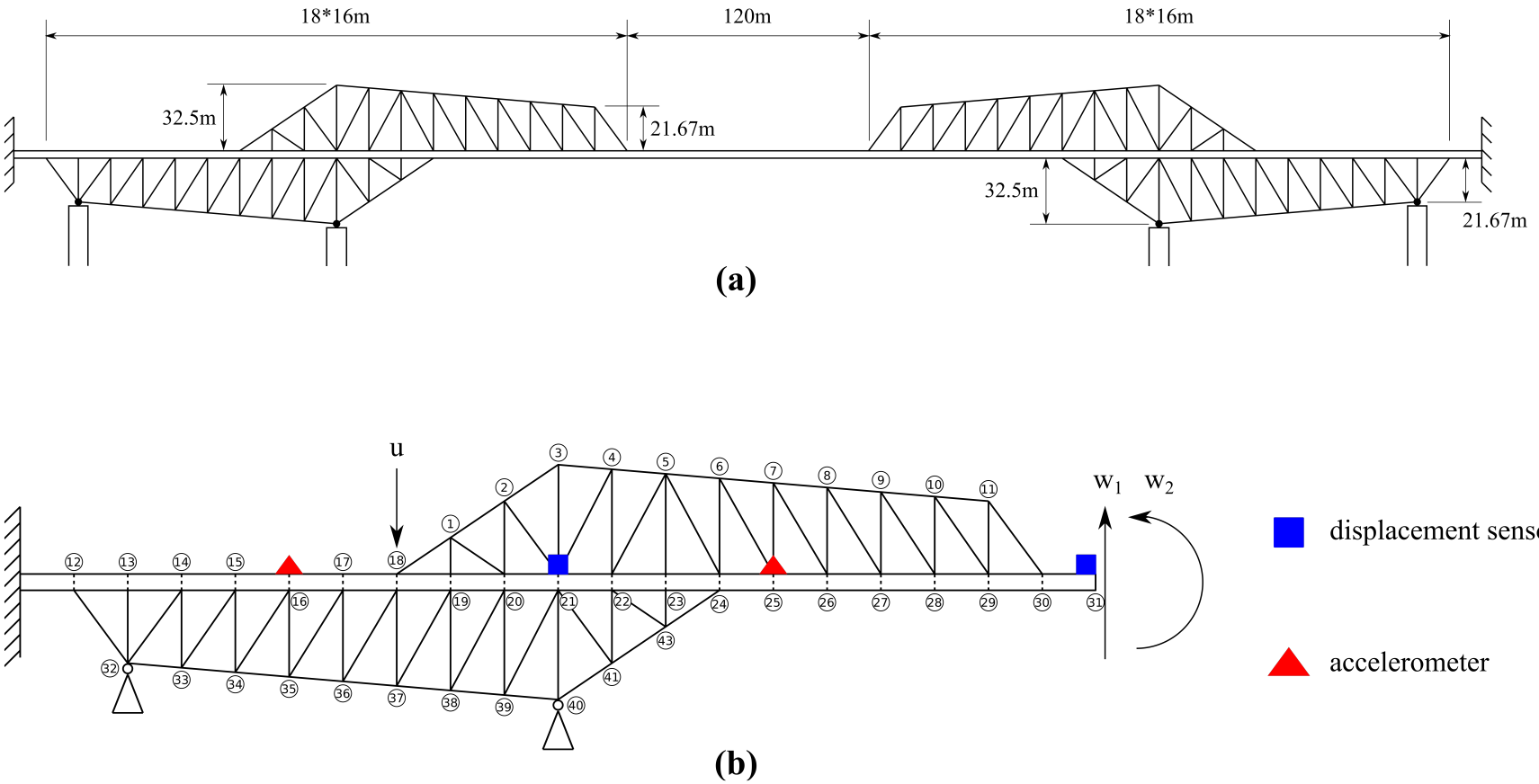

Figure 3.6: (a) The geometry of the truss-beam bridge (b) the investigated substructural model of the bridge with a given sensor setup

1-11 and 32-43 has 2 DOFs i.e. translational motions along horizontal and vertical axes. The axial stiffness of the truss element between nodes i and j is k_{ij}, which is an unknown parameter to be identified. The mass matrix of each truss element is obtained by assuming the element mass is lumped as point masses along the translational DOFs at the ends. The point mass from all the jointed members lumped at each truss node is assumed to be known. Each of the beam nodes 12-31 has 2 DOFs i.e. vertical motion and rotation with neglect of axial motion. The stiffness and mass matrices of the beam element between nodes i and j are given in equation (3.49), where the corresponding stiffness parameter EI_{ij} is unknown. At the nodes of contact between the truss elements and girder, a rigid element transformation relates the truss nodal DOFs to the beam nodal DOFs. The left end of the girder is assumed to be fixed and the bottom truss is connected with pin-joints to the piers, as shown in the figure.

The state-space equations of the model are written in the following form:

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B_1}w_1 + \boldsymbol{B_2}w_2 + \boldsymbol{B_3}u, \;\; \dot{\boldsymbol{\theta}} = \boldsymbol{0} \tag{3.58}$$

where the dynamic state vector $\boldsymbol{x}$ contains all the nodal displacements, rotations and the corresponding velocities. The parameter vector $\boldsymbol{\theta}$ includes all the unknown stiffness parameters of the truss and beam elements. The form of the system matrix $\boldsymbol{A}$ is given in equation (3.54), where $\boldsymbol{K}$ and $\boldsymbol{M}$ are respectively the model stiffness and mass matrices constructed from the assembly of all the element stiffness and mass matrices of the truss and beam elements. $\boldsymbol{B_1}$, $\boldsymbol{B_2}$ and $\boldsymbol{B_3}$ are the affine-input vectors for the measured and unmeasured inputs to the model. 4 sensors are installed on the deck at the positions shown in Figure 3.6(b), where 2 vertical displacements and 2 vertical accelerations are measured to record the response of the bridge to the applied

excitation u.

The proposed algorithm reports that the substructural model of the bridge is fully observable in the presence of unmeasured inputs. Based on the measurements obtained from the given sensor setup, it is theoretically feasible to estimate the unknown properties of the substructure and in the meantime track its dynamics using some suitably chosen system identification methods. Similar as in Example 3.4.3, the result of this example suggests the theoretical possibility of an alternative strategy that can be used to identify or perform health monitoring on a large structure (e.g. as for the bridge), especially when there is a lack of sensors. A substructure of the large structure, whose properties and dynamics are of interest, could be properly selected, modelled and identified while neglecting the remaining elements of the structure which are not of interest. Such strategy of substructural identification would help in tackling problems related to dimensionality and in using the available sensors more efficiently, leading to an improvement of identification accuracy. In the cases of substructural identification and many other realistic scenarios, the observability algorithm proposed in this work would allow researchers to freely investigate the viability of various structural models of interest and sensor patterns without concerning about the computational constrains of the original observability method.

3.5 Conclusions

In this chapter, an efficient algorithm is developed to examine the observability and identifiability of rational nonlinear systems with unmeasured inputs. The underlying theory behind the algorithm is based on the extension of the extended Observability Rank Condition (EORC-DF) to relax the requirement of systems being affine in

inputs. A power series-based computational framework of the observability matrix is established by making use of a variational system of differential equations. In the framework, expensive symbolic computations of Lie derivatives are replaced by cheaper calculations between power series with numerical coefficients. The use of Newton's iteration allows for obtaining a large number of Lie derivatives within a small number of iterations, without the need to perform symbolic differentiations or gradients. Additionally, the introduction of modular operations further improves the efficiency of the algorithm through controlling the growth of numbers. A high-rise shear building, the 2D FE model of a wind turbine and the 2D FE model of a large bridge are successfully tested as examples to demonstrate the usefulness, robustness and efficiency of the proposed algorithm. The obtained observability results of the wind turbine and bridge models also suggest the possibility of performing substructural identification in order for more efficient system identification or structural health monitoring.

Chapter 4

Lie Symmetries of Nonlinear Systems with Unmeasured Inputs

The main objective of this chapter is to propose two computational methods for finding Lie symmetries of nonlinear dynamical systems with unmeasured inputs. The developments are based on extensions of the existing computational methods suggested in [102, 3, 86] for nonlinear systems with fully measured inputs and outputs. Each of the proposed methods has its own advantages and disadvantages in terms of applicability and computation efficiency. The resulting Lie symmetries found from the methods can provide an alternative standpoint to understand the observability and identifiability properties of dynamical systems. They are able to theoretically predict what results could be obtained when identifying a dynamical system with a given setup of sensors, what mathematical relationship would be between the identification results and true solutions, whether the system states and unmeasured inputs could be under-estimated, over-estimated or correctly estimated, etc. In addition, the chapter performs pioneer investigations on the utilization of Lie symmetries for model reduction, sensor re-placement, etc. in order to improve system identification results.

4.1 Lie symmetries of nonlinear systems with fully measured inputs

4.1.1 Lie transformations

This section first briefly sketches some basic elements of Lie symmetry analysis of differential equations; a more detailed exposition can be found in [76, 79, 12, 13].

Let $\boldsymbol{x} = [x_1, x_2, ..., x_n]^T$ denote a vector of variables in a domain $D \subset \mathbb{R}^n$, and consider a subset $S \subseteq \mathbb{R}$. The set of transformations

$$\boldsymbol{\phi}(\boldsymbol{x}, \epsilon) : D \times S \to D \tag{4.1}$$

depending on the parameter ϵ, forms a one-parameter group of transformations on D if:

1. For each value of the parameter $\epsilon \in S$ the transformations are one-to-one onto D;

2. S with the law of composition μ is a group with identity zero;

3. $\boldsymbol{\phi}(\boldsymbol{x}, 0) = \boldsymbol{x}, \ \ \forall \boldsymbol{x} \in D$;

4. $\boldsymbol{\phi}(\boldsymbol{\phi}(\boldsymbol{x}, \epsilon_1), \epsilon_2) = \boldsymbol{\phi}(\boldsymbol{x}, \mu(\epsilon_1, \epsilon_2)), \ \ \forall \boldsymbol{x} \in D, \ \ \forall \epsilon_1, \epsilon_2 \in S$;

This group of transformations defines a one-parameter Lie group of transformations if in addition to satisfying the axioms of the previous definition:

5. ϵ is a continuous parameter, i.e. S is an interval in $\mathbb{R}$;

6. $\boldsymbol{\phi}$ is C^∞ with respect to $\boldsymbol{x}$ in D and an analytic function of ϵ in S;

7. $\mu(\epsilon_1, \epsilon_2)$ is an analytic function of ϵ_1 and ϵ_2, $\ \ \forall \epsilon_1, \epsilon_2 \in S$.

It is indicated by the Lie's First Fundamental Theorem [11, 48, 47, 77] that the Lie group of transformation is equivalent to the solution of the initial value problem

for the system of first-order differential equations:

$$\frac{\partial \boldsymbol{\phi}}{\partial \epsilon} = \boldsymbol{\xi}(\boldsymbol{\phi}), \;\; \boldsymbol{\phi}(\boldsymbol{x}, 0) = \boldsymbol{x} \tag{4.2}$$

where $\boldsymbol{\xi}$ is an analytic tangent vector field, and $\boldsymbol{\xi}(\boldsymbol{x})$ is defined as the infinitesimals of the group of transformations. Lie's First Fundamental Theorem ensures that the infinitesimals contain the essential information for characterizing a one-parameter Lie group of transformations.

4.1.2 Lie symmetries

A one-parameter Lie group of transformations admitted by a set of differential equations is called a one-parameter group of Lie (point) symmetries of the differential equations. Consider nonlinear dynamical systems that are written in the following state-space representation:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) \\ \boldsymbol{y} &= \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}) \end{aligned} \tag{4.3}$$

where $\boldsymbol{x} = [x_1, x_2, ..., x_n]^T$ denotes the augmented state vector [26] containing both the dynamic states and the time-invariant parameters of the underlying system, $\boldsymbol{u} \in \mathbb{R}^l$ the vector of measured and hence known inputs, and $\boldsymbol{y} \in \mathbb{R}^p$ the vector of output measurements. A one-parameter group of Lie symmetries of the system described in equation (4.3) is given by:

$$\boldsymbol{\phi}(\boldsymbol{x}, \epsilon) = [\phi_1, \phi_2, ..., \phi_n]^T \tag{4.4}$$

where the $i^{th} (1 \leq i \leq n)$ component of $\boldsymbol{\phi}(\boldsymbol{x}, \epsilon)$, i.e. ϕ_i, is a Lie symmetry of the i^{th} component of $\boldsymbol{x}$, i.e. x_i, and it is in general an analytic function of $\boldsymbol{x}$ and the

real constant parameter ϵ. A fundamental property of Lie symmetries is that such a group of transformations of $\boldsymbol{x}$ and ϵ satisfies the state-space and output equations of the considered system, leaving the measured inputs and outputs invariant, i.e.

$$\begin{aligned} \dot{\boldsymbol{\phi}} &= \boldsymbol{f}(\boldsymbol{\phi}, \boldsymbol{u}) \\ \boldsymbol{y} &= \boldsymbol{h}(\boldsymbol{\phi}, \boldsymbol{u}) \end{aligned} \tag{4.5}$$

where the vectors of functions $\boldsymbol{f}$ and $\boldsymbol{h}$ between equation (4.3) and equation (4.5) remain the same.

To illustrate the property of Lie symmetries, the following simple dynamical system [86] is used as an example:

$$\begin{aligned} \dot{x}_1 &= \theta_1 x_1^2 + \theta_2 x_1 x_2 + u \\ \dot{x}_2 &= \theta_3 x_1^2 + \theta_4 x_1 x_2 \\ \dot{\theta}_1 &= 0, \;\; \dot{\theta}_2 = 0, \;\; \dot{\theta}_3 = 0, \;\; \dot{\theta}_4 = 0 \\ y &= x_1 \end{aligned} \tag{4.6}$$

The state vector of the system is $\boldsymbol{x} = [x_1, x_2, \theta_1, \theta_2, \theta_3, \theta_4]^T$, and the one-parameter group of Lie symmetries is found to be:

$$\boldsymbol{\phi}(\boldsymbol{x}, \epsilon) = [x_1, e^{\epsilon} x_2, \theta_1, \frac{\theta_2}{e^{\epsilon}}, e^{\epsilon}\theta_3, \theta_4]^T \tag{4.7}$$

$\boldsymbol{\phi}(\boldsymbol{x}, \epsilon)$ satisfies equation (4.6), that is, if one-to-one substitutions of $\boldsymbol{\phi}$ for $\boldsymbol{x}$ are performed in equation (4.6) the equations still hold. In other words, for the solution of equation (4.6) $\boldsymbol{x}^{\star}$ and each value of ϵ, the group of transformations $\boldsymbol{\phi}(\boldsymbol{x}^{\star}, \epsilon)$ is also a solution of the equations, where it should be noted that $\boldsymbol{\phi}(\boldsymbol{x}^{\star}, \epsilon)$ has included the case of $\boldsymbol{\phi}(\boldsymbol{x}^{\star}, \epsilon) = \boldsymbol{x}^{\star}$ when $\epsilon = 0$. Since ϵ can be any value, if one attempts to identify the system based on $u - y$ measurements, it would not be able to find the

true solution of $\boldsymbol{x}$, but rather one of the infinitely many incorrect solutions implied by $\boldsymbol{\phi}(\boldsymbol{x}, \epsilon)$. According to the definition of observability, the system is said to be unobservable, and more specifically, x_2, θ_2 and θ_3 are unobservable. x_1, θ_1 and θ_4 however are observable because their symmetries, i.e. the corresponding components of $\boldsymbol{\phi}(\boldsymbol{x}, \epsilon)$, are not affected by the parameter ϵ. This brings an implication that a system can be said to be observable if its Lie symmetries $\boldsymbol{\phi}(\boldsymbol{x}, \epsilon) \equiv \boldsymbol{x}$ for all possible ϵ.

4.1.3 Computational methods

Two existing methods can be used to compute the groups of Lie symmetries of a given nonlinear system with fully measured inputs and outputs. This section gives a brief review of the computation procedure of the methods, while their mathematical derivations can found in [86, 2, 102, 3] for details.

The first method was proposed by Sedoglavic [86], which is based on computing the following observability matrix of the system in equation (4.3) symbolically:

$$d\boldsymbol{\Omega} = \frac{\partial \boldsymbol{\Omega}}{\partial \boldsymbol{x}}, \quad \boldsymbol{\Omega} = \begin{bmatrix} \boldsymbol{L_f^0 h} \\ \boldsymbol{L_f^1 h} \\ \vdots \\ \boldsymbol{L_f^n h} \end{bmatrix} \tag{4.8}$$

where $\boldsymbol{L_f^i h}$ refers to the i^{th} order extended Lie derivative which is calculated recursively by:

$$\boldsymbol{L_f^i h} = \frac{\partial \boldsymbol{L_f^{i-1} h}}{\partial \boldsymbol{x}} \boldsymbol{f} + \sum_{j=1}^{i} \frac{\partial \boldsymbol{L_f^{i-1} h}}{\partial \boldsymbol{u}^{(j-1)}} \boldsymbol{u}^{(j)} \tag{4.9}$$

The expressions of the infinitesimals of the Lie transformations can then be determined by calculating the kernel of the observability matrix:

$$\boldsymbol{\xi} = ker(d\boldsymbol{\Omega}) \tag{4.10}$$

Once the functions $\boldsymbol{\xi}$ are given, the differential equation in (4.2) can be solved, if mathematically possible, to obtain the solution of $\boldsymbol{\phi}(\boldsymbol{x}, \epsilon)$.

The second computational method and the corresponding algorithm were developed by Urguplu [102] and Anguelova [3]. The method is implemented based on solving the following set of differential equations:

$$\begin{cases} \frac{\partial \boldsymbol{\xi}(\boldsymbol{x})}{\partial \boldsymbol{x}} \boldsymbol{f} - \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} \boldsymbol{\xi}(\boldsymbol{x}) = \mathbf{0} \\ \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}} \boldsymbol{\xi}(\boldsymbol{x}) = \mathbf{0} \\ \frac{\partial \boldsymbol{\phi}}{\partial \epsilon} = \boldsymbol{\xi}(\boldsymbol{\phi}), \;\; \boldsymbol{\phi}(\boldsymbol{x}, 0) = \boldsymbol{x} \end{cases} \tag{4.11}$$

In order to guarantee an analytic solution form equation (4.11), an assumption must be made on the form of $\boldsymbol{\phi}(\boldsymbol{x}, \epsilon)$ in advance. Typical forms of Lie symmetries include translation, scaling, affine, Mobius, quadratic, etc. Substituting the expression of $\boldsymbol{\phi}(\boldsymbol{x}, \epsilon)$ into equation (4.11), the equations can then be solved analytically to determine the unknown coefficients in $\boldsymbol{\phi}(\boldsymbol{x}, \epsilon)$.

In comparison, the first method is computationally cumbersome mainly due to the symbolic computation of the observability matrix; analytically solving the differential equation in (4.2) with the aim of obtaining general forms of Lie symmetries is often computationally intractable or mathematically impossible. The second method is computationally efficient, but the limitation is associated with its inability to deal with general forms of Lie symmetries. In addition, real structural systems in practice are often subject to loads or excitations which are difficult or too expensive to measure; both of the aforementioned computational methods are not capable of handling the situations where parts or potentially all of the inputs are unmeasured for a nonlinear system.

Using the above computational methods, one may obtain multiple one-parameter groups of Lie symmetries denoted by ${}^1\boldsymbol{\phi}(\boldsymbol{x}, \epsilon_1)$, ${}^2\boldsymbol{\phi}(\boldsymbol{x}, \epsilon_2)$, ... These one-parameter

groups can be combined as a multi-parameter group of Lie symmetries through applying the Lie transformations successively:

$$\boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{\epsilon}) = {}^1\boldsymbol{\phi}({}^2\boldsymbol{\phi}(..., \epsilon_2), \epsilon_1) \tag{4.12}$$

where $\boldsymbol{\epsilon}$ is a set of parameters $\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, ...]$.

4.2 Lie symmetries of nonlinear systems with unmeasured inputs

This work considers nonlinear dynamical systems with unmeasured inputs that can be generally written in the following state-space representation:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w}) \\ \boldsymbol{y} &= \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w}) \end{aligned} \tag{4.13}$$

where $\boldsymbol{x} = [x_1, x_2, ..., x_n]^T$ denotes the augmented state vector containing both the dynamic states and the time-invariant parameters of the underlying system. $\boldsymbol{w} = [w_1, w_2, ..., w_m]^T$ denotes the unmeasured and hence unknown inputs, $\boldsymbol{u} \in \mathbb{R}^l$ the vector of measured inputs and $\boldsymbol{y} \in \mathbb{R}^p$ the vector of output measurements. $\boldsymbol{f}$ and $\boldsymbol{h}$ are vectors of nonlinear analytic functions. Assume the system in equation (4.13) contains r one-parameter groups of Lie symmetries. The $i^{th}(1 \leq i \leq r)$ group of the system is a one-parameter, $\epsilon_i \in \mathbb{R}$, group of transformations:

$$\begin{aligned} {}^i\boldsymbol{\phi}_{\boldsymbol{x}}(\boldsymbol{x}, \boldsymbol{w}, \epsilon_i) &= [{}^i\phi_{x,1}, {}^i\phi_{x,2}, ..., {}^i\phi_{x,n}]^T \\ {}^i\boldsymbol{\phi}_{\boldsymbol{w}}(\boldsymbol{x}, \boldsymbol{w}, \epsilon_i) &= [{}^i\phi_{w,1}, {}^i\phi_{w,2}, ..., {}^i\phi_{w,m}]^T \end{aligned} \tag{4.14}$$

where the $j^{th}(1 \leq j \leq n)$ component of ${}^i\boldsymbol{\phi}_{\boldsymbol{x}}$, i.e. ${}^i\phi_{x,j}$, is a Lie symmetry of the j^{th} component of $\boldsymbol{x}$, i.e. x_j, and it is in general an analytic function of $\boldsymbol{x}$, $\boldsymbol{w}$ and

the real constant parameter ϵ_i. Similarly, ${}^i\phi_{w,j}(1 \leq j \leq m)$ is a Lie symmetry of w_j and it is also an analytic function of $\boldsymbol{x}$, $\boldsymbol{w}$ and ϵ_i. A fundamental property of Lie symmetries is that such a group of transformations of $\boldsymbol{x}$, $\boldsymbol{w}$ and ϵ_i fulfils the equations of the considered system, leaving the measured inputs and the output measurements unchanged, i.e.

$$\begin{aligned} {}^i\dot{\boldsymbol{\phi}}_{\boldsymbol{x}} &= \boldsymbol{f}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}}) \\ \boldsymbol{y} &= \boldsymbol{h}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}}) \end{aligned} \tag{4.15}$$

where it should be noted that the functions $\boldsymbol{f}$ and $\boldsymbol{h}$ between equations (4.13) and (4.15) remain the same. Further based on the Lie's First Fundamental Theorem, the following differential equations hold with known initial conditions of ${}^i\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^i\boldsymbol{\phi}_{\boldsymbol{w}}$:

$$\begin{aligned} \frac{\partial^i \boldsymbol{\phi}_{\boldsymbol{x}}}{\partial \epsilon_i} &= {}^i\boldsymbol{\xi}_{\boldsymbol{x}}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}}), \quad {}^i\boldsymbol{\phi}_{\boldsymbol{x}}(\boldsymbol{x}, \boldsymbol{w}, 0) = \boldsymbol{x} \\ \frac{\partial^i \boldsymbol{\phi}_{\boldsymbol{w}}}{\partial \epsilon_i} &= {}^i\boldsymbol{\xi}_{\boldsymbol{w}}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}}), \quad {}^i\boldsymbol{\phi}_{\boldsymbol{w}}(\boldsymbol{x}, \boldsymbol{w}, 0) = \boldsymbol{w} \end{aligned} \tag{4.16}$$

where ${}^i\boldsymbol{\xi}_{\boldsymbol{x}}$ and ${}^i\boldsymbol{\xi}_{\boldsymbol{w}}$ are vectors of nonlinear analytic functions. $\frac{\partial^i \boldsymbol{\phi}_{\boldsymbol{x}}}{\partial \epsilon_i}\big|_{\epsilon_i=0} = {}^i\boldsymbol{\xi}_{\boldsymbol{x}}(\boldsymbol{x}, \boldsymbol{w})$ and $\frac{\partial^i \boldsymbol{\phi}_{\boldsymbol{w}}}{\partial \epsilon_i}\big|_{\epsilon_i=0} = {}^i\boldsymbol{\xi}_{\boldsymbol{w}}(\boldsymbol{x}, \boldsymbol{w})$ are the infinitesimals of the Lie group of transformations.

4.3 Analytic computations of translation, scaling and Mobius Lie symmetries

In this section, a computational method of Lie symmetries of the system in equation (4.13) is derived by extending the works in [102, 3] to account for the existence of the unmeasured inputs. The symmetry computation relies on setting up a system of differential equations whose solution provides the information of ${}^i\boldsymbol{\xi}_{\boldsymbol{x}}$ and ${}^i\boldsymbol{\xi}_{\boldsymbol{w}}$ appearing in equation (4.16). The first differential equation of the system is derived based on the fact that the state-space equation in equation (4.15) remains invariant for all

possible realizations of the parameter ϵ_i, and therefore:

$$\frac{d^i\dot{\boldsymbol{\phi}}_{\boldsymbol{x}}}{d\epsilon_i} = \frac{d\boldsymbol{f}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}})}{d\epsilon_i} \tag{4.17}$$

Since $\frac{d^i\dot{\boldsymbol{\phi}}_{\boldsymbol{x}}}{d\epsilon_i} = \frac{d^i\boldsymbol{\xi}_{\boldsymbol{x}}}{dt}$, equation (4.17) becomes:

$$\frac{d^i\boldsymbol{\xi}_{\boldsymbol{x}}}{dt} = \frac{d\boldsymbol{f}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}})}{d\epsilon_i} \tag{4.18}$$

Applying the chain rule to equation (4.18) yields:

$$\frac{\partial^i\boldsymbol{\xi}_{\boldsymbol{x}}}{\partial^i\boldsymbol{\phi}_{\boldsymbol{x}}}\boldsymbol{f} + \frac{\partial^i\boldsymbol{\xi}_{\boldsymbol{x}}}{\partial^i\boldsymbol{\phi}_{\boldsymbol{w}}}\frac{d^i\boldsymbol{\phi}_{\boldsymbol{w}}}{dt} = \frac{\partial\boldsymbol{f}}{\partial^i\boldsymbol{\phi}_{\boldsymbol{x}}}{}^i\boldsymbol{\xi}_{\boldsymbol{x}} + \frac{\partial\boldsymbol{f}}{\partial^i\boldsymbol{\phi}_{\boldsymbol{w}}}{}^i\boldsymbol{\xi}_{\boldsymbol{w}} \tag{4.19}$$

Similarly, the second differential equation of the system is derived based on the fact that the measurement equation in equation (4.15) remains invariant for all possible ϵ_i, and therefore:

$$\frac{d\boldsymbol{y}}{d\epsilon_i} = \boldsymbol{0} = \frac{d\boldsymbol{h}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}})}{d\epsilon_i} \tag{4.20}$$

Applying the chain rule to equation (4.20) yields:

$$\frac{\partial\boldsymbol{h}}{\partial^i\boldsymbol{\phi}_{\boldsymbol{x}}}{}^i\boldsymbol{\xi}_{\boldsymbol{x}} + \frac{\partial\boldsymbol{h}}{\partial^i\boldsymbol{\phi}_{\boldsymbol{w}}}{}^i\boldsymbol{\xi}_{\boldsymbol{w}} = \boldsymbol{0} \tag{4.21}$$

The obtained equations (4.19) and (4.21) are then evaluated at $\epsilon_i = 0$, leading to the following system of differential equations:

$$\begin{cases} \frac{\partial^i\boldsymbol{\xi}_{\boldsymbol{x}}(\boldsymbol{x},\boldsymbol{w})}{\partial\boldsymbol{x}}\boldsymbol{f} + \frac{\partial^i\boldsymbol{\xi}_{\boldsymbol{x}}(\boldsymbol{x},\boldsymbol{w})}{\partial\boldsymbol{w}}\frac{d\boldsymbol{w}}{dt} - \frac{\partial\boldsymbol{f}}{\partial\boldsymbol{x}}{}^i\boldsymbol{\xi}_{\boldsymbol{x}}(\boldsymbol{x},\boldsymbol{w}) - \frac{\partial\boldsymbol{f}}{\partial\boldsymbol{w}}{}^i\boldsymbol{\xi}_{\boldsymbol{w}}(\boldsymbol{x},\boldsymbol{w}) = \boldsymbol{0} \\ \frac{\partial\boldsymbol{h}}{\partial\boldsymbol{x}}{}^i\boldsymbol{\xi}_{\boldsymbol{x}}(\boldsymbol{x},\boldsymbol{w}) + \frac{\partial\boldsymbol{h}}{\partial\boldsymbol{w}}{}^i\boldsymbol{\xi}_{\boldsymbol{w}}(\boldsymbol{x},\boldsymbol{w}) = \boldsymbol{0} \end{cases} \tag{4.22}$$

where $\boldsymbol{f}$ and $\boldsymbol{h}$ correspond to the functions in equation (4.13). Analytically solving the system of equations in (4.22) yields ${}^i\boldsymbol{\xi}_{\boldsymbol{x}}(\boldsymbol{x},\boldsymbol{w})$ and ${}^i\boldsymbol{\xi}_{\boldsymbol{w}}(\boldsymbol{x},\boldsymbol{w})$, and the Lie's First Fundamental Theorem ensures ${}^i\boldsymbol{\xi}_{\boldsymbol{x}}(\boldsymbol{x},\boldsymbol{w})$ and ${}^i\boldsymbol{\xi}_{\boldsymbol{w}}(\boldsymbol{x},\boldsymbol{w})$ contain the essential information for characterizing ${}^i\boldsymbol{\xi}_{\boldsymbol{x}}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}})$ and ${}^i\boldsymbol{\xi}_{\boldsymbol{w}}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}})$. Subsequently solving the equations in (4.16) allow for obtaining the group of Lie symmetries ${}^i\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^i\boldsymbol{\phi}_{\boldsymbol{w}}$.

In general, obtaining an analytical solution of the above system of differential equations is challenging if no assumptions are made for the symmetries [102]. However, if ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^{i}\boldsymbol{\phi}_{\boldsymbol{w}}$ are assumed to be certain types of symmetries, such as translation, scaling and Mobius, the system of differential equations can be solved automatically and efficiently. Section 4.3.1, 4.3.2 and 4.3.3 discuss in detail the efficient computations of one-parameter translation, scaling and Mobius types of symmetries for the system described in (4.13). It is further assumed in the following sections that all the symmetries occurring in the system are related to translation, scaling and Mobius symmetries and their combinations. This simplifying assumption is often satisfied for a wide range of real world engineering systems. Other types of Lie transformations not studied in this work include affine, quadratic and some more general higher-order polynomial symmetries as investigated in [102] for systems with fully measured inputs. Efficient computations of those types of symmetries in the presence of unmeasured inputs are potentially also feasible using the framework derived herein and will be the focus of future extensions of this work.

4.3.1 Translation symmetries

If the i^{th} group of Lie symmetries are translation symmetries then:

$$\begin{aligned} {}^{i}\phi_{x,1} &= x_1 + \alpha_{i,1}\epsilon_i \\ &\vdots \\ {}^{i}\phi_{x,n} &= x_n + \alpha_{i,n}\epsilon_i \\ {}^{i}\phi_{w,1} &= w_1 + \alpha_{i,n+1}\epsilon_i \\ &\vdots \\ {}^{i}\phi_{w,m} &= w_m + \alpha_{i,n+m}\epsilon_i \end{aligned} \tag{4.23}$$

where $\alpha_{i,1}, ..., \alpha_{i,n+m}$ are constant coefficients to be determined. The corresponding ${}^i\boldsymbol{\xi_x}$ and ${}^i\boldsymbol{\xi_w}$ to ${}^i\boldsymbol{\phi_x}$ and ${}^i\boldsymbol{\phi_w}$ can thus be expressed as:

$$\begin{aligned}
{}^i\xi_{x,1} &= \frac{\partial^i \phi_{x,1}}{\partial \epsilon_i} = \alpha_{i,1} \\
&\vdots \\
{}^i\xi_{x,n} &= \frac{\partial^i \phi_{x,n}}{\partial \epsilon_i} = \alpha_{i,n} \\
{}^i\xi_{w,1} &= \frac{\partial^i \phi_{w,1}}{\partial \epsilon_i} = \alpha_{i,n+1} \\
&\vdots \\
{}^i\xi_{w,m} &= \frac{\partial^i \phi_{w,m}}{\partial \epsilon_i} = \alpha_{i,n+m}
\end{aligned} \tag{4.24}$$

If the expressions of ${}^i\boldsymbol{\xi_x}$ and ${}^i\boldsymbol{\xi_w}$ from equation (4.24) are used in equations (4.19) and (4.21), consequently the system of equations in (4.22) can be simplified to:

$$\begin{cases}
\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \begin{bmatrix} \alpha_{i,1} \\ \vdots \\ \alpha_{i,n} \end{bmatrix} + \begin{bmatrix} \frac{\partial f_1}{\partial w_1} & \cdots & \frac{\partial f_1}{\partial w_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial w_1} & \cdots & \frac{\partial f_n}{\partial w_m} \end{bmatrix} \begin{bmatrix} \alpha_{i,n+1} \\ \vdots \\ \alpha_{i,n+m} \end{bmatrix} = \mathbf{0} \\
\begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \cdots & \frac{\partial h_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_p}{\partial x_1} & \cdots & \frac{\partial h_p}{\partial x_n} \end{bmatrix} \begin{bmatrix} \alpha_{i,1} \\ \vdots \\ \alpha_{i,n} \end{bmatrix} + \begin{bmatrix} \frac{\partial h_1}{\partial w_1} & \cdots & \frac{\partial h_1}{\partial w_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_p}{\partial w_1} & \cdots & \frac{\partial h_p}{\partial w_m} \end{bmatrix} \begin{bmatrix} \alpha_{i,n+1} \\ \vdots \\ \alpha_{i,n+m} \end{bmatrix} = \mathbf{0}
\end{cases} \tag{4.25}$$

4.3.2 Scaling symmetries

If the i^{th} group of Lie symmetries are scaling symmetries then:

$$\begin{aligned}
{}^i\phi_{x,1} &= e^{\alpha_{i,1}\epsilon_i} x_1 \\
&\vdots \\
{}^i\phi_{x,n} &= e^{\alpha_{i,n}\epsilon_i} x_n \\
{}^i\phi_{w,1} &= e^{\alpha_{i,n+1}\epsilon_i} w_1 \\
&\vdots \\
{}^i\phi_{w,m} &= e^{\alpha_{i,n+m}\epsilon_i} w_m
\end{aligned} \tag{4.26}$$

where $\alpha_{i,1}, ..., \alpha_{i,n+m}$ are constant coefficients to be determined. The corresponding ${}^{i}\boldsymbol{\xi}_{\boldsymbol{x}}$ and ${}^{i}\boldsymbol{\xi}_{\boldsymbol{w}}$ to ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^{i}\boldsymbol{\phi}_{\boldsymbol{w}}$ can thus be expressed as:

$$\begin{aligned}
{}^{i}\xi_{x,1} &= \frac{\partial^{i}\phi_{x,1}}{\partial\epsilon_i} = \alpha_{i,1}{}^{i}\phi_{x,1} \\
&\vdots \\
{}^{i}\xi_{x,n} &= \frac{\partial^{i}\phi_{x,n}}{\partial\epsilon_i} = \alpha_{i,n}{}^{i}\phi_{x,n} \\
{}^{i}\xi_{w,1} &= \frac{\partial^{i}\phi_{w,1}}{\partial\epsilon_i} = \alpha_{i,n+1}{}^{i}\phi_{w,1} \\
&\vdots \\
{}^{i}\xi_{w,m} &= \frac{\partial^{i}\phi_{w,m}}{\partial\epsilon_i} = \alpha_{i,n+m}{}^{i}\phi_{w,m}
\end{aligned} \tag{4.27}$$

If the expressions of ${}^{i}\boldsymbol{\xi}_{\boldsymbol{x}}$ and ${}^{i}\boldsymbol{\xi}_{\boldsymbol{w}}$ from equation (4.27) are used in equations (4.19) and (4.21), consequently the system of equations in (4.22) can be simplified to:

$$\begin{cases}
\begin{bmatrix} \alpha_{i,1} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \alpha_{i,n} \end{bmatrix} \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix} - \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \dots & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \begin{bmatrix} \alpha_{i,1}x_1 \\ \vdots \\ \alpha_{i,n}x_n \end{bmatrix} \\
- \begin{bmatrix} \frac{\partial f_1}{\partial w_1} & \dots & \frac{\partial f_1}{\partial w_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial w_1} & \dots & \frac{\partial f_n}{\partial w_m} \end{bmatrix} \begin{bmatrix} \alpha_{i,n+1}w_1 \\ \vdots \\ \alpha_{i,n+m}w_m \end{bmatrix} = \mathbf{0} \\
\begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \dots & \frac{\partial h_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_p}{\partial x_1} & \dots & \frac{\partial h_p}{\partial x_n} \end{bmatrix} \begin{bmatrix} \alpha_{i,1}x_1 \\ \vdots \\ \alpha_{i,n}x_n \end{bmatrix} + \begin{bmatrix} \frac{\partial h_1}{\partial w_1} & \dots & \frac{\partial h_1}{\partial w_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_p}{\partial w_1} & \dots & \frac{\partial h_p}{\partial w_m} \end{bmatrix} \begin{bmatrix} \alpha_{i,n+1}w_1 \\ \vdots \\ \alpha_{i,n+m}w_m \end{bmatrix} = \mathbf{0}
\end{cases} \tag{4.28}$$

4.3.3 Mobius symmetries

If the i^{th} group of Lie symmetries are Mobius symmetries then:

$$\begin{aligned} {}^{i}\phi_{x,1} &= \frac{x_1}{1-\epsilon_i\alpha_{i,1}x_1} \\ &\vdots \\ {}^{i}\phi_{x,n} &= \frac{x_n}{1-\epsilon_i\alpha_{i,n}x_n} \\ {}^{i}\phi_{w,1} &= \frac{w_1}{1-\epsilon_i\alpha_{i,n+1}w_1} \\ &\vdots \\ {}^{i}\phi_{w,m} &= \frac{w_m}{1-\epsilon_i\alpha_{i,n+m}w_m} \end{aligned} \tag{4.29}$$

where $\alpha_{i,1},...,\alpha_{i,n+m}$ are constant coefficients to be determined. The corresponding ${}^{i}\boldsymbol{\xi_x}$ and ${}^{i}\boldsymbol{\xi_w}$ to ${}^{i}\boldsymbol{\phi_x}$ and ${}^{i}\boldsymbol{\phi_w}$ can thus be expressed as:

$$\begin{aligned} {}^{i}\xi_{x,1} &= \frac{\partial^i\phi_{x,1}}{\partial\epsilon_i} = \alpha_{i,1}{}^{i}\phi_{x,1}^2 \\ &\vdots \\ {}^{i}\xi_{x,n} &= \frac{\partial^i\phi_{x,n}}{\partial\epsilon_i} = \alpha_{i,n}{}^{i}\phi_{x,n}^2 \\ {}^{i}\xi_{w,1} &= \frac{\partial^i\phi_{w,1}}{\partial\epsilon_i} = \alpha_{i,n+1}{}^{i}\phi_{w,1}^2 \\ &\vdots \\ {}^{i}\xi_{w,m} &= \frac{\partial^i\phi_{w,m}}{\partial\epsilon_i} = \alpha_{i,n+m}{}^{i}\phi_{w,m}^2 \end{aligned} \tag{4.30}$$

If the expressions of ${}^i\boldsymbol{\xi}_{\boldsymbol{x}}$ and ${}^i\boldsymbol{\xi}_{\boldsymbol{w}}$ from equation (4.30) are used in equations (4.19) and (4.21), consequently the system of equations in (4.22) can be simplified to:

$$\begin{cases} \begin{bmatrix} 2\alpha_{i,1}x_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 2\alpha_{i,n}x_2 \end{bmatrix} \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix} - \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \dots & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \begin{bmatrix} \alpha_{i,1}x_1^2 \\ \vdots \\ \alpha_{i,n}x_n^2 \end{bmatrix} \\ - \begin{bmatrix} \frac{\partial f_1}{\partial w_1} & \dots & \frac{\partial f_1}{\partial w_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial w_1} & \dots & \frac{\partial f_n}{\partial w_m} \end{bmatrix} \begin{bmatrix} \alpha_{i,n+1}w_1^2 \\ \vdots \\ \alpha_{i,n+m}w_m^2 \end{bmatrix} = \mathbf{0} \\ \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \dots & \frac{\partial h_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_p}{\partial x_1} & \dots & \frac{\partial h_p}{\partial x_n} \end{bmatrix} \begin{bmatrix} \alpha_{i,1}x_1^2 \\ \vdots \\ \alpha_{i,n}x_n^2 \end{bmatrix} + \begin{bmatrix} \frac{\partial h_1}{\partial w_1} & \dots & \frac{\partial h_1}{\partial w_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_p}{\partial w_1} & \dots & \frac{\partial h_p}{\partial w_m} \end{bmatrix} \begin{bmatrix} \alpha_{i,n+1}w_1^2 \\ \vdots \\ \alpha_{i,n+m}w_m^2 \end{bmatrix} = \mathbf{0} \end{cases} \tag{4.31}$$

All the systems of equations (4.25), (4.28) and (4.31) can be converted to a linear in the coefficients system:

$$\boldsymbol{M}\boldsymbol{\alpha}_i = \mathbf{0} \tag{4.32}$$

where $\boldsymbol{\alpha}_i = [\alpha_{i,1}, ..., \alpha_{i,n+m}]^T$ and $\boldsymbol{M}$ is a matrix of functions of $\boldsymbol{x}$, $\boldsymbol{w}$ and $\boldsymbol{u}$. $\boldsymbol{M}$ can be calculated symbolically by:

$$\boldsymbol{M} = \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\alpha}_i} \tag{4.33}$$

where $\boldsymbol{P}$ is a vector of the left hand side of equations (4.25), (4.28) or (4.31) that can be commonly expressed as:

$$\boldsymbol{P} = \begin{bmatrix} \frac{\partial {}^i\boldsymbol{\xi}_{\boldsymbol{x}}}{\partial \boldsymbol{x}}\boldsymbol{f} - \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}{}^i\boldsymbol{\xi}_{\boldsymbol{x}} - \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{w}}{}^i\boldsymbol{\xi}_{\boldsymbol{w}} \\ \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}{}^i\boldsymbol{\xi}_{\boldsymbol{x}} + \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{w}}{}^i\boldsymbol{\xi}_{\boldsymbol{w}} \end{bmatrix} \tag{4.34}$$

The vector of coefficients $\boldsymbol{\alpha}_i$ can then be determined by calculating the kernel of $\mathbf{M}$ symbolically, i.e. $\boldsymbol{\alpha}_i = ker(\boldsymbol{M})$ giving rise to different bases as groups of realizations of $\boldsymbol{\alpha}_i$. Each basis corresponds to a different parameter ϵ_i. An algorithm was presented in [102] to calculate $\boldsymbol{\alpha_i}$ efficiently for rational nonlinear systems by specializing symbolic variables in $\boldsymbol{M}$ to random values and performing computations over a finite field.

4.3.4 r-parameter group of Lie symmetries

It is often the case that a dynamical system contains multiple one-parameter groups of Lie symmetries. Multiple translation, scaling and Mobius symmetries of the system in equation (4.13) can be obtained by calculating the kernel of $\boldsymbol{M}$ for multiple solutions of bases with each basis corresponding to the coefficients of an independent group of translation, scaling or Mobius symmetries. These obtained groups of translations, scalings and Mobius can be treated separately, or alternatively they can be combined into a single multi-parameter group of Lie symmetries. The principle behind the way of combination is based on the property that Lie transformations of ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^{i}\boldsymbol{\phi}_{\boldsymbol{w}}$ are themselves Lie transformations of $\boldsymbol{x}$ and $\boldsymbol{w}$ that satisfy the equations of the original system [76].

For example, it is assumed that there exists r one-parameter groups of symmetries including r_t groups of translations and r_s groups of scalings without the consideration of Mobius. If one successively applies the translations and scalings in spite of their sequence to transform $\boldsymbol{x}$ and $\boldsymbol{w}$ iteratively, the result gives an r-parameter group of Lie symmetries of the system. Suppose all the translation symmetries are combined first. Let $\boldsymbol{\alpha}_1, ..., \boldsymbol{\alpha}_{r_t}$ be the coefficients of the translations, and the j^{th} component of the combination is given by:

$$x_j + \alpha_{1,j}\epsilon_1 + ... + \alpha_{r_t,j}\epsilon_{r_t} \tag{4.35}$$

Next all the scaling symmetries are combined. Let $\boldsymbol{\alpha}_{r_t+1}, ..., \boldsymbol{\alpha}_r$ be the coefficients of the scalings, and the j^{th} component of the combination is given by:

$$e^{\alpha_{r_t+1,j}\epsilon_{r_t+1}+...+\alpha_{r,j}\epsilon_r}x_j \tag{4.36}$$

Combining all the one-parameter groups of symmetries using equations (4.35) and

(4.36) gives:

$$\phi_{x,j} = e^{\alpha_{r_t+1,j}\epsilon_{r_t+1}+...+\alpha_{r,j}\epsilon_r}(x_j + \alpha_{1,j}\epsilon_1 + ... + \alpha_{r_t,j}\epsilon_{r_t}) \tag{4.37}$$

where $\phi_{x,j}$ is the j^{th} component of $\boldsymbol{\phi_x}$. $\boldsymbol{\phi_x}(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{\epsilon})$ is used to denote the r-parameter group of Lie symmetries with $\boldsymbol{\epsilon} = [\epsilon_1, ..., \epsilon_r]$. $\boldsymbol{\phi_w}(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{\epsilon})$ can be obtained in a similar fashion.

4.3.5 From Lie symmetries to observability

Identical to their one-parameter sub-groups of Lie symmetries, $\boldsymbol{\phi_x}$ and $\boldsymbol{\phi_w}$ as the transformations of $\boldsymbol{x}$ and $\boldsymbol{w}$ satisfy the equations of the considered system leaving $\boldsymbol{u}$ and $\boldsymbol{y}$ unchanged, i.e.

$$\begin{aligned} \dot{\boldsymbol{\phi}}_{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{\phi_x}, \boldsymbol{u}, \boldsymbol{\phi_w}) \\ \boldsymbol{y} &= \boldsymbol{h}(\boldsymbol{\phi_x}, \boldsymbol{u}, \boldsymbol{\phi_w}) \end{aligned} \tag{4.38}$$

This fundamental property builds the relationship between the r-parameter group of Lie symmetries and the observability of the system. If $\phi_{x,j} \not\equiv x_j$, then x_j is unobservable given the measurements of $\boldsymbol{u}$ and $\boldsymbol{y}$. Similarly, if $\phi_{w,j} \not\equiv w_j$, then w_j is unobservable given the measurements of $\boldsymbol{u}$ and $\boldsymbol{y}$. On the contrary, if $[\boldsymbol{\phi_x}, \boldsymbol{\phi_w}] \equiv [\boldsymbol{x}, \boldsymbol{w}]$, then all the states and unmeasured inputs are observable resulting in a fully observable underlying system.

4.3.6 Model reduction

The model of an unobservable system can be reduced to an equivalent model with a minimum number of unobservable states and unmeasured inputs utilizing the results of Lie symmetries. If the system in equation (4.13) is found to contain an r-parameter group of Lie symmetries, then at least a total of r states and unmeasured inputs are

unobservable whose symmetries are functions of $\boldsymbol{\epsilon}$. There exist however a set of transformations of the states and unmeasured inputs, $\boldsymbol{x_T}$, with respect to which the system model can be re-written or reduced such that the reduced model contains up to $n + m - r$ observable states and unmeasured inputs and a minimum number of unobservable variables. In the case where all the unobservable variables could vanish, the model reduction would lead to a fully observable model.

To obtain $\boldsymbol{x_T}$ in the case of transforming an unobservable model to be observable, if possible, a transformation $\boldsymbol{S}$ is first sought by combining the components of $\boldsymbol{\phi_x}$ and $\boldsymbol{\phi_w}$, i.e. $\boldsymbol{S}(\boldsymbol{\phi_x}, \boldsymbol{\phi_w})$, where the goal is to eliminate all the parameters $\boldsymbol{\epsilon}$. $\boldsymbol{x_T}$ is then introduced to be identically equal to $\boldsymbol{S}(\boldsymbol{\phi_x}, \boldsymbol{\phi_w})$. This process will be demonstrated through the following example where some added properties of this transformation allow for obtaining an observable reduced model.

4.3.7 Algorithm

An algorithm is presented in the following to summarize the procedure of computing translation, scaling and Mobius symmetries of the system in equation (4.13). All the computations involved are symbolic, and the resulting multi-parameter group of Lie symmetries from the algorithm is a combination of all the translations, scalings and Mobius computed using the methods described in Section 4.3.1-4.3.4.

Algorithm 4.1

Input: the state-space and measurement equations of the considered system

Output: an r-parameter group of Lie symmetries

1. Compute $\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}$, $\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{w}}$, $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}$ and $\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{w}}$;

2. First the translation symmetries are computed. Let ${}^1\boldsymbol{\alpha} = [\alpha_1, ..., \alpha_n]$ and ${}^2\boldsymbol{\alpha} = [\alpha_{n+1}, ..., \alpha_{n+m}]$. Compute $\boldsymbol{P_1} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}({}^1\boldsymbol{\alpha})^T + \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{w}}({}^2\boldsymbol{\alpha})^T$ and $\boldsymbol{P_2} = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}({}^1\boldsymbol{\alpha})^T + \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{w}}({}^2\boldsymbol{\alpha})^T$;

3. Let $\boldsymbol{P} = [\boldsymbol{P_1}^T, \boldsymbol{P_2}^T]^T$, and compute $\boldsymbol{M} = \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\alpha}}$;

4. Compute $\boldsymbol{\alpha} = ker(\boldsymbol{M})$ and obtain r_t bases of $\boldsymbol{\alpha}$;

5. Substitute the coefficients into equation (4.23) to obtain the translations $[{}^1\boldsymbol{\phi_x}, {}^1\boldsymbol{\phi_w}]$, ..., $[{}^{r_t}\boldsymbol{\phi_x}, {}^{r_t}\boldsymbol{\phi_w}]$;

6. Now the scaling symmetries are computed. Compute $\boldsymbol{P_1} = diag({}^1\boldsymbol{\alpha})\boldsymbol{f} - \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} diag(\boldsymbol{x})({}^1\boldsymbol{\alpha})^T - \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{w}} diag(\boldsymbol{w})({}^2\boldsymbol{\alpha})^T$ and $\boldsymbol{P_2} = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}} diag(\boldsymbol{x})({}^1\boldsymbol{\alpha})^T + \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{w}} diag(\boldsymbol{w})({}^2\boldsymbol{\alpha})^T$;

7. Let $\boldsymbol{P} = [\boldsymbol{P_1}^T, \boldsymbol{P_2}^T]^T$, and compute $\boldsymbol{M} = \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\alpha}}$

8. Compute $\boldsymbol{\alpha} = ker(\boldsymbol{M})$ and obtain r_s bases of $\boldsymbol{\alpha}$;

9. Substitute the coefficients into equation (4.26) to obtain the scalings $[{}^{r_t+1}\boldsymbol{\phi_x}, {}^{r_t+1}\boldsymbol{\phi_w}]$, ..., $[{}^{r_t+r_s}\boldsymbol{\phi_x}, {}^{r_t+r_s}\boldsymbol{\phi_w}]$;

10. Then the Mobius symmetries are computed. Compute $\boldsymbol{P_1} = 2diag({}^1\boldsymbol{\alpha})diag(\boldsymbol{x})\boldsymbol{f} - \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} diag(\boldsymbol{x})^2({}^1\boldsymbol{\alpha})^T - \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{w}} diag(\boldsymbol{w})^2({}^2\boldsymbol{\alpha})^T$ and $\boldsymbol{P_2} = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}} diag(\boldsymbol{x})^2({}^1\boldsymbol{\alpha})^T + \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{w}} diag(\boldsymbol{w})^2({}^2\boldsymbol{\alpha})^T$;

11. Let $\boldsymbol{P} = [\boldsymbol{P_1}^T, \boldsymbol{P_2}^T]^T$, and compute $\boldsymbol{M} = \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{\alpha}}$;

12. Compute $\boldsymbol{\alpha} = ker(\boldsymbol{M})$ and obtain r_m bases of $\boldsymbol{\alpha}$;

13. Substitute the coefficients into equation (4.29) to obtain the Mobius $[{}^{r_t+r_s+1}\boldsymbol{\phi_x}, {}^{r_t+r_s+1}\boldsymbol{\phi_w}]$, ..., $[{}^{r}\boldsymbol{\phi_x}, {}^{r}\boldsymbol{\phi_w}]$;

14. Combine $[{}^1\boldsymbol{\phi_x}, {}^1\boldsymbol{\phi_w}]$, ..., $[{}^{r}\boldsymbol{\phi_x}, {}^{r}\boldsymbol{\phi_w}]$ to obtain $\boldsymbol{\phi_x}$ and $\boldsymbol{\phi_w}$.

4.3.8 Illustrative example 1: a 2 degrees of freedom (2DOFs) mass-spring system

Consider a 2 DOFs mass-spring system as shown in Figure 4.1. The displacements of the two masses m are denoted as x_1 and x_2 respectively, and the corresponding velocities are denoted as v_1 and v_2. k_1 and k_2 are the effective stiffness of the springs. The 2 DOFs system is driven by an unmeasured force $F(t)$ applied at the second

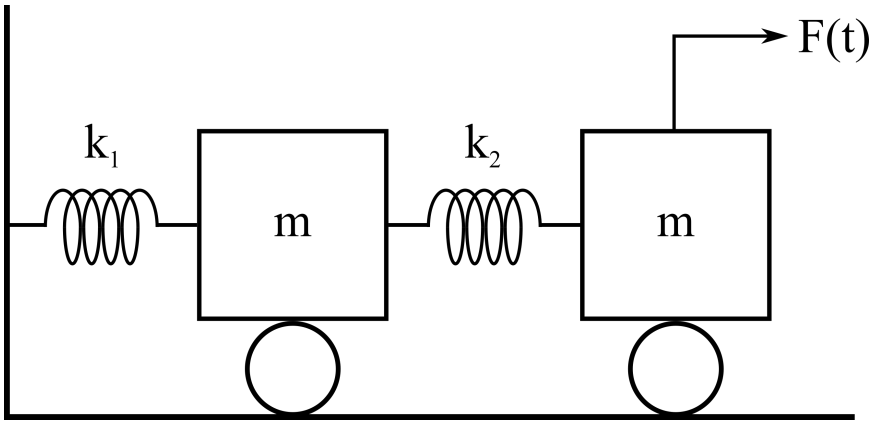

Figure 4.1: A 2 DOFs mass-spring system

mass. The state-space equations of the underlying system are given by:

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ k_1 \\ k_2 \\ m \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ \big(-k_1 x_1 + k_2(x_2 - x_1)\big)/m \\ \big(k_2(x_1 - x_2) + F\big)/m \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{4.39}$$

where k_1, k_2 and m are the unknown parameters to be identified. The displacements x_1 and x_2 are measured, and therefore the measurement equations of the system are given by:

$$\boldsymbol{y} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{4.40}$$

For the example system, Algorithm 4.1 gives a one-parameter group of scaling Lie symmetries:

$$\begin{bmatrix} \boldsymbol{\phi_x} \\ \boldsymbol{\phi_w} \end{bmatrix} = [x_1, x_2, v_1, v_2, e^{\epsilon}k_1, e^{\epsilon}k_2, e^{\epsilon}m, e^{\epsilon}F]^T \tag{4.41}$$

As can be seen from the results of symmetries, the symmetries of x_1, x_2, v_1 and v_2 are identically equal to themselves, which indicates that those states are observable. On the contrary, the parameters k_1, k_2 and m are unidentifiable and the unmeasured

excitation F is unobservable. The observability results suggested by the symmetries are in agreement with the results output from the observability algorithm EORC-DF.

4.3.9 Illustrative example 2: a 2 degrees of freedom (2DOFs) mass-spring system with a Bouc-Wen element

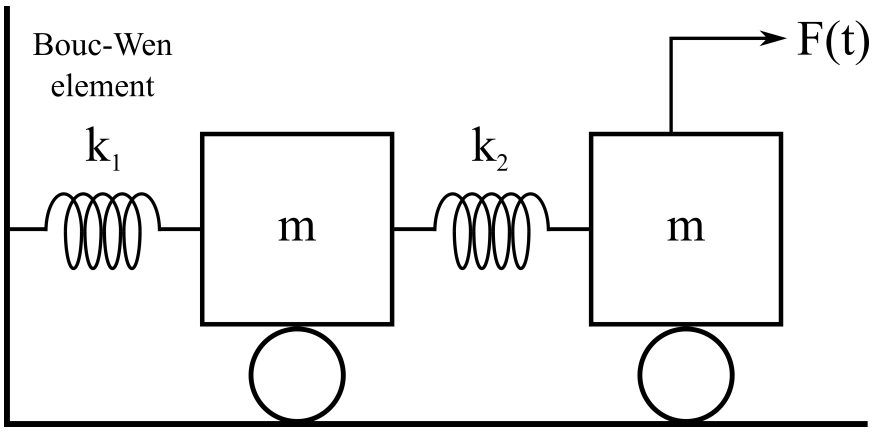

Figure 4.2: A 2 DOFs mass-spring system with a Bouc-Wen element

Consider a 2 DOFs mass-spring system as shown in Figure 4.2. The displacements of the two masses m are denoted as x_1 and x_2 respectively, and the corresponding velocities are denoted as v_1 and v_2. k_1 and k_2 are the effective stiffness of the springs. The first spring is assumed to be a Bouc-Wen element with the elastic displacement r. The 2 DOFs system is driven by an unmeasured force $F(t)$ applied at the second mass. The state-space equations of the underlying system are given by:

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ r \\ k_1 \\ k_2 \\ m \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ \big(-k_1 r + k_2(x_2 - x_1)\big)/m \\ \big(k_2(x_1 - x_2) + F\big)/m \\ v_1 - \beta|v_1||r|r - \gamma v_1|r|^2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{4.42}$$

where β and γ are the Bouc-Wen hysteretic parameters. The parameters k_1, k_2, m,

β and γ are unknown and thus are to be identified given measurements. It should be noted that the exponent of the Bouc-Wen model is assumed to be known and equal to 2. This is not due to the limitations of the method suggested in this work which would allow for studying non-rational systems, but for presenting the results within a more concise way. The displacements x_1 and x_2 are measured, and therefore the measurement equations of the system are given by:

$$\boldsymbol{y} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{4.43}$$

The proposed method in this work cannot be applied directly to the underlying system because the system is not smooth due to the existence of the absolute value operators. However, as discussed in [26], it can be divided into several smooth branches under different conditions of the states, and these smooth branches are then allowed to be examined separately. The state-space equations to describe those branches are given in the following, while the measurement equations remain the same:

$$(A): \frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ r \\ k_1 \\ k_2 \\ m \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ \big(-k_1 r + k_2(x_2 - x_1)\big)/m \\ \big(k_2(x_1 - x_2) + F\big)/m \\ v_1 - \beta v_1 r^2 - \gamma v_1 r^2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad when \quad v_1 > 0 \quad and \quad r > 0 \tag{4.44}$$

$$(B): \frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ r \\ k_1 \\ k_2 \\ m \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ \big(-k_1 r + k_2(x_2 - x_1)\big)/m \\ \big(k_2(x_1 - x_2) + F\big)/m \\ v_1 + \beta v_1 r^2 - \gamma v_1 r^2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad when \quad v_1 < 0 \quad and \quad r > 0$$

$$(C): \frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ r \\ k_1 \\ k_2 \\ m \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ \big(-k_1 r + k_2(x_2 - x_1)\big)/m \\ \big(k_2(x_1 - x_2) + F\big)/m \\ v_1 + \beta v_1 r^2 - \gamma v_1 r^2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad when \quad v_1 > 0 \quad and \quad r < 0$$

$$(D): \frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ r \\ k_1 \\ k_2 \\ m \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ \big(-k_1 r + k_2(x_2 - x_1)\big)/m \\ \big(k_2(x_1 - x_2) + F\big)/m \\ v_1 - \beta v_1 r^2 - \gamma v_1 r^2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad when \quad v_1 < 0 \quad and \quad r < 0$$

For branches A and D, Algorithm 4.1 gives a 2-parameter group of Lie symmetries that is a combination of a group of translations and a group of scalings:

$$\begin{bmatrix} \boldsymbol{\phi_x} \\ \boldsymbol{\phi_w} \end{bmatrix}_{AD} = [x_1, x_2, v_1, v_2, r, e^{\epsilon_1} k_1, e^{\epsilon_1} k_2, e^{\epsilon_1} m, \beta + \epsilon_2, \gamma - \epsilon_2, e^{\epsilon_1} F]^T \tag{4.45}$$

while for branches B and C, the algorithm outputs the following 2-parameter group of Lie symmetries:

$$\begin{bmatrix} \boldsymbol{\phi_x} \\ \boldsymbol{\phi_w} \end{bmatrix}_{BC} = [x_1, x_2, v_1, v_2, r, e^{\epsilon_1} k_1, e^{\epsilon_1} k_2, e^{\epsilon_1} m, \beta + \epsilon_3, \gamma + \epsilon_3, e^{\epsilon_1} F]^T \tag{4.46}$$

As can be seen from the results of symmetries, the symmetries of x_1, x_2, v_1, v_2 and r are identically equal to themselves, which indicates that those states are observable. On the contrary, the parameters k_1, k_2, m, β and γ are unidentifiable and the unmeasured excitation F is unobservable within the corresponding branches. The observability results suggested by the symmetries are in agreement with the results output from the observability algorithm EORC-DF.

In the following, a transformation $\boldsymbol{S}$ is determined for combining the resulting Lie symmetries so as to eliminate all ϵ_1, ϵ_2 and ϵ_3:

$$\boldsymbol{S}(\boldsymbol{\phi_x}, \boldsymbol{\phi_w}) = \begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ r \\ \frac{e^{\epsilon_1} k_1}{e^{\epsilon_1} m} \\ \frac{e^{\epsilon_1} k_2}{e^{\epsilon_1} m} \\ \beta + \epsilon_2 + \gamma - \epsilon_2 \\ \beta + \epsilon_3 - (\gamma + \epsilon_3) \\ \frac{e^{\epsilon_1} F}{e^{\epsilon_1} m} \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ r \\ \frac{k_1}{m} \\ \frac{k_2}{m} \\ \beta + \gamma \\ \beta - \gamma \\ \frac{F}{m} \end{bmatrix} \tag{4.47}$$

A set of new dimensionless variables, f_1, f_2, δ_1, δ_2 and F_m, are then introduced such that:

$$\boldsymbol{x_T} = \begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ r \\ f_1 \\ f_2 \\ \delta_1 \\ \delta_2 \\ F_m \end{bmatrix} \equiv \begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ r \\ \frac{k_1}{m} \\ \frac{k_2}{m} \\ \beta + \gamma \\ \beta - \gamma \\ \frac{F}{m} \end{bmatrix} \tag{4.48}$$

and the models of the 4 branches A, B, C and D can be re-written with respect to

$\boldsymbol{x_T}$ as follows:

$$(A): \frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ r \\ f_1 \\ f_2 \\ \delta_1 \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ -f_1 r + f_2(x_2 - x_1) \\ f_2(x_1 - x_2) + F_m \\ v_1 - \delta_1 v_1 r^2 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad when \quad v_1 > 0 \quad and \quad r > 0 \tag{4.49}$$

$$(B): \frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ r \\ f_1 \\ f_2 \\ \delta_2 \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ -f_1 r + f_2(x_2 - x_1) \\ f_2(x_1 - x_2) + F_m \\ v_1 + \delta_2 v_1 r^2 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad when \quad v_1 < 0 \quad and \quad r > 0$$

$$(C): \frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ r \\ f_1 \\ f_2 \\ \delta_2 \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ -f_1 r + f_2(x_2 - x_1) \\ f_2(x_1 - x_2) + F_m \\ v_1 + \delta_2 v_1 r^2 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad when \quad v_1 > 0 \quad and \quad r < 0$$

$$(D): \frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ r \\ f_1 \\ f_2 \\ \delta_1 \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ -f_1 r + f_2(x_2 - x_1) \\ f_2(x_1 - x_2) + F_m \\ v_1 - \delta_1 v_1 r^2 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad when \quad v_1 < 0 \quad and \quad r < 0$$

It should be noted that in the reduced models in equation (4.49), $f_1 = \frac{k_1}{m}$ and $f_2 = \frac{k_2}{m}$ are related to the natural frequencies of the system, δ_1 and δ_2 are the new Bouc-Wen parameters and $F_m = \frac{F}{m}$ is a new unmeasured input to the system. Given the

same measurements in equation (4.43), the reduced models are observable with all its dynamic states and parameters as well as the unmeasured input observable, which can be verified by the observability algorithm EORC-DF.

4.4 Analytic and power series solutions of general Lie symmetries

Section 4.3 provides a method which allows for efficient computations of certain types of Lie symmetries of the considered system in equation (4.13). Alternative to that, this section proposes a different method with the aim of computing the analytic and power series solutions of general Lie symmetries.

4.4.1 Lie symmetries of the augmented system

To derive the proposed framework, the system in equation (4.13) is augmented, similarly as in Chapter 3 (see details in Section 3.2), by including the states $\boldsymbol{x}$ (containing both dynamic states and unknown parameters in this chapter), the unmeasured inputs $\boldsymbol{w}$ and their time derivatives up to order k in a common state vector, $\boldsymbol{x^k}$:

$$\boldsymbol{x^k} = [\boldsymbol{x}^T, \boldsymbol{w}^T, \dot{\boldsymbol{w}}^T, ..., \boldsymbol{w}^{(k)T}]^T \tag{4.50}$$

such that the state-space and measurement equations of the system with respect to $\boldsymbol{x^k}$ become:

$$\begin{aligned} \dot{\boldsymbol{x}}^{\boldsymbol{k}} &= \begin{bmatrix} \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{w}) \\ \dot{\boldsymbol{w}} \\ \vdots \\ \boldsymbol{w}^{(k)} \\ \boldsymbol{w}^{(k+1)} \end{bmatrix} = \boldsymbol{F^k}(\boldsymbol{x^k},\boldsymbol{u},\boldsymbol{w}^{(k+1)}) \\ \boldsymbol{y} &= \boldsymbol{h}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{w}) \end{aligned} \tag{4.51}$$

where $\boldsymbol{w}^{(k)} = \frac{d^k \boldsymbol{w}}{dt^k}$. It should be noted that the augmented system in equation (4.51) still contains r unmeasured inputs which now coincide with the $(k+1)^{th}$ order time

derivatives of the original unmeasured inputs, i.e. $\boldsymbol{w}^{(k+1)}$.

Recall that the (extended) Lie derivatives of the output function of the augmented system are introduced using the following formulation:

$$\boldsymbol{L_f^j h} = \frac{\partial \boldsymbol{L_f^{j-1} h}}{\partial \boldsymbol{x}} \boldsymbol{f} + \sum_{v=1}^{j} \frac{\partial \boldsymbol{L_f^{j-1} h}}{\partial \boldsymbol{w}^{(v-1)}} \boldsymbol{w}^{(v)} + \sum_{v=1}^{j} \frac{\partial \boldsymbol{L_f^{j-1} h}}{\partial \boldsymbol{u}^{(v-1)}} \boldsymbol{u}^{(v)} \tag{4.52}$$

where $\boldsymbol{L_f^j h}$ is defined as the j^{th} order Lie derivative which can be calculated recursively from the previous order using equation (4.52), given the zero-order $\boldsymbol{L_f^0 h} = \boldsymbol{h}$. A straightforward calculation based on the chain rule shows that the j^{th} order Lie derivative is equivalent to:

$$\boldsymbol{L_f^j h} = \frac{d \boldsymbol{L_f^{j-1} h}}{dt} = \frac{d^j \boldsymbol{h}}{dt^j} \tag{4.53}$$

Now it is assumed the augmented system in equation (4.51) contains r one-parameter groups of Lie symmetries, and its $i^{th} (1 \leq i \leq r)$ group is a one-parameter, $\epsilon_i \in \mathbb{R}$, group of transformations:

$$\begin{aligned} {}^i\boldsymbol{\phi}_{\boldsymbol{x^k}}(\boldsymbol{x^k}, \boldsymbol{w}^{(k+1)}, \epsilon_i) &= [{}^i\boldsymbol{\phi}_{\boldsymbol{x}}^T, {}^i\boldsymbol{\phi}_{\boldsymbol{w}}^T, {}^i\boldsymbol{\phi}_{\dot{\boldsymbol{w}}}^T, ..., {}^i\boldsymbol{\phi}_{\boldsymbol{w}^{(k)}}^T]^T \\ {}^i\boldsymbol{\phi}_{\boldsymbol{w}^{(k+1)}}(\boldsymbol{x^k}, \boldsymbol{w}^{(k+1)}, \epsilon_i) &= [{}^i\phi_{w^{(k+1)},1}, {}^i\phi_{w^{(k+1)},2}, ..., {}^i\phi_{w^{(k+1)},m}]^T \end{aligned} \tag{4.54}$$

where ${}^i\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^i\boldsymbol{\phi}_{\boldsymbol{w}}$ are respectively the corresponding Lie symmetries of $\boldsymbol{x}$ and $\boldsymbol{w}$, and in general are both vectors of nonlinear analytic functions of $\boldsymbol{x^k}$, $\boldsymbol{w}^{(k+1)}$ and the real constant parameter ϵ_i. Obviously, such group of Lie transformations fulfils the equations of the augmented system leaving the measured inputs and the output measurements unchanged, i.e.

$$\begin{aligned} {}^i\dot{\boldsymbol{\phi}}_{\boldsymbol{x^k}} &= \boldsymbol{F^k}({}^i\boldsymbol{\phi}_{\boldsymbol{x^k}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}^{(k+1)}}) \\ \boldsymbol{y} &= \boldsymbol{h}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}}) \end{aligned} \tag{4.55}$$

It should be noted that if the group of Lie symmetries could be determined, it ensures that ${}^i\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^i\boldsymbol{\phi}_{\boldsymbol{w}}$ would be determined as the components of ${}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}$ which fulfils the equations of the original system as in equation (4.15). Further based on the Lie's First Fundamental Theorem, the following differential equations hold with known initial conditions of ${}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}$ and ${}^i\boldsymbol{\phi}_{\boldsymbol{w}^{(k+1)}}$:

$$\begin{aligned} \frac{\partial^i \boldsymbol{\phi}_{\boldsymbol{x}^k}}{\partial \epsilon_i} &= {}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}({}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}^{(k+1)}}), \quad {}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}(\boldsymbol{x}^k, \boldsymbol{w}^{(k+1)}, 0) = \boldsymbol{x}^k \\ \frac{\partial^i \boldsymbol{\phi}_{\boldsymbol{w}^{(k+1)}}}{\partial \epsilon_i} &= {}^i\boldsymbol{\xi}_{\boldsymbol{w}^{(k+1)}}({}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}^{(k+1)}}), \quad {}^i\boldsymbol{\phi}_{\boldsymbol{w}^{(k+1)}}(\boldsymbol{x}^k, \boldsymbol{w}^{(k+1)}, 0) = \boldsymbol{w}^{(k+1)} \end{aligned} \tag{4.56}$$

where ${}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}$ and ${}^i\boldsymbol{\xi}_{\boldsymbol{w}^{(k+1)}}$ are vectors of nonlinear analytic functions. $\frac{\partial^i \boldsymbol{\phi}_{\boldsymbol{x}^k}}{\partial \epsilon_i}\big|_{\epsilon_i=0} = {}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}(\boldsymbol{x}^k, \boldsymbol{w}^{(k+1)})$ and $\frac{\partial^i \boldsymbol{\phi}_{\boldsymbol{w}^{(k+1)}}}{\partial \epsilon_i}\big|_{\epsilon_i=0} = {}^i\boldsymbol{\xi}_{\boldsymbol{w}^{(k+1)}}(\boldsymbol{x}^k, \boldsymbol{w}^{(k+1)})$ are the infinitesimals of the Lie group of transformations.

4.4.2 Analytic computation of the ${}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}$ functions

This section aims to compute the vector of functions ${}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}$ by solving a set of equations analytically. This set of equations are derived based on the fact that the following time derivatives of the measurement equation in equation (4.55) up to any order $j \geq 0$ remain invariant for all possible ϵ_i,

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{h}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}}) \\ \dot{\boldsymbol{y}} &= \frac{d\boldsymbol{h}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}})}{dt} \\ &\vdots \\ \boldsymbol{y}^{(j)} &= \frac{d^j\boldsymbol{h}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}})}{dt^j} \end{aligned} \tag{4.57}$$

and therefore,

$$\begin{aligned}
\frac{d\boldsymbol{y}}{d\epsilon_i} &= \mathbf{0} = \frac{d\boldsymbol{h}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}})}{d\epsilon_i} \\
\frac{d\dot{\boldsymbol{y}}}{d\epsilon_i} &= \mathbf{0} = \frac{d\frac{d\boldsymbol{h}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}})}{dt}}{d\epsilon_i} \\
&\vdots \\
\frac{d\boldsymbol{y}^{(j)}}{d\epsilon_i} &= \mathbf{0} = \frac{d\frac{d^j\boldsymbol{h}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}})}{dt^j}}{d\epsilon_i}
\end{aligned} \tag{4.58}$$

Applying the chain rule to equation (4.58) yields:

$$\begin{aligned}
\frac{\partial \boldsymbol{h}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}})}{\partial {}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}} {}^i\boldsymbol{\xi}_{\boldsymbol{x}^k} + \frac{\partial \boldsymbol{h}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}})}{\partial {}^i\boldsymbol{\phi}_{\boldsymbol{w}^{(k+1)}}} {}^i\boldsymbol{\xi}_{\boldsymbol{w}^{(k+1)}} &= \mathbf{0} \\
\frac{\partial \frac{d\boldsymbol{h}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}})}{dt}}{\partial {}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}} {}^i\boldsymbol{\xi}_{\boldsymbol{x}^k} + \frac{\partial \frac{d\boldsymbol{h}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}})}{dt}}{\partial {}^i\boldsymbol{\phi}_{\boldsymbol{w}^{(k+1)}}} {}^i\boldsymbol{\xi}_{\boldsymbol{w}^{(k+1)}} &= \mathbf{0} \\
&\vdots \\
\frac{\partial \frac{d^j\boldsymbol{h}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}})}{dt^j}}{\partial {}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}} {}^i\boldsymbol{\xi}_{\boldsymbol{x}^k} + \frac{\partial \frac{d^j\boldsymbol{h}({}^i\boldsymbol{\phi}_{\boldsymbol{x}}, \boldsymbol{u}, {}^i\boldsymbol{\phi}_{\boldsymbol{w}})}{dt^j}}{\partial {}^i\boldsymbol{\phi}_{\boldsymbol{w}^{(k+1)}}} {}^i\boldsymbol{\xi}_{\boldsymbol{w}^{(k+1)}} &= \mathbf{0}
\end{aligned} \tag{4.59}$$

The obtained equations in (4.59) are then evaluated at $\epsilon_i = 0$, leading to the following set of equations:

$$\begin{aligned}
\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}^k} {}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}(\boldsymbol{x}^k, \boldsymbol{w}^{(k+1)}) + \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{w}^{(k+1)}} {}^i\boldsymbol{\xi}_{\boldsymbol{w}^{(k+1)}}(\boldsymbol{x}^k, \boldsymbol{w}^{(k+1)}) &= \mathbf{0} \\
\frac{\partial \frac{d\boldsymbol{h}}{dt}}{\partial \boldsymbol{x}^k} {}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}(\boldsymbol{x}^k, \boldsymbol{w}^{(k+1)}) + \frac{\partial \frac{d\boldsymbol{h}}{dt}}{\partial \boldsymbol{w}^{(k+1)}} {}^i\boldsymbol{\xi}_{\boldsymbol{w}^{(k+1)}}(\boldsymbol{x}^k, \boldsymbol{w}^{(k+1)}) &= \mathbf{0} \\
&\vdots \\
\frac{\partial \frac{d^j\boldsymbol{h}}{dt^j}}{\partial \boldsymbol{x}^k} {}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}(\boldsymbol{x}^k, \boldsymbol{w}^{(k+1)}) + \frac{\partial \frac{d^j\boldsymbol{h}}{dt^j}}{\partial \boldsymbol{w}^{(k+1)}} {}^i\boldsymbol{\xi}_{\boldsymbol{w}^{(k+1)}}(\boldsymbol{x}^k, \boldsymbol{w}^{(k+1)}) &= \mathbf{0}
\end{aligned} \tag{4.60}$$

where $\boldsymbol{h}$ corresponds to the measurement function in equation (4.13). From equation (4.53) it is known that $\frac{d^j\boldsymbol{h}}{dt^j} = \boldsymbol{L}_{\boldsymbol{f}}^{\boldsymbol{j}}\boldsymbol{h}$. Furthermore, it can be deduced from equation (4.52) that the Lie derivatives of the augmented system up to order j are independent of $\boldsymbol{w}^{(j+1)}$ and its higher order time derivatives, and therefore $\frac{\partial \boldsymbol{L}_{\boldsymbol{f}}^{\boldsymbol{j}}\boldsymbol{h}}{\partial \boldsymbol{w}^{(k+1)}} = \mathbf{0}$ for $j =$

$0, ..., k$. For $j = k$ equation (4.60) thus becomes:

$$\begin{aligned} \frac{\partial \boldsymbol{L_f^0 h}}{\partial \boldsymbol{x^k}} {}^i\boldsymbol{\xi}_{\boldsymbol{x^k}}(\boldsymbol{x^k}, \boldsymbol{w}^{(k+1)}) &= \mathbf{0} \\ \frac{\partial \boldsymbol{L_f h}}{\partial \boldsymbol{x^k}} {}^i\boldsymbol{\xi}_{\boldsymbol{x^k}}(\boldsymbol{x^k}, \boldsymbol{w}^{(k+1)}) &= \mathbf{0} \\ &\vdots \\ \frac{\partial \boldsymbol{L_f^k h}}{\partial \boldsymbol{x^k}} {}^i\boldsymbol{\xi}_{\boldsymbol{x^k}}(\boldsymbol{x^k}, \boldsymbol{w}^{(k+1)}) &= \mathbf{0} \end{aligned} \tag{4.61}$$

Equation (4.61) can be expressed in the form of matrix multiplication:

$$\begin{bmatrix} \frac{\partial \boldsymbol{L_f^0 h}}{\partial \boldsymbol{x^k}} \\ \frac{\partial \boldsymbol{L_f h}}{\partial \boldsymbol{x^k}} \\ \vdots \\ \frac{\partial \boldsymbol{L_f^k h}}{\partial \boldsymbol{x^k}} \end{bmatrix} {}^i\boldsymbol{\xi}_{\boldsymbol{x^k}}(\boldsymbol{x^k}, \boldsymbol{w}^{(k+1)}) = d\boldsymbol{\Omega^k}{}^i\boldsymbol{\xi}_{\boldsymbol{x^k}}(\boldsymbol{x^k}, \boldsymbol{w}^{(k+1)}) = \mathbf{0} \tag{4.62}$$

It can be immediately observed that the structure of the matrix $d\boldsymbol{\Omega^k}$ is identical to the k-row observability matrix introduced in Chapter 3 (see details in Section 3.2). Using equation (4.62) at $k = k_0$, ${}^i\boldsymbol{\xi}_{\boldsymbol{x^k}}$ can be obtained analytically by computing the kernel of $d\boldsymbol{\Omega^k}$ symbolically for multiple solutions of bases. ${}^i\boldsymbol{\xi}_{\boldsymbol{x^k}} = ker(d\boldsymbol{\Omega^k})$ results in r bases of the kernel i.e. ${}^1\boldsymbol{\xi}_{\boldsymbol{x^k}}, ..., {}^r\boldsymbol{\xi}_{\boldsymbol{x^k}}$ corresponding respectively to r groups of Lie symmetries of the augmented system in equation (4.55). Taking into consideration that different choices of k may lead to different ${}^i\boldsymbol{\xi}_{\boldsymbol{x^k}}(i = 1, ..., r)$, the results of ${}^i\boldsymbol{\xi}_{\boldsymbol{x^k}}$ should be determined at a choice of $k = k_0$ for which ${}^i\boldsymbol{\xi}_{\boldsymbol{x}}$ and ${}^i\boldsymbol{\xi}_{\boldsymbol{w}}$ as the components of ${}^i\boldsymbol{\xi}_{\boldsymbol{x^k}}$ remain invariant for any $k > k_0$. This is associated with the k-row observability properties of $\boldsymbol{x}$ and $\boldsymbol{w}$ suggested from $d\boldsymbol{\Omega^k}$ being convergent as discussed in Chapter 3. Furthermore, it should be noted that since the Lie derivatives up to order k are independent of $\boldsymbol{w}^{(k+1)}$, the matrix $d\boldsymbol{\Omega^k}$ is independent of $\boldsymbol{w}^{(k+1)}$; consequently the bases of the kernel of $d\boldsymbol{\Omega^k}$, i.e. ${}^i\boldsymbol{\xi}_{\boldsymbol{x^k}}(i = 1, ..., r)$, are independent of $\boldsymbol{w}^{(k+1)}$. It is

therefore possible to write equation (4.62) as:

$$\begin{bmatrix} \frac{\partial \boldsymbol{L}_{\boldsymbol{f}}^{\mathbf{0}}\boldsymbol{h}}{\partial \boldsymbol{x}^{\boldsymbol{k}}} \\ \frac{\partial \boldsymbol{L}_{\boldsymbol{f}}\boldsymbol{h}}{\partial \boldsymbol{x}^{\boldsymbol{k}}} \\ \vdots \\ \frac{\partial \boldsymbol{L}_{\boldsymbol{f}}^{\boldsymbol{k}}\boldsymbol{h}}{\partial \boldsymbol{x}^{\boldsymbol{k}}} \end{bmatrix} {}^{i}\boldsymbol{\xi}_{\boldsymbol{x}^{\boldsymbol{k}}}(\boldsymbol{x}^{\boldsymbol{k}}) = d\boldsymbol{\Omega}^{\boldsymbol{k}i}\boldsymbol{\xi}_{\boldsymbol{x}^{\boldsymbol{k}}}(\boldsymbol{x}^{\boldsymbol{k}}) = \mathbf{0} \tag{4.63}$$

The following algorithm summarizes the procedure of symbolically computing the ${}^{i}\boldsymbol{\xi}_{\boldsymbol{x}^{\boldsymbol{k}}}$ functions.

Algorithm 4.2

Input: the state-space and measurement equations of the system in equation (4.13)

Output: the ${}^{i}\boldsymbol{\xi}_{\boldsymbol{x}^{\boldsymbol{k}}}(i = 1, ..., r)$ functions

Initialization: Set $k = 0$, $\boldsymbol{x}^{\boldsymbol{k}} = [\boldsymbol{x}^T, \boldsymbol{w}^T]^T$, $\boldsymbol{L}_{\boldsymbol{f}}^{\boldsymbol{k}}\boldsymbol{h} = \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{w})$, $d\boldsymbol{\Omega}^{\boldsymbol{k}} = \frac{\partial \boldsymbol{L}_{\boldsymbol{f}}^{\boldsymbol{k}}\boldsymbol{h}}{\partial \boldsymbol{x}^{\boldsymbol{k}}}$

1. Set $k = k + 1$;

2. Set $\boldsymbol{x}^{\boldsymbol{k}} = [\boldsymbol{x}^{\boldsymbol{k-1}^T}, \boldsymbol{w}^{(k)T}]^T$;

3. Compute $\boldsymbol{L}_{\boldsymbol{f}}^{\boldsymbol{k}}\boldsymbol{h} = \frac{\partial \boldsymbol{L}_{\boldsymbol{f}}^{\boldsymbol{k-1}}\boldsymbol{h}}{\partial \boldsymbol{x}}\boldsymbol{f} + \sum_{v=1}^{k} \frac{\partial \boldsymbol{L}_{\boldsymbol{f}}^{\boldsymbol{k-1}}\boldsymbol{h}}{\partial \boldsymbol{w}^{(v-1)}}\boldsymbol{w}^{(v)} + \sum_{v=1}^{k} \frac{\partial \boldsymbol{L}_{\boldsymbol{f}}^{\boldsymbol{k-1}}\boldsymbol{h}}{\partial \boldsymbol{u}^{(v-1)}}\boldsymbol{u}^{(v)}$;

4. Compute and arrange $d\boldsymbol{\Omega}^{\boldsymbol{k}} = \begin{bmatrix} d\boldsymbol{\Omega}^{\boldsymbol{k-1}} & \mathbf{0} \\ \frac{\partial \boldsymbol{L}_{\boldsymbol{f}}^{\boldsymbol{k}}\boldsymbol{h}}{\partial \boldsymbol{x}^{\boldsymbol{k}}} & \end{bmatrix}$;

5. Compute the rank of $d\boldsymbol{\Omega}^{\boldsymbol{k}}$, and if $rank(d\boldsymbol{\Omega}^{\boldsymbol{k}}) < n+(k+1)m$, detect the observability of $\boldsymbol{x}$ and $\boldsymbol{w}$;

6. Go to step **8** if $rank(d\boldsymbol{\Omega}^{\boldsymbol{k}}) = n + (k + 1)m$, or $\boldsymbol{x}$ and $\boldsymbol{w}$ are observable, or the observability of $\boldsymbol{x}$ and $\boldsymbol{w}$ has been convergent;

7. Go to step **1**;

8. Compute r bases of the kernel of $d\boldsymbol{\Omega}^{\boldsymbol{k}}$, yielding ${}^{1}\boldsymbol{\xi}_{\boldsymbol{x}^{\boldsymbol{k}}}, ..., {}^{r}\boldsymbol{\xi}_{\boldsymbol{x}^{\boldsymbol{k}}}$.

4.4.3 Analytic and power series solutions of ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^{i}\boldsymbol{\phi}_{\boldsymbol{w}}$

With the expressions of ${}^{i}\boldsymbol{\xi}_{\boldsymbol{x}^{\boldsymbol{k}}}(i = 1, ..., r)$ obtained from Algorithm 4.2, it is now sufficient to compute ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^{i}\boldsymbol{\phi}_{\boldsymbol{w}}(i = 1, ..., r)$ i.e. the groups of Lie symmetries of

the considered system in (4.13), using equation (4.56) i.e.

$$\frac{\partial^i \boldsymbol{\phi}_{\boldsymbol{x}^k}}{\partial \epsilon_i} = {}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}({}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}), \quad {}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}(\boldsymbol{x}^k, \boldsymbol{w}^{(k+1)}, 0) = \boldsymbol{x}^k \tag{4.64}$$

It should be noted that ${}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}$ in equation (4.64) is not a function of ${}^i\boldsymbol{\phi}_{\boldsymbol{w}^{(k+1)}}$ because ${}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}\big|_{\epsilon_i=0}$ is not a function of $\boldsymbol{w}^{(k+1)}$ as shown in the previous section. Analytically solving equation (4.64) as an initial value problem allows for obtaining ${}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}(i = 1, ..., r)$ and thus obtaining ${}^i\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^i\boldsymbol{\phi}_{\boldsymbol{w}}$ as the components of ${}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}$. For certain types of Lie symmetries, such as translation, scaling and Mobius symmetries as described in Section 4.3, their analytic solutions are always computable.

In the general case of Lie symmetries, however, analytically solving equation (4.64) can become an intractable problem or computationally cumbersome. To alleviate this constraint, the main objective of this section is to explore methods for determining the power series solutions or expansions of ${}^i\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^i\boldsymbol{\phi}_{\boldsymbol{w}}$. Given the power series expansions of ${}^i\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^i\boldsymbol{\phi}_{\boldsymbol{w}}$ with the accuracy of a specific order, it is possible to determine the full ${}^i\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^i\boldsymbol{\phi}_{\boldsymbol{w}}$ based on, for example, the use of Hermite-Pade approximation as in [86]. In addition, the relationships between the components of ${}^i\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^i\boldsymbol{\phi}_{\boldsymbol{w}}$ can be deduced from the relationships between their power series expansions, such that the goal of model reduction can be achieved through combining those components to eliminate ϵ_i. Numerical analyses of ${}^i\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^i\boldsymbol{\phi}_{\boldsymbol{w}}$ are also feasible using their power series approximations.

Consider the following power series expansion of ${}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}$ with respect to ϵ_i:

$${}^i\boldsymbol{\phi}_{\boldsymbol{x}^k} = {}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}\big|_{\epsilon_i=0} + \frac{d^i \boldsymbol{\phi}_{\boldsymbol{x}^k}}{d\epsilon_i}\bigg|_{\epsilon_i=0} \epsilon_i + \frac{d^{2i} \boldsymbol{\phi}_{\boldsymbol{x}^k}}{d\epsilon_i^2}\bigg|_{\epsilon_i=0} \frac{\epsilon_i^2}{2} + ... \tag{4.65}$$

The coefficients of the power series can be calculated recursively by:

$$\frac{d^{ji} \boldsymbol{\phi}_{\boldsymbol{x}^k}}{d\epsilon_i^j}\bigg|_{\epsilon_i=0} = \frac{d\frac{d^{j-1\,i} \boldsymbol{\phi}_{\boldsymbol{x}^k}}{d\epsilon_i^{j-1}}}{d\epsilon_i}\bigg|_{\epsilon_i=0} \tag{4.66}$$

Applying the chain rule to equation (4.66) yields:

$$\left.\frac{d^j {}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}}{d\epsilon_i^j}\right|_{\epsilon_i=0} = \left.\frac{\partial \frac{d^{j-1} {}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}}{d\epsilon_i^{j-1}}}{\partial {}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}} {}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}\right|_{\epsilon_i=0} = \frac{\left.\partial \frac{d^{j-1} {}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}}{d\epsilon_i^{j-1}}\right|_{\epsilon_i=0}}{\partial \boldsymbol{x}^k} {}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}(\boldsymbol{x}^k) \tag{4.67}$$

with the initial coefficient:

$$ {}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}\big|_{\epsilon_i=0} = \boldsymbol{x}^k \tag{4.68}$$

Again determining the power series expansion of ${}^i\boldsymbol{\phi}_{\boldsymbol{x}^k} (i = 1, ..., r)$ using equations (4.65), (4.67) and (4.68) would lead to determining the power series expansions of ${}^i\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^i\boldsymbol{\phi}_{\boldsymbol{w}} (i = 1, ..., r)$ as its components. More efficiently, if ${}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}$ in equation (4.64) is further a vector of rational or polynomial functions, the power series expansion of ${}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}$ can be computed through solving equation (4.64) for a power series solution by means of Newton's iteration.

For the sake of simplicity, it is assumed that ${}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}$ is a vector of nonlinear polynomial functions. Details about solving rational ordinary differential equations, in a more general case, using Newton's iteration can be seen in Section 3.3.1 and 3.3.2. The objective of the use of Newton's iteration is to solve equation (4.64) for the power series expansion of ${}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}$ up to an order of interest, say z, through a number of iterative computations. The iterations start from the initial condition $\boldsymbol{x}^k$ which is the first term of the power series. At each iteration, the power series expansion of ${}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}$ is computed up to a certain order smaller than z, and it is then used for the next iteration to compute the power series up to a higher order closer to z. Such iterations terminate when order z is reached. Let ${}^i\boldsymbol{\phi}_{\boldsymbol{x}^k,\boldsymbol{q}}$ denote the power series computed at the q^{th} iteration of the Newton's method. Equation (4.64) can be expressed in the form of $\boldsymbol{P} = \boldsymbol{0}$ as in Section 3.3.1,

$$\boldsymbol{P}(\frac{\partial^i \boldsymbol{\phi}_{\boldsymbol{x}^k}}{\partial \epsilon_i}, {}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}) = \frac{\partial^i \boldsymbol{\phi}_{\boldsymbol{x}^k}}{\partial \epsilon_i} - {}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}({}^i\boldsymbol{\phi}_{\boldsymbol{x}^k}) = \boldsymbol{0} \tag{4.69}$$

The formula of Newton's iteration for equation (4.69) is obtained by means of linearization of the equation around the point ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k}} = {}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k,q-1}}$, yielding:

$$\frac{\partial^i \boldsymbol{\phi}_{\boldsymbol{x^k,q}}}{\partial \epsilon_i} - \frac{\partial^i \boldsymbol{\phi}_{\boldsymbol{x^k,q-1}}}{\partial \epsilon_i} + \left.\frac{\partial \boldsymbol{P}}{\partial^i \boldsymbol{\phi}_{\boldsymbol{x^k}}}\right|_{{}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k}}={}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k,q-1}}} ({}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k,q}} - {}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k,q-1}}) + \boldsymbol{P}\Big|_{{}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k}}={}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k,q-1}}} = \boldsymbol{0} \tag{4.70}$$

where $\frac{\partial \boldsymbol{P}}{\partial^i \boldsymbol{\phi}_{\boldsymbol{x^k}}}$ and $\boldsymbol{P}$ are matrices with power series elements after evaluating at ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k}} = {}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k,q-1}}$. Equation (4.70) is thus a first order linear ODE with a known initial condition, and can be solved analytically using the method of integrating factor, giving the solution:

$${}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k,q}} = {}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k,q-1}} - e^{-\int \frac{\partial \boldsymbol{P}}{\partial^i \boldsymbol{\phi}_{\boldsymbol{x^k}}} d\epsilon_i} \int e^{\int \frac{\partial \boldsymbol{P}}{\partial^i \boldsymbol{\phi}_{\boldsymbol{x^k}}} d\epsilon_i} \boldsymbol{P} d\epsilon_i \tag{4.71}$$

Within equation (4.71), the formal integrations and differentiations are allowed to perform over power series, and the power series of the matrix exponentials and inverses involved can be computed efficiently by using Newton's iteration (see details in Section 3.3.2). If $\boldsymbol{A_g}$ and $\boldsymbol{B_g}$ respectively represent the power series expansions of $e^{-\int \frac{\partial \boldsymbol{P}}{\partial^i \boldsymbol{\phi}_{\boldsymbol{x^k}}} d\epsilon_i}$ and its inverse $e^{\int \frac{\partial \boldsymbol{P}}{\partial^i \boldsymbol{\phi}_{\boldsymbol{x^k}}} d\epsilon_i}$ computed at the g^{th} iteration of the corresponding Newton's iteration applied, the formula for updating $\boldsymbol{A_g}$ and $\boldsymbol{B_g}$ is given by:

$$\begin{aligned} \boldsymbol{A_g} &= \boldsymbol{A_{g-1}} + \boldsymbol{A_{g-1}} \int \boldsymbol{B_{g-1}} (-\frac{\partial \boldsymbol{P}}{\partial^i \boldsymbol{\phi}_{\boldsymbol{x^k}}}) \boldsymbol{A_{g-1}} - \boldsymbol{B_{g-1}} \frac{\partial \boldsymbol{A_{g-1}}}{\partial \epsilon_i} d\epsilon_i \\ \boldsymbol{B_g} &= 2\boldsymbol{B_{g-1}} - \boldsymbol{B_{g-1}} \boldsymbol{A_g} \boldsymbol{B_{g-1}} \end{aligned} \tag{4.72}$$

with the initial conditions $\boldsymbol{A_0} = \boldsymbol{B_0} = \boldsymbol{I}$. Equation (4.71) therefore becomes:

$${}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k,q}} = {}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k,q-1}} - \boldsymbol{A_q} \int \boldsymbol{B_q} \boldsymbol{P} d\epsilon_i \tag{4.73}$$

Starting from ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k,0}} = \boldsymbol{x^k}$, equation (4.73) can readily be used to update ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k,q}}$ from ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k,q-1}}$, where ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k,q}}$ is the power series expansion of ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k}}$ containing more correct terms than ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k,q-1}}$, i.e. is correct up to a higher order than ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x^k,q-1}}$. It has

been proved that the convergence of Newton's iteration is quadratic, which means the number of additional correct terms computed doubles at each iteration. At the q^{th} iteration, the obtained ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}^{k},\boldsymbol{q}}$ have the first 2^q terms correct and should therefore be truncated at order $2^q - 1$, in order to abandon the incorrect terms of larger orders. The iterative process is continued until the q_0^{th} iteration when the maximum order of correct terms $2^{q_0} - 1 \geq z$. The computed power series is then truncated at order z leading to the result of the power series expansion of ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}^{k}}$ of order z.

The following algorithm summarizes the procedure of computing the power series solutions of ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^{i}\boldsymbol{\phi}_{\boldsymbol{w}}$ up to a chosen order z given the polynomial functions ${}^{i}\boldsymbol{\xi}_{\boldsymbol{x}^{k}}$.

Algorithm 4.3

Input: the vector of polynomial functions ${}^{i}\boldsymbol{\xi}_{\boldsymbol{x}^{k}} (i = 1, ..., r)$, and the maximum order of derivation taken into consideration z

Output: the power series solutions of ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^{i}\boldsymbol{\phi}_{\boldsymbol{w}} (i = 1, ..., r)$ of order z

Preprocessing: Convert equation (4.64) to $\boldsymbol{P} = \boldsymbol{0}$ using equation (4.69), and compute $\frac{\partial \boldsymbol{P}}{\partial {}^{i}\boldsymbol{\phi}_{\boldsymbol{x}^{k}}}$ symbolically

Initialization: Set ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}^{k},\boldsymbol{0}} = \boldsymbol{x}^{k}$ and $q = 0$

1. Set $q = q + 1$;

2. Evaluate $\boldsymbol{P}$ and $\frac{\partial \boldsymbol{P}}{\partial {}^{i}\boldsymbol{\phi}_{\boldsymbol{x}^{k}}}$ at ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}^{k}} = {}^{i}\boldsymbol{\phi}_{\boldsymbol{x}^{k},\boldsymbol{q-1}}$, truncated at order $2^q - 1$;

3. Set $g = 0$ and $\boldsymbol{A_0} = \boldsymbol{B_0} = \boldsymbol{I}$, and compute $\boldsymbol{A_q}$ and $\boldsymbol{B_q}$:

a. Set $g = g + 1$;

b. Compute $\boldsymbol{A_g} = \boldsymbol{A_{g-1}} + \boldsymbol{A_{g-1}} \int \boldsymbol{B_{g-1}} (-\frac{\partial \boldsymbol{P}}{\partial {}^{i}\boldsymbol{\phi}_{\boldsymbol{x}^{k}}}) \boldsymbol{A_{g-1}} - \boldsymbol{B_{g-1}} \frac{\partial \boldsymbol{A_{g-1}}}{\partial \epsilon_i} d\epsilon_i$ truncated at order $2^g - 1$;

c. Compute $\boldsymbol{B_g} = 2\boldsymbol{B_{g-1}} - \boldsymbol{B_{g-1}} \boldsymbol{A_g} \boldsymbol{B_{g-1}}$ truncated at order $2^g - 1$;

d. Go to step **4** if $g = q$. Otherwise go to step **a**;

4. Compute ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}^{\boldsymbol{k}},\boldsymbol{q}} = {}^{i}\boldsymbol{\phi}_{\boldsymbol{x}^{\boldsymbol{k}},\boldsymbol{q-1}} - \boldsymbol{A}_{\boldsymbol{q}} \int \boldsymbol{B}_{\boldsymbol{q}} \boldsymbol{P} d\epsilon_i$ truncated at order $2^q - 1$;

5. Go to step **6** if $2^q - 1 \geq z$. Otherwise go to step **1**;

6. Truncate ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}^{\boldsymbol{k}},\boldsymbol{q}}$ at order z, and obtain the power series of ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^{i}\boldsymbol{\phi}_{\boldsymbol{w}}$ as the components of ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}^{\boldsymbol{k}},\boldsymbol{q}}$.

It should be noted that, when using the power series solutions of ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^{i}\boldsymbol{\phi}_{\boldsymbol{w}}$ for numerical analysis, the obtained power series from Algorithm 4.3 may not always be good numerical approximations of ${}^{i}\boldsymbol{\phi}_{\boldsymbol{x}}$ and ${}^{i}\boldsymbol{\phi}_{\boldsymbol{w}}$ for all possible ϵ, even if the power series are truncated at a sufficiently large order. Undertaking convergence analysis is often required to determine whether the obtained power series are convergent to the true Lie symmetries and what the interval of convergence is. According to Taylor's Theorem, such a convergence analysis is based on dealing with the Taylor remainder $R_n(\epsilon) = \frac{1}{n!}\int_0^{\epsilon}(\epsilon - \tau)^{ni}\boldsymbol{\xi}_{\boldsymbol{x},\boldsymbol{w}}{}^{(n)}(\tau)d\tau$ such that $\lim_{n\to\infty} R_n(\epsilon) = 0$. The level of truncation error can also be properly estimated by using the Taylor remainder.

4.4.4 Illustrative example 3: a 2 degrees of freedom (2DOFs) mass-spring system with a nonlinear element

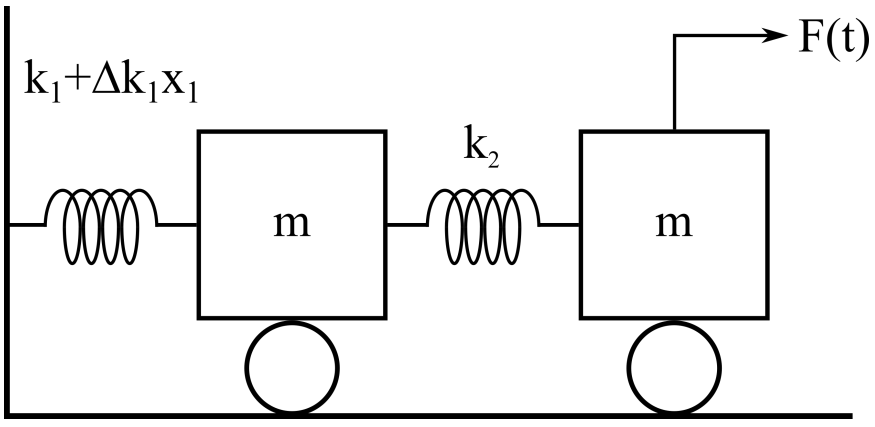

Figure 4.3: A 2 DOFs mass-spring system with a nonlinear element

Consider a 2 DOFs mass-spring system as shown in Figure 4.3. The displacements

of the two masses m are denoted as x_1 and x_2 respectively, and the corresponding velocities are denoted as v_1 and v_2. The stiffness of the nonlinear spring that connects the first mass to the fixed support is displacement-proportional and given by $k_1 + \Delta k_1 x_1$. The stiffness of the second spring is k_2. The 2 DOFs system is driven by an unmeasured force $F(t)$ applied at the second mass. The state-space equations of the underlying system are given by:

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ k_1 \\ \Delta k_1 \\ k_2 \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ \big(-(k_1 x_1 + \Delta k_1 x_1^2) + k_2(x_2 - x_1)\big)/m \\ \big(k_2(x_1 - x_2) + F\big)/m \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{4.74}$$

where k_1, Δk_1, k_2 are unknown parameters to be identified based on output measurements. The mass parameter m is assumed to be known. The accelerations $\ddot{x}_1$ and $\ddot{x}_2$ are measured, and therefore the output measurement equations of the system are given by:

$$\boldsymbol{y} = \begin{bmatrix} \big(-(k_1 x_1 + \Delta k_1 x_1^2) + k_2(x_2 - x_1)\big)/m \\ \big(k_2(x_1 - x_2) + F\big)/m \end{bmatrix} \tag{4.75}$$

Algorithm 4.2 is used to compute the vector of functions ${}^i\boldsymbol{\xi}_{\boldsymbol{x}^k}(i = 1, ..., r)$ for the example system. The number of bases of the kernel of the observability matrix occurring in the algorithm is 1, which implies that the system contains only $r = 1$ one-parameter group of Lie symmetries. The following results of $\boldsymbol{\xi}_{\boldsymbol{x}}$ and $\boldsymbol{\xi}_{\boldsymbol{w}}$ as the components of $\boldsymbol{\xi}_{\boldsymbol{x}^k}$ contain non-zero terms, while the other components of $\boldsymbol{\xi}_{\boldsymbol{x}^k}$ are all zeros which can be ignored for symmetry computation:

$$\begin{aligned} \boldsymbol{\xi}_{\boldsymbol{x}} &= \begin{bmatrix} 1 & \frac{k_1+k_2}{k_2} & 0 & 0 & -2\Delta k_1 & 0 & 0 \end{bmatrix}^T \\ \boldsymbol{\xi}_{\boldsymbol{w}} &= k_1 \end{aligned} \tag{4.76}$$

Using the expressions of $\boldsymbol{\xi_x}$ and $\boldsymbol{\xi_w}$ in equation (4.76), equation (4.64) for the example system can be written as:

$$\begin{aligned} \frac{\partial \boldsymbol{\phi_{x^k}}}{\partial \epsilon_i} &= \frac{\partial}{\partial \epsilon_i}\begin{bmatrix}\boldsymbol{\phi_x}\\ \boldsymbol{\phi_w}\end{bmatrix} = \frac{\partial}{\partial \epsilon_i}\begin{bmatrix}\phi_{x_1}\\ \phi_{x_2}\\ \phi_{v_1}\\ \phi_{v_2}\\ \phi_{k_1}\\ \phi_{\Delta k_1}\\ \phi_{k_2}\\ \phi_w\end{bmatrix} = \begin{bmatrix}1\\ \frac{\phi_{k_1}+\phi_{k_2}}{\phi_{k_2}}\\ 0\\ 0\\ -2\phi_{\Delta k_1}\\ 0\\ 0\\ \phi_{k_1}\end{bmatrix} \\ \begin{bmatrix}\boldsymbol{\phi_x}\\ \boldsymbol{\phi_w}\end{bmatrix}\Big|_{\epsilon=0} &= \begin{bmatrix}x_1 & x_2 & v_1 & v_2 & k_1 & \Delta k_1 & k_2 & F\end{bmatrix}^T \end{aligned} \tag{4.77}$$

Equation (4.77) can be solved analytically as an initial value problem to obtain $\boldsymbol{\phi_x}$ and $\boldsymbol{\phi_w}$, i.e. the one-parameter group of Lie symmetries of the system in equations (4.74) and (4.75). Alternatively, the power series solutions of $\boldsymbol{\phi_x}$ and $\boldsymbol{\phi_w}$ can be computed by means of Newton's iteration using Algorithm 4.3. The results of the analytic and power series solutions of $\boldsymbol{\phi_x}$ and $\boldsymbol{\phi_w}$ given from the algorithms are in absolute agreement:

$$\begin{bmatrix}\boldsymbol{\phi_x}\\ \boldsymbol{\phi_w}\end{bmatrix} = \begin{bmatrix} x_1+\epsilon \\ x_2+\frac{k_1+k_2}{k_2}\epsilon-\frac{\Delta k_1}{k_2}\epsilon^2 \\ v_1 \\ v_2 \\ k_1-2\Delta k_1\epsilon \\ \Delta k_1 \\ k_2 \\ F+k_1\epsilon-\Delta k_1\epsilon^2 \end{bmatrix} \tag{4.78}$$

It should be noted that $\boldsymbol{\phi_x}$ and $\boldsymbol{\phi_w}$ are not translation, scaling or Mobius type symmetries, but rather more general polynomial symmetries. Therefore Algorithm 4.1 presented in Section 4.3.7 is not capable of finding them. Linking the results of symmetries to the observability properties of the system, the symmetries of v_1, v_2, Δk_1 and k_2 are identically equal to themselves, which indicates that those states are observable. On the contrary, x_1, x_2, k_1 and the unmeasured excitation F are

unobservable.

In the following, a transformation $\boldsymbol{S}$ is determined for combining the resulting Lie symmetries so as to eliminate ϵ from as many elements of the vector in equation (4.78) as possible, resulting in:

$$\boldsymbol{S}(\boldsymbol{\phi_x}, \boldsymbol{\phi_w}) = \begin{bmatrix} \phi_{x_1} \\ \phi_{x_2} \\ \phi_{v_1} \\ \phi_{v_2} \\ 2\phi_{\Delta k_1}\phi_{x_1} + \phi_{k_1} \\ \phi_{\Delta k_1} \\ \phi_{k_2} \\ \phi_{k_2}(\phi_{x_1} - \phi_{x_2}) + \phi_w \end{bmatrix} = \begin{bmatrix} x_1 + \epsilon \\ x_2 + \frac{k_1+k_2}{k_2}\epsilon - \frac{\Delta k_1}{k_2}\epsilon^2 \\ v_1 \\ v_2 \\ 2\Delta k_1 x_1 + k_1 \\ \Delta k_1 \\ k_2 \\ k_2(x_1 - x_2) + F \end{bmatrix} \tag{4.79}$$

Two new variables, δ_1 and δ_2, are then introduced such that:

$$\boldsymbol{x_T} = \begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ \delta_1 \\ \Delta k_1 \\ k_2 \\ \delta_2 \end{bmatrix} \equiv \begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ 2\Delta k_1 x_1 + k_1 \\ \Delta k_1 \\ k_2 \\ k_2(x_1 - x_2) + F \end{bmatrix} \tag{4.80}$$

and the state-space and measurement equations of the system can be re-written with respect to $\boldsymbol{x_T}$ as follows:

$$\begin{aligned} \frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ v_1 \\ v_2 \\ \delta_1 \\ \Delta k_1 \\ k_2 \end{bmatrix} &= \begin{bmatrix} v_1 \\ v_2 \\ \big(-(\delta_1 x_1 - \Delta k_1 x_1^2) + k_2(x_2 - x_1)\big)/m \\ \delta_2/m \\ 2\Delta k_1 v_1 \\ 0 \\ 0 \end{bmatrix} \\ \boldsymbol{y} &= \begin{bmatrix} \big(-(\delta_1 x_1 - \Delta k_1 x_1^2) + k_2(x_2 - x_1)\big)/m \\ \delta_2/m \end{bmatrix} \end{aligned} \tag{4.81}$$

Note that in the reduced model in equation (4.81) of the example system, δ_1 is a newly introduced dynamic state, and δ_2 can be treated as an input to the system which is now measured as it appears in the measurement equations. Given the input-output

measurements, x_1 and x_2 are unobservable while the remaining dynamic states and parameters of the model are all observable. This is the case of transforming an unobservable system to be an equivalent model with a minimum number of unobservable states, as described in Section 4.3.6.

In addition to model reduction, the resulting group of Lie symmetries in equation (4.78) is also an implication that if the sensor placement is changed to measure the displacement(s), i.e. x_1 or x_2 or both of them, leading to the following measurement scenarios:

$$\begin{aligned} \boldsymbol{y} &= \begin{bmatrix} x_1 \\ \big(k_2(x_1 - x_2) + F\big)/m \end{bmatrix} \quad or \\ \boldsymbol{y} &= \begin{bmatrix} \big(- (k_1 x_1 + \Delta k_1 x_1^2) + k_2(x_2 - x_1)\big)/m \\ x_2 \end{bmatrix} \quad or \\ \boldsymbol{y} &= \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \end{aligned} \tag{4.82}$$

the example system described by the state-space equations in (4.74) would become fully observable.

4.5 Conclusions

This chapter proposes two novel methods for finding Lie symmetries of nonlinear dynamical systems with unmeasured inputs, one presented in Section 4.3 and the other presented in Section 4.4. Both methods are obtained based on solving a set of differential equations which are derived from 1) the First Fundamental Theorem of Lie, and 2) a fundamental property that Lie symmetries fully satisfy the state-space and output equations of the considered system. Each method has its own advantages and disadvantages in terms of applicability and computation efficiency. The first method is more computationally efficient and is applicable to large engineering systems, such

as a finite element model of a bridge or a skyscraper model. When using the method, assumptions on the forms of Lie symmetries must be made in order to obtain analytic solutions. For relatively simple systems, such as linear structural systems, translation, scaling, affine and Mobius symmetries are commonly encountered in practice, and therefore these symmetries can be used as proper assumptions. For more complicated nonlinear systems, which would not lead to symmetries of some known types, the second method can be used to compute some arbitrary nonlinear Lie symmetries existed. The obtained solutions may be either power series or in analytic forms, depending on whether the differential equations can be analytically solved. The limitation of this method, however, is associated with its high computational cost. The symbolic computation of the observability matrix requires significantly large physical memory in a computer. So far, the second method is only applicable to systems with relatively small size. Furthermore, the work discusses the basic ideas on the use of Lie symmetries to perform model reduction and sensor re-placement for the purpose of improving observability properties, although this has not readily led to a systematic, standardized and automatic procedure.

Chapter 5

Observability and Identification of the Damage-Healing Hysteretic Model

This chapter applies the proposed observability and Lie symmetry algorithms in Chapter 3 and Chapter 4 to study a complex single degree of freedom mass-spring-damper system subject to an earthquake excitation. The nonlinear behaviour of the spring element of the system is described by the damage-healing hysteretic model which is introduced to account for the mechanism of self-healing materials [97, 98, 99]. With the detection of observability properties and the computation of Lie symmetries, the highly nonlinear, non-smooth and unobservable system under study is successfully transformed to become observable through the process of model reduction. The technique of Unscented Kalman filter (UKF) [52] and the recently suggested discontinuous Unscented Kalman filter [24] are used to identify the reduced system using contaminated earthquake record and simulated output data. The performance of the DUKF over the UKF for non-smooth estimation problems is highlighted through the results of identification.

5.1 The damage-healing hysteretic model

Development of novel self-healing materials has gained increasing attention in recent years, while a lot of research has been performed on the engineering materials with self-healing property such as concrete, asphalt, mortar, etc. For better modelling the self-healing mechanism, Triantafyllou et al. [97, 98, 99] extended the generalized Bouc-Wen model to a damage-healing hysteretic model by introducing additional operators to describe how the damage evolves based on the hysteretic energy accumulation, and how the damage is recovered considering the property of self-healing materials.

5.1.1 Constitutive formulation

The triaxial hysteretic formulation has been considered to characterize the elastic and inelastic behaviour of the investigated model [97, 99]. The formulation mainly consists of two evolution equations that are derived from the principles of the additive decomposition of strain rates, the flow rule, the kinematic hardening law and the consistency condition. They are the total stress evolution and the backstress evolution:

$$\begin{aligned} \{\dot{\sigma}\} &= [D]([I] - H_1 H_2[R])\{\dot{\epsilon}\} \\ \{\dot{\eta}\} &= H_1 H_2 G(\{\eta\}, \Phi)[\tilde{R}]\{\dot{\epsilon}\} \end{aligned} \tag{5.1}$$

where $\{\sigma\}$ is the stress tensor, $\{\eta\}$ is the backstress tensor, $\{\epsilon\}$ is the total strain tensor, $[D]$ is the elastic constitutive matrix, $[I]$ is a 6×6 identity matrix and Φ is the yield surface. $[R]$ is the projection of the plastic strain tensor onto the total strain

tensor, defined as:

$$\begin{aligned}[R] &= \frac{\partial \Phi}{\partial \{\sigma\}}[\tilde{R}] \\ [\tilde{R}] &= \big(-(\frac{\partial \Phi}{\partial \{\eta\}})^T G(\{\eta\}, \Phi) + (\frac{\partial \Phi}{\partial \{\sigma\}})^T [D] \frac{\partial \Phi}{\partial \{\sigma\}} \big)^{-1} (\frac{\partial \Phi}{\partial \{\sigma\}})^T [D]\end{aligned} \tag{5.2}$$

H_1 and H_2 are defined as continuous functions of the yield function and the stress field:

$$H_1 = \left| \frac{\Phi}{\Phi_0} \right|^N, N \geq 1, \;\; H_2 = \beta + \gamma sgn\big((\frac{\partial \Phi}{\partial \{\sigma\}})^T \{\dot{\sigma}\}\big) \tag{5.3}$$

where N, β and γ are the Bouc-Wen hysteretic parameters, and Φ_0 is the maximum value of the yield function. G is the kinematic hardening function where the Armstrong-Frederick (AF) model is used:

$$G(\{\eta\}, \Phi) = \frac{2}{3} h \frac{\partial \Phi}{\partial \{\sigma\}} - c \sqrt{\frac{2}{3} (\frac{\partial \Phi}{\partial \{\sigma\}})^T \frac{\partial \Phi}{\partial \{\sigma\}}} \{\eta\} \tag{5.4}$$

where $\frac{h}{c}$ is the saturated value of the backstress.

In [98], damage evolution equations that are associated with stiffness degradation and strength deterioration are introduced to the stress-strain relation. In this work, only the stiffness degradation is taken into account:

$$\{\dot{\sigma}\} = \frac{1}{v_\eta} [D]([I] - H_1 H_2 [R])\{\dot{\epsilon}\}, \;\; v_\eta = 1.0 + c_\eta E_h \tag{5.5}$$

where v_η is a degradation parameter derived by assuming the degradation evolves linearly with the hysteretic energy with a unity initial value. c_η is a material parameter, and E_h is the hysteretic energy of the i^{th} micro-element. In addition, the fundamental property of self-healing material is formulated as h_s to partially offset the effect of the stiffness degradation:

$$\begin{aligned}\{\dot{\sigma}\} &= \frac{h_s}{v_\eta} [D]([I] - H_1 H_2 [R])\{\dot{\epsilon}\} \\ h_s &= 1 + w_h (v_\eta - 1) \frac{1 + sgn\big((\frac{\partial \Phi}{\partial \{\sigma\}})^T \{\dot{\sigma}\}\big)}{2}\end{aligned} \tag{5.6}$$

where w_h is the healing effectiveness parameter that is assumed to be time-independent.

5.1.2 A mass-spring-damper system

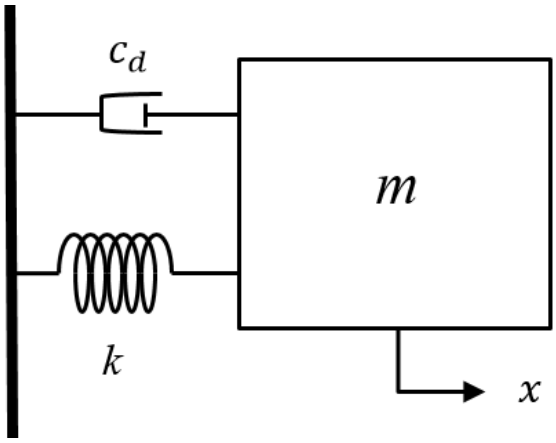

Figure 5.1: A mass-spring-damper system

In this work, the investigated model is a mass-spring-damper system as shown in Figure 5.1. If the excitation applied to the mass is an earthquake, the equation of motion of the dynamical system can be established as follows:

$$m\ddot{x} + c_d\dot{x} + f_s = -\ddot{x}_g m, \quad f_s = A\sigma \tag{5.7}$$

where x is the relative displacement of the mass with respect to the ground, m is the mass parameter, c_d is the damping coefficient and $\ddot{x}$ is the ground acceleration. f_s is the restoring force of the uniaxial spring element, where A is the cross-section area of the spring, and the σ is the stress whose dynamic evolution follows the damage-healing hysteretic formulation (5.6). A von Mises type of yield criterion is considered, leading the function of the yield surface and its critical value to be:

$$\Phi = (\sigma - \eta)^2, \quad \Phi_0 = \sigma_y^2 \tag{5.8}$$

where σ_y is the yield strength. The accumulated plastic dissipated energy E_h is regarded as a time-variant dynamic state, which can be evaluated by integrating the

product of stress and plastic strain rate over time. Therefore, the increasing rate of E_h is:

$$\dot{E}_h = \sigma \dot{\epsilon}_{pl} = \sigma H_1 H_2 R \dot{\epsilon} \tag{5.9}$$

Combining all of the above equations gives the following complete state-space representation of the dynamical system where the evolution equations of all dynamic states, including the displacement of the mass x, the velocity denoted by v, the stress σ, the backstress η and the hysteretic energy E_h, have been established.

$$\begin{aligned}
\dot{x} &= v, \;\; \dot{v} = -\frac{A}{m}\sigma - \frac{c_d}{m}v - \ddot{x}_g \\
\dot{\sigma} &= \frac{1 + w_h c_\eta E_h \frac{1+sgn\left((\sigma-\eta)v\right)}{2}}{1 + c_\eta E_h} E \\
&\quad \Big(1 - (\beta + \gamma sgn((\sigma-\eta)v))(\frac{\sigma-\eta}{\sigma_y})^{2N} \frac{E}{\frac{2}{3}h - \sqrt{\frac{2}{3}} c\eta sgn(\sigma-\eta) + E}\Big)\frac{v}{L} \\
\dot{\eta} &= (\beta + \gamma sgn((\sigma-\eta)v))(\frac{\sigma-\eta}{\sigma_y})^{2N} \frac{\frac{2}{3}h - \sqrt{\frac{2}{3}} c\eta sgn(\sigma-\eta)}{\frac{2}{3}h - \sqrt{\frac{2}{3}} c\eta sgn(\sigma-\eta) + E} E \frac{v}{L} \\
\dot{E}_h &= \sigma(\beta + \gamma sgn((\sigma-\eta)v))(\frac{\sigma-\eta}{\sigma_y})^{2N} \frac{E}{\frac{2}{3}h - \sqrt{\frac{2}{3}} c\eta sgn(\sigma-\eta) + E} \frac{v}{L}
\end{aligned} \tag{5.10}$$

where in a one-dimensional uniaxial spring element the $[D]$ is the Young's modulus E, and the total strain rate $\dot{\epsilon}$ can be calculated from dividing the velocity v by the length of the spring L. It should be noted that in the function $sgn\big((\frac{\partial \Phi}{\partial \{\sigma\}})^T \{\dot{\sigma}\}\big)$, $\frac{\partial \Phi}{\partial \{\sigma\}}$ can be calculated since the function of Φ is known in equation (5.8). It is further considered that the stiffness of the system is always positive so that the signs of $\dot{\sigma}$ and $\dot{\epsilon}$ are always consistent. Hence, implicit equations are avoided for the convenience of

simulation, and the following relation holds:

$$\begin{aligned} sgn\big((\frac{\partial \Phi}{\partial\{\sigma\}})^T\{\dot{\sigma}\}\big) &= sgn\big(2(\sigma-\eta)\dot{\epsilon}\big) \\ &= sgn\big(2(\sigma-\eta)\frac{v}{L}\big) = sgn\big((\sigma-\eta)v\big) \end{aligned} \tag{5.11}$$

5.2 Observability analysis

5.2.1 The observability of non-smooth systems

Recall that the state-space and measurement equations of nonlinear systems can be generally written as:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}), \;\; \boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}) \tag{5.12}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is the state vector, $\boldsymbol{u}$ is the vector containing measured inputs and $\boldsymbol{y}$ is the output measurement vector. $\boldsymbol{f}$ and $\boldsymbol{h}$ are nonlinear functions of $\boldsymbol{x}$ and $\boldsymbol{u}$. Many engineering systems are non-smooth due to the existence of discontinuous functions in their equations. These non-smooth systems are often associated with the physical phenomena of hysteresis, damage, sliding, impact etc. The dynamics of such a system can be expressed in smooth branches where each of the branches can be independently described by an analytic state-space representation which can be regarded as a subsystem of the overall system. Following the idea, Chatzis et al. [26] deduced the observability of a non-smooth system by separating it into a set of smooth subsystems:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f_1}(\boldsymbol{x}, \boldsymbol{u}), \;\; when \;\; \boldsymbol{x} \in \mathbb{R}_1^n \\ &\vdots \\ \dot{\boldsymbol{x}} &= \boldsymbol{f_l}(\boldsymbol{x}, \boldsymbol{u}), \;\; when \;\; \boldsymbol{x} \in \mathbb{R}_l^n \end{aligned} \tag{5.13}$$

where $\boldsymbol{f_i}(\boldsymbol{x}, \boldsymbol{u})$ is an analytic function within $\mathbb{R}_i^n$. One is then allowed to separately implement the observability methods to analyze the observability of each of the smooth subsystems.

The investigated dynamical system described by the state-space equations (5.10) is a non-smooth system due to the existence of $sgn((\sigma - \eta)v)$ and $sgn(\sigma - \eta)$ in the equations of $\dot{\sigma}$, $\dot{\eta}$ and $\dot{E}_h$. This system can be separated into 4 smooth branches: (a) when $(\sigma - \eta) > 0$ and $v > 0$, (b) when $(\sigma - \eta) > 0$ and $v < 0$, (c) when $(\sigma - \eta) < 0$ and $v < 0$, (d) when $(\sigma - \eta) < 0$ and $v > 0$:

$$
\begin{aligned}
& (a) \quad when\ (\sigma - \eta) > 0\ and\ v > 0 \qquad (5.14)\\
\dot{x} &= v,\ \dot{v} = -\frac{A}{m}\sigma - \frac{c_d}{m}v - \ddot{x}_g\\
\dot{\sigma} &= \frac{1 + w_h c_\eta E_h}{1 + c_\eta E_h}E\\
&\quad \Big(1 - (\beta + \gamma)(\frac{\sigma - \eta}{\sigma_y})^{2N}\frac{E}{\frac{2}{3}h - \sqrt{\frac{2}{3}}c\eta + E}\Big)\frac{v}{L}\\
\dot{\eta} &= (\beta + \gamma)(\frac{\sigma - \eta}{\sigma_y})^{2N}\frac{\frac{2}{3}h - \sqrt{\frac{2}{3}}c\eta}{\frac{2}{3}h - \sqrt{\frac{2}{3}}c\eta + E}E\frac{v}{L}\\
\dot{E}_h &= \sigma(\beta + \gamma)(\frac{\sigma - \eta}{\sigma_y})^{2N}\frac{E}{\frac{2}{3}h - \sqrt{\frac{2}{3}}c\eta + E}\frac{v}{L}
\end{aligned}
$$

$$
\begin{aligned}
& (b) \quad when\ (\sigma - \eta) > 0\ and\ v < 0\\
\dot{x} &= v,\ \dot{v} = -\frac{A}{m}\sigma - \frac{c_d}{m}v - \ddot{x}_g\\
\dot{\sigma} &= \frac{1}{1 + c_\eta E_h}E\\
&\quad \Big(1 - (\beta - \gamma)(\frac{\sigma - \eta}{\sigma_y})^{2N}\frac{E}{\frac{2}{3}h - \sqrt{\frac{2}{3}}c\eta + E}\Big)\frac{v}{L}\\
\dot{\eta} &= (\beta - \gamma)(\frac{\sigma - \eta}{\sigma_y})^{2N}\frac{\frac{2}{3}h - \sqrt{\frac{2}{3}}c\eta}{\frac{2}{3}h - \sqrt{\frac{2}{3}}c\eta + E}E\frac{v}{L}\\
\dot{E}_h &= \sigma(\beta - \gamma)(\frac{\sigma - \eta}{\sigma_y})^{2N}\frac{E}{\frac{2}{3}h - \sqrt{\frac{2}{3}}c\eta + E}\frac{v}{L}
\end{aligned}
$$

$$(c) \quad when\ (\sigma - \eta) < 0\ and\ v < 0$$

$$\begin{aligned}
\dot{x} &= v,\ \ \dot{v} = -\frac{A}{m}\sigma - \frac{c_d}{m}v - \ddot{x}_g \\
\dot{\sigma} &= \frac{1 + w_h c_\eta E_h}{1 + c_\eta E_h} E \\
&\quad \Big(1 - (\beta + \gamma)(\frac{\sigma - \eta}{\sigma_y})^{2N} \frac{E}{\frac{2}{3}h + \sqrt{\frac{2}{3}}c\eta + E}\Big)\frac{v}{L} \\
\dot{\eta} &= (\beta + \gamma)(\frac{\sigma - \eta}{\sigma_y})^{2N} \frac{\frac{2}{3}h + \sqrt{\frac{2}{3}}c\eta}{\frac{2}{3}h + \sqrt{\frac{2}{3}}c\eta + E} E\frac{v}{L} \\
\dot{E}_h &= \sigma(\beta + \gamma)(\frac{\sigma - \eta}{\sigma_y})^{2N} \frac{E}{\frac{2}{3}h + \sqrt{\frac{2}{3}}c\eta + E}\frac{v}{L}
\end{aligned}$$

$$(d) \quad when\ (\sigma - \eta) < 0\ and\ v > 0$$

$$\begin{aligned}
\dot{x} &= v,\ \ \dot{v} = -\frac{A}{m}\sigma - \frac{c_d}{m}v - \ddot{x}_g \\
\dot{\sigma} &= \frac{1}{1 + c_\eta E_h} E \\
&\quad \Big(1 - (\beta - \gamma)(\frac{\sigma - \eta}{\sigma_y})^{2N} \frac{E}{\frac{2}{3}h + \sqrt{\frac{2}{3}}c\eta + E}\Big)\frac{v}{L} \\
\dot{\eta} &= (\beta - \gamma)(\frac{\sigma - \eta}{\sigma_y})^{2N} \frac{\frac{2}{3}h + \sqrt{\frac{2}{3}}c\eta}{\frac{2}{3}h + \sqrt{\frac{2}{3}}c\eta + E} E\frac{v}{L} \\
\dot{E}_h &= \sigma(\beta - \gamma)(\frac{\sigma - \eta}{\sigma_y})^{2N} \frac{E}{\frac{2}{3}h + \sqrt{\frac{2}{3}}c\eta + E}\frac{v}{L}
\end{aligned}$$

5.2.2 Unobservable dynamic states and parameters

Suppose the relative displacement is the measured output, i.e. $\boldsymbol{y} = x$, and the earthquake is the measured input, i.e. $\boldsymbol{u} = \ddot{x}_g$. Given the input-output measurements, it is expected to know whether the dynamic states of the investigated system are observable and its parameters are identifiable. 5 dynamic states and all of the remaining time-invariant parameters in equation (5.10) are included in the augmented state vector $\boldsymbol{x}$ such that:

$$\boldsymbol{x} = [x, v, \sigma, \eta, E_h, A_m, c_{dm}, L, E, \sigma_y, c_\eta, w_h, c, h, \beta, \gamma, N]^T \tag{5.15}$$

where A_m and c_{dm} are the normalized cross-area $\frac{A}{m}$ and damping coefficient $\frac{c_d}{m}$ respectively. The corresponding function $\boldsymbol{f}$ can be expressed as:

$$\boldsymbol{f} = [\dot{x}, \dot{v}, \dot{\sigma}, \dot{\eta}, \dot{E}_h, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]^T \tag{5.16}$$

Various observability methods can be used to examine the observability of the subsystems (a), (b), (c) and (d) with fully measured input-output, such as the geometric Observability Rank Condition (ORC) introduced by Hermann and Krener [44] and Algorithm 3.3 proposed in Section 3.3.3 for the case of inputs being fully measured, i.e. $r = 0$. As has been explained before, the ORC is capable of handling general nonlinearities with analytic equations, while Algorithm 3.3 is a robust algorithm applicable to systems with rational equations. The ORC algorithm is extremely inefficient to deal with the systems with complicated equations on a standard computer because it requires significant amount of physical memory. For the reason of computational feasibility, Algorithm 3.3 is used first. However due to the existence of the exponent N in the equations, none of the subsystems (a), (b), (c) and (d) are rational nonlinear systems. An assumption must be made that N is known so

as to apply the algorithm. It should be noted that a dynamic state or a parameter found to be unobservable in a rational system with such an assumption will also be unobservable in the original system, although this is not true for observable states [26].

Algorithm 3.3 reports that based on the given input-output measurements, the unobservable states of the subsystems (a) and (c) detected are:

$$\boldsymbol{x}_{ac}^{u} = [\sigma, \eta, E_h, A_m, L, E, \sigma_y, c_\eta, c, h, \beta, \gamma]^T \tag{5.17}$$

with the transcendence degree of 4, and the unobservable states of the subsystems (b) and (d) are:

$$\boldsymbol{x}_{bd}^{u} = [\sigma, \eta, E_h, A_m, L, E, \sigma_y, c_\eta, w_h, c, h, \beta, \gamma]^T \tag{5.18}$$

with the transcendence degree of 5. The number of unobservable states is different and usually larger than the transcendence degree where the latter exhibits the number of unobservable states which if known would result in all the remaining states being observable. It should be noted that the healing operator is activated only when $(\sigma - \eta)v$ is positive, so the healing effectiveness parameter w_h appears in the corresponding subsystems (a) and (c) but does not exist in (b) and (d), and therefore w_h is unobservable in (b) and (d). As can be seen from the results of Algorithm 3.3, the investigated dynamical systems is a highly unobservable system with at least a total of 12 unobservable dynamic states and parameters.

5.2.3 Lie symmetries

Algorithm 4.1 proposed in Section 4.3.7 is used to calculate the groups of Lie symmetries of the subsystems (a), (b), (c) and (d). See details of the concept and computation of Lie symmetry in Chapter 4. For (a) and (c), the algorithm gives a 3-parameter

group of Lie symmetries that is a combination of a group of translations and 2 groups of scalings:

$$\boldsymbol{\phi}_{ac} = [x, v, e^{\epsilon_2}\sigma, e^{\epsilon_2}\eta, \frac{e^{\epsilon_2}}{e^{\epsilon_3}}E_h, \frac{A_m}{e^{\epsilon_2}}, c_{dm}, e^{\epsilon_3}L, e^{\epsilon_2}e^{\epsilon_3}E, e^{\epsilon_2}\sigma_y, \frac{e^{\epsilon_3}}{e^{\epsilon_2}}c_\eta, w_h, e^{\epsilon_3}c, e^{\epsilon_2}e^{\epsilon_3}h, \beta - \epsilon_1, \gamma_+\epsilon_1]^T \tag{5.19}$$

while for (b) and (d), the algorithm outputs the following 4-parameter group of Lie symmetries that is a combination of a group of translations and 3 groups of scalings:

$$\boldsymbol{\phi}_{bd} = [x, v, e^{\epsilon_2}\sigma, e^{\epsilon_2}\eta, \frac{e^{\epsilon_2}}{e^{\epsilon_3}}E_h, \frac{A_m}{e^{\epsilon_2}}, c_{dm}, e^{\epsilon_3}L, e^{\epsilon_2}e^{\epsilon_3}E, e^{\epsilon_2}\sigma_y, \frac{e^{\epsilon_3}}{e^{\epsilon_2}}c_\eta, e^{\epsilon_4}w_h, e^{\epsilon_3}c, e^{\epsilon_2}e^{\epsilon_3}h, \beta + \epsilon_1, \gamma_+\epsilon_1]^T \tag{5.20}$$

As can be seen from the results of symmetries, the symmetries of x, v and c_{dm} are identically equal to themselves, which indicates that those dynamic states and parameter are observable. On the contrary, σ, η, E_h, A_m, L, E, σ_y, c_η, w_h, c, h, β and γ are unobservable within the corresponding subsystems. The observability results suggested by the symmetries are in absolute agreement with the results output from Algorithm 3.3.

5.2.4 Model reduction

In the following, the state-space equations of the smooth subsystems are reduced by combining the unobservable dynamic states and parameters to eliminate them. The goal is to obtain 4 equivalent observable subsystems through model reduction. To achieve so, σ, η and σ_y at stress-level are first transformed to their corresponding

quantities at displacement-level by submitting the following relations:

$$r = \frac{\sigma L}{E}, \;\; \eta' = \frac{\eta L}{E}, \;\; r_y = \frac{\sigma_y L}{E} \tag{5.21}$$

where r is known as the elastic displacement. Thus L can be eliminated in the main frames, and the second and third equations of (5.10) become:

$$\begin{aligned} \dot{v} &= -\frac{AE}{mL} r - c_{dm} v - \ddot{x}_g \\ \dot{r} &= \frac{h_s}{v_\eta}(1 - H_2(\frac{r-\eta'}{r_y})^{2N} R)v \end{aligned} \tag{5.22}$$

where $\frac{AE}{mL}$ is essentially the normalized stiffness $k_m = \frac{k}{m}$. To eliminate L from the fifth equation, E_h is replaced by $E'_h = \frac{L^2}{E} E_h$. It is found that H_1 and H_2 always occur together as a product in the equations, and they can be rearranged as:

$$H_1 H_2 = \frac{\beta + \gamma sgn((r-\eta')v)}{r_y^{2N}} (r-\eta')^{2N} \tag{5.23}$$

A close inspection of the Bouc-Wen model indicates that identification of the combination $\frac{\beta+\gamma sgn((r-\eta')v)}{r_y^{2N}}$ is possible, but identification of the individual parameters except N is not. To this end, two new parameters are introduced:

$$\Delta_1 = \frac{\beta + \gamma}{r_y^{2N}}, \;\; \Delta_2 = \frac{\beta - \gamma}{r_y^{2N}} \tag{5.24}$$

to represent the combinations of β, γ, r_y and N. Furthermore, E can be completely eliminated in the fraction of G and R by substituting $h' = \frac{h}{E}$. Finally, in order to construct a new system which is mathematically equivalent to the original one, the following combinations are introduced:

$$c' = \frac{c}{L}, \;\; c'_\eta = \frac{E}{L^2} c_\eta \tag{5.25}$$

So far the unobservable parameters E, L and two of β, γ and r_y have been combined to the other states. Now the state vector $\boldsymbol{x}$ of the system is reduced to:

$$\boldsymbol{x_T} = [x, v, r, \eta', E'_h, k_m, c_{dm}, c'_\eta, w_h, c', h', \Delta_1, \Delta_2, N]^T \tag{5.26}$$

The same model reduction strategy can also be achieved in a more systematic way based the use of Lie symmetries following the procedure described in Section 4.3.6. A transformation $\boldsymbol{S}$ is first determined for combining the groups of Lie symmetries in equations (5.19) and (5.20) so as to eliminate all ϵ_1, ϵ_2 and ϵ_3:

$$\boldsymbol{S}(\boldsymbol{\phi}) = \begin{bmatrix} x \\ v \\ \frac{e^{\epsilon_2}\sigma e^{\epsilon_3}L}{e^{\epsilon_2}e^{\epsilon_3}E} \\ \frac{e^{\epsilon_2}\eta e^{\epsilon_3}L}{e^{\epsilon_2}e^{\epsilon_3}E} \\ \frac{e^{2\epsilon_3}L^2}{e^{\epsilon_2}e^{\epsilon_3}E}\frac{e^{\epsilon_2}}{e^{\epsilon_3}}E_h \\ \frac{Ae^{\epsilon_2}e^{\epsilon_3}E}{e^{\epsilon_2}me^{\epsilon_3}L} \\ c_{dm} \\ \frac{e^{\epsilon_2}e^{\epsilon_3}E}{e^{2\epsilon_3}L^2}\frac{e^{\epsilon_3}}{e^{\epsilon_2}}c_\eta \\ w_h \\ \frac{e^{\epsilon_3}c}{e^{\epsilon_3}L} \\ \frac{e^{\epsilon_2}e^{\epsilon_3}h}{e^{\epsilon_2}e^{\epsilon_3}E} \\ \frac{\beta-\epsilon_1+\gamma+\epsilon_1}{(\frac{e^{\epsilon_2}\sigma_y e^{\epsilon_3}L}{e^{\epsilon_2}e^{\epsilon_3}E})^{2N}} \\ \frac{\beta+\epsilon_1-\gamma-\epsilon_1}{(\frac{e^{\epsilon_2}\sigma_y e^{\epsilon_3}L}{e^{\epsilon_2}e^{\epsilon_3}E})^{2N}} \\ N \end{bmatrix} = \begin{bmatrix} x \\ v \\ \frac{\sigma L}{E} \\ \frac{\eta L}{E} \\ \frac{L^2}{E}E_h \\ \frac{AE}{mL} \\ c_{dm} \\ \frac{E}{L^2}c_\eta \\ w_h \\ \frac{c}{L} \\ \frac{h}{E} \\ \frac{\beta+\gamma}{(\frac{\sigma_y L}{E})^{2N}} \\ \frac{\beta-\gamma}{(\frac{\sigma_y L}{E})^{2N}} \\ N \end{bmatrix} \tag{5.27}$$

A new reduced state vector $\boldsymbol{x_T}$ is then introduced such that:

$$\begin{aligned} \boldsymbol{x_T} &= [x, v, r, \eta', E_h', k_m, c_{dm}, c_\eta', w_h, c', h', \Delta_1, \Delta_2, N]^T \\ &\equiv [x, v, \frac{\sigma L}{E}, \frac{\eta L}{E}, \frac{L^2}{E}E_h, \frac{AE}{mL}, c_{dm}, \frac{E}{L^2}c_\eta, w_h, \frac{c}{L}, \frac{h}{E}, \frac{\beta+\gamma}{(\frac{\sigma_y L}{E})^{2N}}, \frac{\beta-\gamma}{(\frac{\sigma_y L}{E})^{2N}}, N]^T \end{aligned} \tag{5.28}$$

With respect to $\boldsymbol{x_T}$, the state-space equations of the subsystems (a), (b), (c) and (d) can be re-written respectively as:

$$(A) \quad when \ (\sigma - \eta') > 0 \ and \ v > 0 \tag{5.29}$$

$$\begin{aligned}
\dot{x} &= v, \ \dot{v} = -k_m r - c_{dm} v - \ddot{x}_g \\
\dot{r} &= \frac{1 + w_h c'_\eta E'_h}{1 + c'_\eta E'_h}(1 - \Delta_1 (r - \eta')^{2N} \frac{1}{\frac{2}{3}h' - \sqrt{\frac{2}{3}} c' \eta' + 1}) v \\
\dot{\eta}' &= \Delta_1 (r - \eta')^{2N} \frac{\frac{2}{3}h' - \sqrt{\frac{2}{3}} c' \eta'}{\frac{2}{3}h' - \sqrt{\frac{2}{3}} c' \eta' + 1} v \\
\dot{E}'_h &= r \Delta_1 (r - \eta')^{2N} \frac{1}{\frac{2}{3}h' - \sqrt{\frac{2}{3}} c' \eta' + 1} v
\end{aligned}$$

$$(B) \quad when \ (\sigma - \eta') > 0 \ and \ v < 0$$

$$\begin{aligned}
\dot{x} &= v, \ \dot{v} = -k_m r - c_{dm} v - \ddot{x}_g \\
\dot{r} &= \frac{1}{1 + c'_\eta E'_h}(1 - \Delta_2 (r - \eta')^{2N} \frac{1}{\frac{2}{3}h' - \sqrt{\frac{2}{3}} c' \eta' + 1}) v \\
\dot{\eta}' &= \Delta_2 (r - \eta')^{2N} \frac{\frac{2}{3}h' - \sqrt{\frac{2}{3}} c' \eta'}{\frac{2}{3}h' - \sqrt{\frac{2}{3}} c' \eta' + 1} v \\
\dot{E}'_h &= r \Delta_2 (r - \eta')^{2N} \frac{1}{\frac{2}{3}h' - \sqrt{\frac{2}{3}} c' \eta' + 1} v
\end{aligned}$$

$$(C) \quad when \ (\sigma - \eta') < 0 \ and \ v < 0$$

$$\begin{aligned}
\dot{x} &= v, \ \dot{v} = -k_m r - c_{dm} v - \ddot{x}_g \\
\dot{r} &= \frac{1 + w_h c'_\eta E'_h}{1 + c'_\eta E'_h}(1 - \Delta_1 (r - \eta')^{2N} \frac{1}{\frac{2}{3}h' + \sqrt{\frac{2}{3}} c' \eta' + 1}) v \\
\dot{\eta}' &= \Delta_1 (r - \eta')^{2N} \frac{\frac{2}{3}h' + \sqrt{\frac{2}{3}} c' \eta'}{\frac{2}{3}h' + \sqrt{\frac{2}{3}} c' \eta' + 1} v \\
\dot{E}'_h &= r \Delta_1 (r - \eta')^{2N} \frac{1}{\frac{2}{3}h' + \sqrt{\frac{2}{3}} c' \eta' + 1} v
\end{aligned}$$

$$
\begin{aligned}
(D) \quad & when\ (\sigma - \eta') < 0\ and\ v > 0 \\
\dot{x} &= v,\ \ \dot{v} = -k_m r - c_{dm} v - \ddot{x}_g \\
\dot{r} &= \frac{1}{1 + c'_\eta E'_h}(1 - \Delta_2 (r - \eta')^{2N} \frac{1}{\frac{2}{3}h' + \sqrt{\frac{2}{3}} c' \eta' + 1}) v \\
\dot{\eta}' &= \Delta_2 (r - \eta')^{2N} \frac{\frac{2}{3}h' + \sqrt{\frac{2}{3}} c' \eta'}{\frac{2}{3}h' + \sqrt{\frac{2}{3}} c' \eta' + 1} v \\
\dot{E}'_h &= r \Delta_2 (r - \eta')^{2N} \frac{1}{\frac{2}{3}h' + \sqrt{\frac{2}{3}} c' \eta' + 1} v
\end{aligned}
$$

An additional benefit of the model reduction is that the ORC with the assistance of model decomposition becomes more efficient to perform accurate observability testing for the 4 reduced subsystems without requiring any assumption. For the subsystems (A) and (C), the observable states and the unobservable parameter are respectively:

$$
\boldsymbol{x}^o_{AC} = [x, v, r, \eta', E'_h, k_m, c_{dm}, c'_\eta, w_h, c', h', \Delta_1, N],\ \ \boldsymbol{x}^u_{AC} = [\Delta_2] \tag{5.30}
$$

while for the subsystems (B) and (D), the observable states and the unobservable parameters are respectively:

$$
\boldsymbol{x}^o_{BD} = [x, v, r, \eta', E'_h, k_m, c_{dm}, c'_\eta, c', h', \Delta_2, N],\ \ \boldsymbol{x}^u_{BD} = [w_h, \Delta_1] \tag{5.31}
$$

All of the reduced subsystems (A), (B), (C) and (D) can be expressed only in terms of the corresponding observable dynamic states and parameters. Finally, as explained in [26], the overall non-smooth system is in fact fully observable since the union of the observable states from different branches is the reduced state vector i.e. $\boldsymbol{x}^o_{AC} \cup \boldsymbol{x}^o_{BD} = \boldsymbol{x}_{\boldsymbol{T}}$.

5.3 System identification methods

After obtaining the observable reduced subsystems in equation (5.29) which equivalently describe the dynamics of the investigated system as the full state-space equations in (5.10), the investigated system can now readily be identified, i.e. the dynamic states can be tracked over time and the parameters can be estimated, using suitably chosen system identification methods. This section provides a basic overview of the Extended Kalman filter, the Unscented Kalman filter, the discontinuous Unscented Kalman filter and their algorithmic implementations, while the details of these Kalman filtering techniques can be found in [92, 54, 53, 23, 25, 27, 24].

5.3.1 The Extended Kalman filter

The standard Kalman filter, appearing as a state estimator, was developed on linear time-invariant state-space models of dynamical systems [54]. The Kalman filtering algorithm succeeds by propagating the state vector mean and covariance, which can be used to represent the approximated probability distribution of the state vector, stepwise over time through a recursive two-stage process, time update and measurement update. In the stage of time update, the state vector mean and covariance are propagated from the previous time step to the current; the stage of measurement update then takes into consideration the output measurement of the current step to update the mean and covariance in a recursive least square sense (see [92] for the details of Kalman filter derivation). The Extended Kalman filter (EKF) provides a direct way to implement the standard filtering algorithm for nonlinear estimation problems, where the system, input and output matrices processed in the algorithm are all obtained by linearising the state transition and observation functions of the

nonlinear system [92].

Assume a nonlinear dynamical system whose discrete state-space and measurement equations are written as:

$$\begin{aligned} \boldsymbol{x_k} &= \boldsymbol{F}(\boldsymbol{x_{k-1}}, \boldsymbol{u_{k-1}}) + \boldsymbol{w_{k-1}} \\ \boldsymbol{y_k} &= \boldsymbol{H}(\boldsymbol{x_k}, \boldsymbol{u_k}) + \boldsymbol{v_k} \end{aligned} \tag{5.32}$$

where $\boldsymbol{x_k} \in \mathbb{R}^n$ is the state vector at time-step t_k. $\boldsymbol{w_k}$ is used to account for the modelling error and $\boldsymbol{v_k}$ is associated with the measurement noise, both of which are considered to be Gaussian white noise processes with covariance matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ respectively. The steps of an EKF algorithm are then summarized in the following:

The EKF Algorithm

1. Initialize the filter at time t_0:

$$\begin{aligned} \boldsymbol{\hat{x}_0} &= E[\boldsymbol{x_0}] \\ \boldsymbol{P_0} &= E[(\boldsymbol{x_0} - \boldsymbol{\hat{x}_0})(\boldsymbol{x_0} - \boldsymbol{\hat{x}_0})^T] \end{aligned}$$

$\star$ For $k = 1, 2, ...,$ perform the time update:

2. Predicted mean and covariance:

$$\begin{aligned} \boldsymbol{\hat{x}_{k|k-1}} &= \boldsymbol{F}(\boldsymbol{\hat{x}_{k-1|k-1}}, \boldsymbol{u_{k-1}}) \\ \boldsymbol{P_{k|k-1}} &= \boldsymbol{F_{k-1}} \boldsymbol{P_{k-1|k-1}} \boldsymbol{F_{k-1}}^T + \boldsymbol{Q} \end{aligned}$$

where

$$\boldsymbol{F_{k-1}} = \left. \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{x}} \right|_{\boldsymbol{\hat{x}_{k-1|k-1}}, \boldsymbol{u_{k-1}}}$$

$\star$ Perform the measurement update:

3. Calculate the Kalman gain:

$$\begin{aligned} \boldsymbol{S_k} &= \boldsymbol{H_k P_{k|k-1} H_k}^T + \boldsymbol{R} \\ \boldsymbol{K_k} &= \boldsymbol{P_{k|k-1} H_k}^T (\boldsymbol{S_k})^{-1} \end{aligned}$$

where

$$\boldsymbol{H_k} = \left. \frac{\partial \boldsymbol{H}}{\partial \boldsymbol{x}} \right|_{\hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}}, \boldsymbol{u_k}}$$

4. Improve the predictions of the mean and covariance using the latest measurement:

$$\begin{aligned} \hat{\boldsymbol{x}}_{\boldsymbol{k|k}} &= \hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}} + \boldsymbol{K_k}(\boldsymbol{y_k} - \boldsymbol{H}(\hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}}, \boldsymbol{u_k})) \\ \boldsymbol{P_{k|k}} &= (\boldsymbol{I} - \boldsymbol{K_k H_k}) \boldsymbol{P_{k|k-1}} \end{aligned}$$

5.3.2 The Unscented Kalman filter

To handle the identification of highly nonlinear systems, the Unscented Kalman filter (UKF) was proposed by making use of the so-called 'Unscented Transform (UT)', instead of linearisation, to achieve robust nonlinear transformations existing in the filtering algorithm [53]. The UKF algorithm succeeds by approximating the state vector as a Gaussian random variable (GRV) represented by a set of carefully chosen deterministic points known as the Sigma Points, which usually results in a faster convergence rate and a more accurate estimation compared to the EKF. Consider the nonlinear dynamical system described in equation (5.32). The steps of an UKF algorithm are then summarized in the following:

The UKF Algorithm

1. Initialize the filter at time t_0:

$$\begin{aligned} \hat{\boldsymbol{x}}_{\boldsymbol{0}} &= E[\boldsymbol{x_0}] \\ \boldsymbol{P_0} &= E[(\boldsymbol{x_0} - \hat{\boldsymbol{x}}_{\boldsymbol{0}})(\boldsymbol{x_0} - \hat{\boldsymbol{x}}_{\boldsymbol{0}})^T] \end{aligned}$$

$\star$ For $k = 1, 2, ...,$ perform the time update:

2. Choose $2n$ Sigma Points $\hat{\boldsymbol{x}}_{\boldsymbol{k-1}}^{(\boldsymbol{i})}$:

$$\begin{aligned}
\hat{\boldsymbol{x}}_{\boldsymbol{k-1}}^{(\boldsymbol{i})} &= \hat{\boldsymbol{x}}_{\boldsymbol{k-1|k-1}} + \tilde{\boldsymbol{x}}^{(\boldsymbol{i})}, \;\; i = 1, ..., 2n \\
\tilde{\boldsymbol{x}}^{(\boldsymbol{i})} &= \sqrt{n\boldsymbol{P}_{\boldsymbol{k-1|k-1}}}_i^T, \;\; i = 1, ..., n \\
\tilde{\boldsymbol{x}}^{(\boldsymbol{i+n})} &= -\sqrt{n\boldsymbol{P}_{\boldsymbol{k-1|k-1}}}_i^T, \;\; i = 1, ..., n
\end{aligned}$$

3. Predicted mean and covariance

$$\begin{aligned}
\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{(\boldsymbol{i})} &= \boldsymbol{F}(\hat{\boldsymbol{x}}_{\boldsymbol{k-1}}^{(\boldsymbol{i})}, \boldsymbol{u}_{\boldsymbol{k-1}}) \\
\hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}} &= \sum_{i=1}^{2n} W_i \hat{\boldsymbol{x}}_{\boldsymbol{k}}^{(\boldsymbol{i})} \\
\boldsymbol{P}_{\boldsymbol{k|k-1}} &= \sum_{i=1}^{2n} W_i (\hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}} - \hat{\boldsymbol{x}}_{\boldsymbol{k}}^{(\boldsymbol{i})})(\hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}} - \hat{\boldsymbol{x}}_{\boldsymbol{k}}^{(\boldsymbol{i})})^T + \boldsymbol{Q}
\end{aligned}$$

where W_i are properly determined weights of the Sigma points.

$\star$ Perform the measurement update:

4. Calculate the output mean:

$$\begin{aligned}
\hat{\boldsymbol{y}}_{\boldsymbol{k}}^{(\boldsymbol{i})} &= \boldsymbol{H}(\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{(\boldsymbol{i})}, \boldsymbol{u}_{\boldsymbol{k}}) \\
\hat{\boldsymbol{y}}_{\boldsymbol{k}} &= \sum_{i=1}^{2n} W_i \hat{\boldsymbol{y}}_{\boldsymbol{k}}^{(\boldsymbol{i})}
\end{aligned}$$

5. Calculate the Kalman gain:

$$\boldsymbol{K}_{\boldsymbol{k}} = \boldsymbol{P}_{\boldsymbol{xy}} \boldsymbol{P}_{\boldsymbol{y}}^{-1}$$

where

$$\begin{aligned}
\boldsymbol{P}_{\boldsymbol{y}} &= \sum_{i=1}^{2n} W_i (\hat{\boldsymbol{y}}_{\boldsymbol{k}} - \hat{\boldsymbol{y}}_{\boldsymbol{k}}^{(\boldsymbol{i})})(\hat{\boldsymbol{y}}_{\boldsymbol{k}} - \hat{\boldsymbol{y}}_{\boldsymbol{k}}^{(\boldsymbol{i})})^T + \boldsymbol{R} \\
\boldsymbol{P}_{\boldsymbol{xy}} &= \sum_{i=1}^{2n} W_i (\hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}} - \hat{\boldsymbol{x}}_{\boldsymbol{k}}^{(\boldsymbol{i})})(\hat{\boldsymbol{y}}_{\boldsymbol{k}} - \hat{\boldsymbol{y}}_{\boldsymbol{k}}^{(\boldsymbol{i})})^T
\end{aligned}$$

6. Improve the predictions of the mean and covariance using the latest measurement:

$$\begin{aligned} \hat{\boldsymbol{x}}_{\boldsymbol{k|k}} &= \hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}} + \boldsymbol{K_k}(\boldsymbol{y_k} - \hat{\boldsymbol{y}}_{\boldsymbol{k}}) \\ \boldsymbol{P_{k|k}} &= \boldsymbol{P_{k|k-1}} - \boldsymbol{K_k P_y K_k}^T \end{aligned}$$

5.3.3 The discontinuous Unscented Kalman filter

For a non-smooth nonlinear system, however, when the UKF propagates its state vector into a specific time interval, the mean of the state vector lies in one of the subsystems of the full system. Within that time interval, some of the components of the state vector are observable and they are therefore converging to their true solutions, while the other components are unobservable. Taking the investigated system of this work as an example, when the state vector $\boldsymbol{x_T}$ lies in the subsystem (A), the parameter Δ_1 is unidentifiable and the remaining components of $\boldsymbol{x_T}$ are observable as shown in equation (5.30). During such time intervals it is argued that the optimal choice would be to update only the observable parts of the state vector and purposely maintain the unobservable ones invariant to avoid their divergence. These unobservable parts of the state vector would be updated when they become observable in a different subsystem, e.g. the parameter Δ_1 becomes identifiable in the subsystems (B) and (D). Following this idea, the discontinuous Unscented Kalman filter (DUKF) [24] was developed as a modification of the UKF to improve the convergence rate and estimation accuracy for non-smooth problems.

In the DUKF algorithm, a transformation vector $\boldsymbol{T}$ is introduced to separate the state vector and covariance matrix into observable $\boldsymbol{o}$, unobservable $\boldsymbol{u}$ and cross $\boldsymbol{uo}$ components. The algorithm updates the observable components of the estimates of the mean vector and covariance matrix during the Kalman updating step: $\hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}} \rightarrow$

$\hat{\boldsymbol{x}}_{\boldsymbol{k|k}}$ and $\boldsymbol{P_{k|k-1}} \rightarrow \boldsymbol{P_{k|k}}$, using an appropriate Kalman gain matrix defined based on the observable components. The unobservable parts are retained invariant, while the cross terms of the covariance matrix are updated using the Schmidt-Kalman filter [85]. The steps of an DUKF algorithm are summarized in the following:

The DUKF Algorithm

$\star$ Steps 1-5 are identical to the UKF's steps 1-5.

6. Separate to unobservable and observable components:

$$\begin{aligned}
\hat{\boldsymbol{x}}'_{\boldsymbol{k|k-1}} &= \boldsymbol{T}\hat{\boldsymbol{x}}_{\boldsymbol{k|k-1}} = [\hat{\boldsymbol{x}}^{\boldsymbol{o}}_{\boldsymbol{k|k-1}}{}^{T}\hat{\boldsymbol{x}}^{\boldsymbol{u}}_{\boldsymbol{k|k-1}}{}^{T}]^{T} \\
\boldsymbol{P}'_{\boldsymbol{k|k-1}} &= \boldsymbol{T}\boldsymbol{P}_{\boldsymbol{k|k-1}}\boldsymbol{T}^{T} = \begin{bmatrix} \boldsymbol{P}^{\boldsymbol{o}}_{\boldsymbol{k|k-1}} & \boldsymbol{P}^{\boldsymbol{uo}}_{\boldsymbol{k|k-1}} \\ \boldsymbol{P}^{\boldsymbol{uo}}_{\boldsymbol{k|k-1}} & \boldsymbol{P}^{\boldsymbol{u}}_{\boldsymbol{k|k-1}} \end{bmatrix} \\
\boldsymbol{P}'_{\boldsymbol{xy}} &= \boldsymbol{T}\boldsymbol{P}_{\boldsymbol{xy}} = \begin{bmatrix} \boldsymbol{P}^{\boldsymbol{o}}_{\boldsymbol{xy}} \\ \boldsymbol{P}^{\boldsymbol{u}}_{\boldsymbol{xy}} \end{bmatrix}
\end{aligned}$$

7. Calculate the Kalman gain:

$$\boldsymbol{K}^{\boldsymbol{o}}_{\boldsymbol{k}} = \boldsymbol{P}^{\boldsymbol{o}}_{\boldsymbol{xy}}\boldsymbol{P}_{\boldsymbol{y}}{}^{-1}$$

8. Update the observable components:

$$\begin{aligned}
\hat{\boldsymbol{x}}^{\boldsymbol{o}}_{\boldsymbol{k|k}} &= \hat{\boldsymbol{x}}^{\boldsymbol{o}}_{\boldsymbol{k|k-1}} + \boldsymbol{K}^{\boldsymbol{o}}_{\boldsymbol{k}}(\boldsymbol{y_k} - \hat{\boldsymbol{y}}_{\boldsymbol{k}}) \\
\boldsymbol{P}^{\boldsymbol{o}}_{\boldsymbol{k|k}} &= \boldsymbol{P}^{\boldsymbol{o}}_{\boldsymbol{k|k-1}} - \boldsymbol{K}^{\boldsymbol{o}}_{\boldsymbol{k}}\boldsymbol{P}_{\boldsymbol{y}}\boldsymbol{K}^{\boldsymbol{o}T}_{\boldsymbol{k}}
\end{aligned}$$

9. Retain the unobservable components invariant:

$$\begin{aligned}
\hat{\boldsymbol{x}}^{\boldsymbol{u}}_{\boldsymbol{k|k}} &= \hat{\boldsymbol{x}}^{\boldsymbol{u}}_{\boldsymbol{k-1|k-1}} \\
\boldsymbol{P}^{\boldsymbol{u}}_{\boldsymbol{k|k}} &= \boldsymbol{P}^{\boldsymbol{u}}_{\boldsymbol{k-1|k-1}}
\end{aligned}$$

10. Update the cross terms:

$$\boldsymbol{P}^{\boldsymbol{uo}}_{\boldsymbol{k|k}} = \boldsymbol{P}^{\boldsymbol{uo}}_{\boldsymbol{k|k-1}} - \boldsymbol{K}^{\boldsymbol{o}}_{\boldsymbol{k}}\boldsymbol{P}^{\boldsymbol{u}}_{\boldsymbol{xy}}$$

11. Rearrange the terms:

$$\begin{aligned}
\hat{\boldsymbol{x}}'_{\boldsymbol{k|k}} &= [\hat{\boldsymbol{x}}^{\boldsymbol{o}}_{\boldsymbol{k|k}}{}^{T} \hat{\boldsymbol{x}}^{\boldsymbol{u}}_{\boldsymbol{k|k}}{}^{T}]^{T} \\
\boldsymbol{P}'_{\boldsymbol{k|k}} &= \begin{bmatrix} \boldsymbol{P}^{\boldsymbol{o}}_{\boldsymbol{k|k}} & \boldsymbol{P}^{\boldsymbol{uo}}_{\boldsymbol{k|k}} \\ \boldsymbol{P}^{\boldsymbol{uo}}_{\boldsymbol{k|k}} & \boldsymbol{P}^{\boldsymbol{u}}_{\boldsymbol{k|k}} \end{bmatrix} \\
\hat{\boldsymbol{x}}_{\boldsymbol{k|k}} &= \boldsymbol{T}' \hat{\boldsymbol{x}}'_{\boldsymbol{k|k}} \\
\boldsymbol{P}_{\boldsymbol{k|k}} &= \boldsymbol{T}' \boldsymbol{P}'_{\boldsymbol{k|k}} \boldsymbol{T}'^{T}
\end{aligned}$$

5.4 Results

In this section, the mass-spring-damper dynamical system shown in Figure 5.1 subject to an earthquake is identified based on the displacement measurement. Recall that the nonlinear dynamic behaviour of the spring element is formulated by the damage-healing hysteretic model. The state-space equations of the system are those corresponding to the state vector $\boldsymbol{x_T}$ in equation (5.28), which can be separated into 4 smooth subsystems (A), (B), (C) and (D) in (5.29).

The earthquake acceleration signal with the sampling frequency of 100Hz and the duration of 20s as shown in Figure 5.2(a) is obtained from the record of Northridge earthquake (1994) [23]. It was filtered with a low-frequency cutoff of 0.13Hz and a high-frequency cutoff of 30Hz. The dynamical system is first simulated forward to calculate its displacement response $x(t)$ to the earthquake as shown in Figure 5.2(b). Both the earthquake and displacement signals are then contaminated with Gaussian white noise with 1% noise-to-signal RMS ratio, and they are used backward as the input-output measurements for this identification problem. The true values of the parameters of the system used for simulation are listed in Table 5.1.

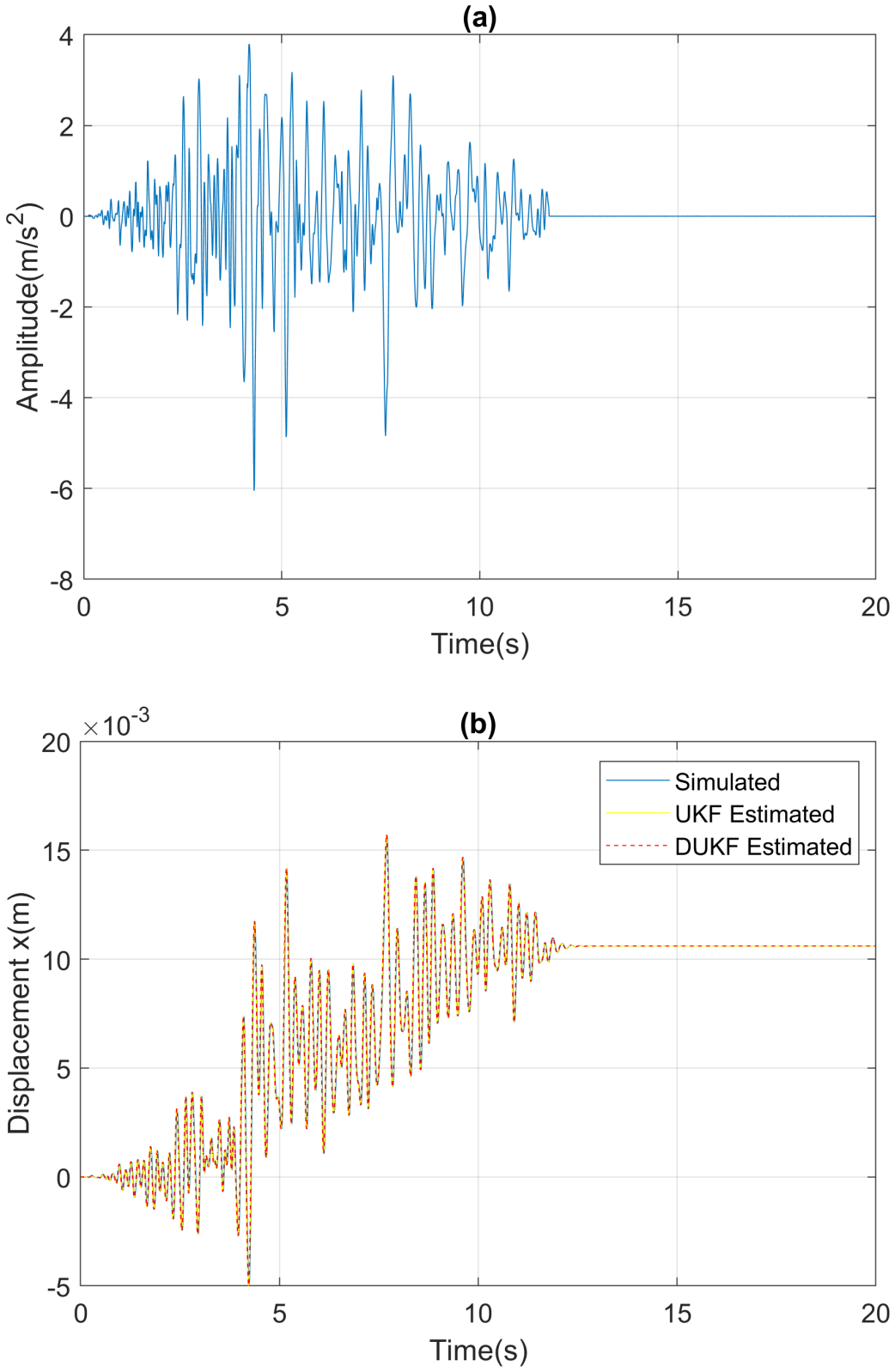

Figure 5.2: (a) The earthquake acceleration signal (b) the simulated and estimated displacements

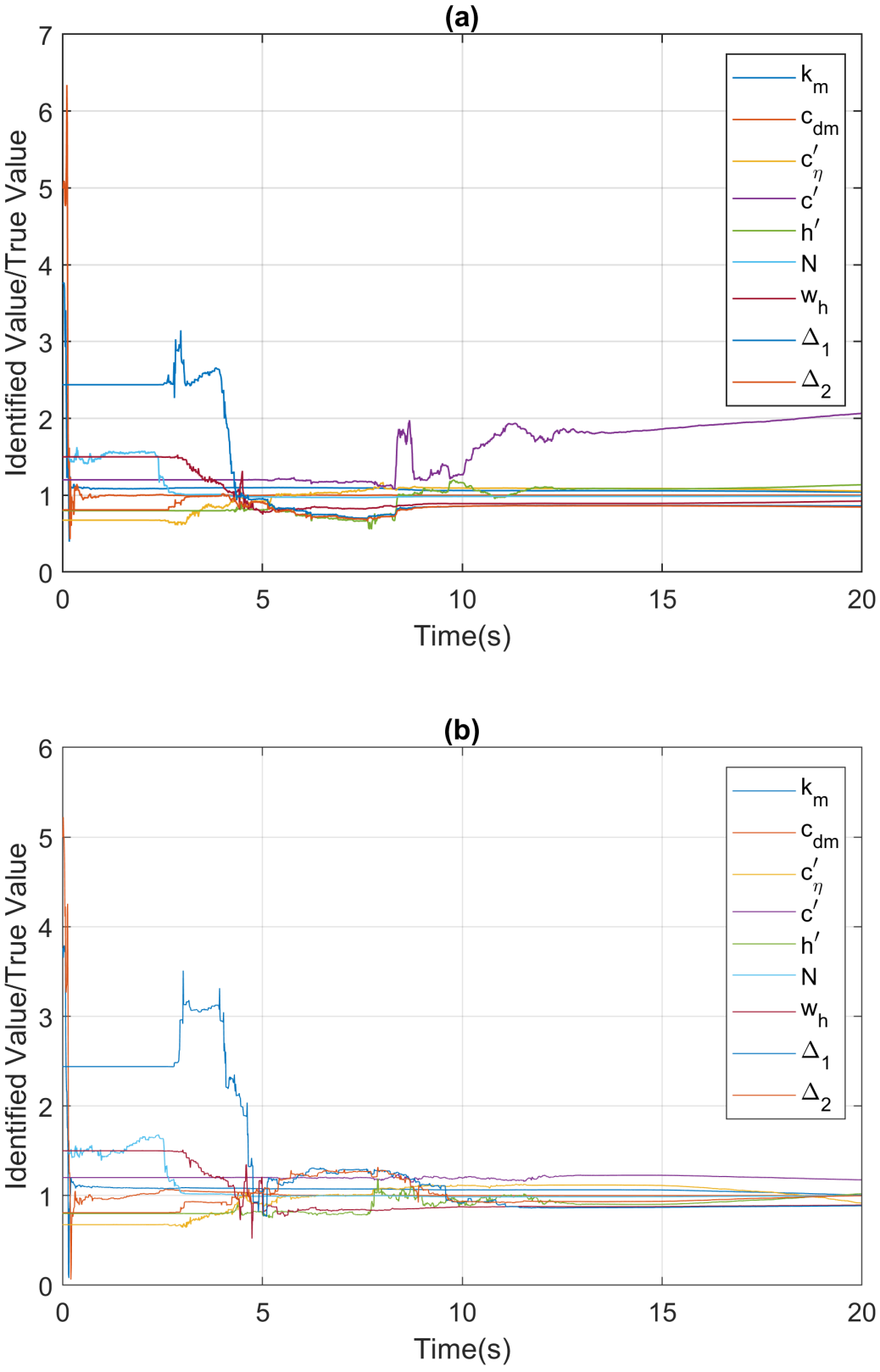

Figure 5.3: Parameter identification from (a) the UKF (b) the DUKF

Parameter		k_m $\times 10^3$	c_{dm} $\times 10$	c'_η $\times 10^3$	c' $\times 1$	h' $\times 10^{-3}$	N $\times 1$	w_h $\times 10^{-1}$	Δ_1 $\times 10^{10}$	Δ_2 $\times 10^{-9}$
	True value	1.33	1.00	4.44	3.33	5.00	2.00	2.00	1.23	-7.41
	Initial value	5	5	3	4	4	3	3	3	-6
DUKF	Identified	1.3373	0.9999	4.0701	3.9167	5.0995	1.9996	1.7892	1.0899	-7.3951
	Error %	0.55	0.01	8.33	17.62	1.99	0.02	10.54	11.39	0.20
UKF	Identified	1.3850	1.0002	4.6827	6.8758	5.6834	1.9744	1.8476	1.0586	-6.2759
	Error %	4.14	0.02	5.47	106.48	13.67	1.28	7.62	13.93	15.30

Table 5.1: True values, initial guesses, identified values and estimation errors of the parameters

The UKF and DUKF algorithms are implemented to identify all the parameters k_m, c_{dm}, c'_η, w_h, c', h', Δ_1, Δ_2 and N (assume their values are unknown), and in the meantime to track the dynamic states x, v, r, η' and E'_h over time. As can be seen in Figure 5.2(b), the UKF and the DUKF successfully track the displacement $x(t)$. The estimations of parameters from the UKF and the DUKF are shown in Figure 5.3(a) and Figure 5.3(b) respectively. Plots of the ratio of identified value over true value are presented such that the estimated values of parameters can be visibly compared to their true values. The initial guesses of parameters used in implementations, the estimated values and estimation errors of those parameters are listed in Table 5.1.

Overall performance of the DUKF is superior to that of the UKF through comparing the results of estimation errors. For the UKF, local divergence of some parameters, such as Δ_2, occurs; these parameters diverge from their true solutions when travelling through the time intervals where they are unidentifiable in the corresponding subsystems. It poses adverse effects on the overall convergence of not only themselves but also the other parameters, such as c' and h'. This problem is effectively addressed in the DUKF by retaining the parameters constant to avoid local divergence in the corresponding branches. The parameters like w_h can still be accurately identified by the UKF as long as they converge faster in identifiable branches than they diverge in unidentifiable branches, as indicated in [27, 24]. For the DUKF, one of the main identification errors is likely from the fact that at a given time instance the estimated subsystem used may be different from the real subsystem because the condition of subsystem is evaluated from the estimated states rather than the real states.

5.5 Unknown restoring force

The previous sections show the mass-spring-damper system in Figure 5.1 can be successfully reduced and identified when the behaviour of the spring element is assumed to be described by the damage-healing hysteretic model. Moreover the restoring force of the spring can be treated as an unknown input, and it is of interest to examine whether it is possible to identify the mass-spring-damper system based on some suitably chosen measurements. Consider the following state-space equations of the system:

$$\frac{d}{dt}\begin{bmatrix} x \\ v \\ c_{dm} \end{bmatrix} = \begin{bmatrix} v \\ -c_{dm}v - f_{sm} - \ddot{x}_g \\ 0 \end{bmatrix} \tag{5.33}$$

where $f_{sm} = \frac{f_s}{m}$ is the scaled restoring force of the spring, which is treated as an unmeasured (unknown) input to the system. The displacement and acceleration of the system are assumed to be measured, and therefore:

$$\boldsymbol{y} = \begin{bmatrix} x \\ -c_{dm}v - f_{sm} - \ddot{x}_g \end{bmatrix} \tag{5.34}$$

Given the state-space and output measurement equations, the EORC-DF algorithm reports that the investigated system is observable. This means that it is theoretically feasible to identify the scaled damping parameter c_{dm}, the velocity v and the scaled restoring force f_{sm}. It indicates that the system can be identified even if no model is assumed for the restoring force as long as a proper input estimation method is used. This is important because one can remove the bias introduced by a model and also the problem with less unknown quantities becomes simpler to tackle.

5.6 Conclusions

In this chapter, a complex single degree of freedom mass-spring-damper system under an earthquake excitation is studied. The spring element of the system is described by the damage-healing hysteretic model which is introduced to account for nonlinear behaviour of self-healing materials. The algorithm proposed in Chapter 3 and the concept of non-smooth observability are utilized to analyze the observability properties of the model. The groups of Lie symmetries of the model are calculated using the algorithm proposed in Chapter 4, and the investigated model, which is observed highly nonlinear, non-smooth and unobservable, is reduced to a fully observable model given the displacement measurement. The Unscented Kalman filter (UKF) and the discontinuous Unscented Kalman filter (DUKF) are implemented respectively to identify the unknown parameters of the reduced model, and the superior performance of the DUKF for parametric identification of non-smooth systems is discussed.

Chapter 6

Conclusions and Future Work

In this concluding chapter, the main contributions of this thesis are summarized and some important directions of future work are discussed.

6.1 Summary and conclusions

The central contribution of the thesis is on the developments of robust and efficient frameworks for computing observability and Lie symmetries of linear or nonlinear dynamical systems with or without unmeasured inputs. These developed tools are expected to play impactful roles in structural health monitoring and potentially in any scientific or industrial field where system identification techniques are employed. In a more detailed perspective, the contribution of each chapter is summarized in the following:

In Chapter 2, a robust algorithm is developed to implement the Observability Rank Condition (ORC) for the observability of the dynamic states and the identifiability of the parameters of large linear systems. The category of linear systems with unknown parameters covers a wide range of civil and mechanical systems that are frequently encountered in practice. Applications and superior performance of the algorithm are shown by successfully testing several examples of large engineering

systems, including the 100-floor high-rise shear building and the 3D FE model of a large truss-beam bridge. To the best of author's knowledge, none of the existing observability methods have the ability to deal with the problems of such sizes and complexities. The occurrence of the algorithm allows researchers and engineers to freely investigate the viability of various structural models and sensor patterns in realistic system identification campaigns without concerning about the limitations of the standard ORC implementation.

In Chapter 3, an efficient algorithm is developed to examine the observability and identifiability of rational nonlinear systems with unmeasured inputs. The category of systems of interest covers a broad range of civil, mechanical, biological, chemical and electrical engineering systems, including those systems whose state-space equations can be rationalized. The underlying theory behind the algorithm is based on the extension of the extended Observability Rank Condition (EORC-DF) to relax the requirement of systems being affine in inputs. A power series-based computational framework enables the algorithm to be implemented robustly for large and complex systems. A high-rise shear building, the 2D FE model of a wind turbine and the 2D FE model of a large bridge are successfully tested as examples to demonstrate the usefulness, robustness and efficiency of the proposed algorithm. To the best of author's knowledge, none of the existing unknown-input observability methods have the ability to deal with the problems of such sizes and complexities. The obtained observability results of the wind turbine and bridge models also suggest the possibility of performing substructural identification in order for more efficient system identification or structural health monitoring.

In Chapter 4, two computational methods are developed for Lie symmetries of

nonlinear systems with unmeasured inputs. The first method allows for efficiently computing the analytic solutions of translation, scaling, Mobius and some commonly encountered Lie symmetries with known forms. The second method is able to find the analytic or power series solutions of arbitrary nonlinear Lie symmetries although in a more computationally expensive way. The resulting Lie symmetries found from the methods can provide an alternative standpoint to understand the observability and identifiability properties of dynamical systems. More importantly, Lie symmetries imply the mathematical relationship between the true solutions of the system's states, parameters and unmeasured inputs and their other possible solutions. In addition, the chapter performs pioneer investigations on the utilization of Lie symmetries for model reduction, sensor re-placement, etc. for the purpose of improving system identification results.

In Chapter 5, a single degree of freedom mass-spring-damper system subject to an earthquake excitation is studied to demonstrate the applications of observability and Lie symmetry analyses for system identification. The spring element of the system is described by the damage-healing hysteretic model which is developed to account for nonlinear behaviour of healing materials. With the detection of observability properties and the computation of Lie symmetries, the highly nonlinear, non-smooth and unobservable system under study is successfully transformed to become observable through the process of model reduction. The techniques of Unscented Kalman filter (UKF) and discontinuous Unscented Kalman filter (DUKF) are used to identify the reduced system using contaminated earthquake record and simulated output data. The performance of the DUKF over the UKF for non-smooth estimation problems is discussed through the results of identification.

6.2 Future directions and open questions

Following the developments and studies on the observability and Lie symmetry problems in this thesis, this section provides a number of recommendations for future work and highlight several open questions awaiting researchers to solve.

Comparing the algorithms introduced in Chapter 2 and Chapter 3, the algorithm in Chapter 2 provides a more efficient means to assess observability properties, although at the cost of narrower applicability. It will therefore be worth further extending the algorithm for linear systems with unmeasured inputs based on using the theory of EORC-DF. A straightforward work that can be done in the near future is to extend it for linear systems with unknown parameters, unmeasured inputs and direct feedthrough, which are written in the following state-space representation:

$$\dot{\boldsymbol{x}} = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{x} + \sum_{i=1}^{m} \boldsymbol{B_{ui}}(\boldsymbol{\theta})u_i + \sum_{i=1}^{r} \boldsymbol{B_{wi}}(\boldsymbol{\theta})w_i, \qquad \dot{\boldsymbol{\theta}} = \boldsymbol{0}$$
$$\boldsymbol{y} = \boldsymbol{C}(\boldsymbol{\theta})\boldsymbol{x} + \sum_{i=1}^{m} \boldsymbol{D_{ui}}(\boldsymbol{\theta})u_i + \sum_{i=1}^{r} \boldsymbol{D_{wi}}(\boldsymbol{\theta})w_i$$

where $\boldsymbol{x}$ is the dynamic state vector, $\boldsymbol{\theta}$ is the unknown parameter vector, $\boldsymbol{u} = [u_1, ..., u_m]^T$ is the vector of measured inputs and $\boldsymbol{w} = [w_1, ..., w_r]^T$ is the vector of unmeasured inputs. $\boldsymbol{y}$ is the output measurement vector. This extended algorithm would be remarkably useful at least in structural health monitoring, as tested structures in practice are often large linear systems subject to loads and excitations that are too difficult, expensive or impossible to measure. In addition, it would be of interest to investigate transfer-function based methods, as in [40], for assessing the global identifiability of linear systems.

The proposed observability algorithm in Chapter 3 can succeed in determining observable and k-row unobservable dynamic states, parameters and unmeasured inputs

of a rational nonlinear system. Theoretically concluding those k-row unobservable quantities to be unobservable, or maybe observable at a larger k, is a challenge problem, although a practical guideline has been given to do so with sufficient confidence in the examples. In a future extension of the work, a solid criterion will need to be derived, for which the algorithm can be stopped at a k where all the quantities have been determined as observable or unobservable.

All the observability and Lie symmetry studies in this thesis and most of the relevant literature are devoted to investigating continuous-time systems. Discrete-time systems, however, are also frequently encountered as in practice system outputs are measured and processed in discrete time. It is therefore highly demanded to develop robust algorithms for computing the observability and Lie symmetries of discrete-time nonlinear systems. The considered systems may be in general written in the following form:

$$\boldsymbol{x_k} = \boldsymbol{f}(\boldsymbol{x_{k-1}}, \boldsymbol{\theta_{k-1}}, \boldsymbol{u_{k-1}}, \boldsymbol{w_{k-1}}), \;\; \boldsymbol{\theta_k} = \boldsymbol{\theta_{k-1}}$$

$$\boldsymbol{y_k} = \boldsymbol{h}(\boldsymbol{x_k}, \boldsymbol{\theta_k}, \boldsymbol{u_k}, \boldsymbol{w_k})$$

where $\boldsymbol{x_k}$, $\boldsymbol{\theta_k}$, $\boldsymbol{u_k}$, $\boldsymbol{w_k}$ and $\boldsymbol{y_k}$ are respectively the dynamic states, unknown parameters, measured inputs, unmeasured inputs and output measurements at time step t_k. In the case of all inputs being measured, i.e. $\boldsymbol{w_k} = \emptyset$, the observability matrix of discrete-time nonlinear systems are given by:

$$\boldsymbol{\Omega} = \begin{bmatrix} \frac{\partial \boldsymbol{y_0}}{\partial \boldsymbol{x_0}} & \frac{\partial \boldsymbol{y_0}}{\partial \boldsymbol{\theta_0}} \\ \frac{\partial \boldsymbol{y_1}}{\partial \boldsymbol{x_0}} & \frac{\partial \boldsymbol{y_1}}{\partial \boldsymbol{\theta_0}} \\ \vdots & \vdots \\ \frac{\partial \boldsymbol{y_{n+l-1}}}{\partial \boldsymbol{x_0}} & \frac{\partial \boldsymbol{y_{n+l-1}}}{\partial \boldsymbol{\theta_0}} \end{bmatrix}$$

It would be important to further investigate the structure of the observability matrix of discrete-time systems with unmeasured inputs. The computations of both the matrices with and without unmeasured inputs are expected to have similar complexities

as those of continuous-time systems. Furthermore, it will be of interest to explore what relationship would be between the observability and symmetry properties of the continuous-time model and the discrete-time model of the same system.

Chapter 4 sketches basic ideas on the utilization of Lie symmetries for model reduction and sensor re-placement to improve system observability and identifiability. Developing a more systematic, standardized and automatic approach for applying Lie symmetries to achieve the model reduction will be a future research direction.

Appendix A

List of Publications

X. Shi, M.N. Chatzis and M.S. Williams. Robust computation of the observability of large linear systems with unknown parameters. Proceedings of the Sixth International Symposium on Life-Cycle Civil Engineering (IALCCE), Ghent, Belgium, 2018.

X. Shi, M.N. Chatzis, S.P. Triantafyllou and M.S. Williams. Observability and identification of damage-healing hysteretic model. Proceedings of the Seventh World Conference on Structural Control and Monitoring (7WCSCM), Qingdao, China, 2018.

X. Shi and M.N. Chatzis. Lie symmetries, observability and model transformation of nonlinear systems with unknown inputs. Proceedings of the XI International Conference on Structural Dynamics (Eurodyn), Athens, Greece, 2020.

X. Shi, M.N. Chatzis and M.S. Williams. A Robust Algorithm to Test the Observability of Large Linear Systems with Unknown Parameters. Mechanical Systems and Signal Processing, 2020, under review.

X. Shi, K. Maes and M.N. Chatzis. An Efficient Algorithm to Test the Observability

of Rational Nonlinear Systems with Unmeasured Inputs. Mechanical Systems and Signal Processing, 2020, manuscript to be submitted.

X. Shi and M.N. Chatzis. Lie Symmetries of Nonlinear Systems with Unmeasured Inputs. Mechanical Systems and Signal Processing, 2020, manuscript in preparation.

Appendix B

Conference Presentations

A robust algorithm to compute the observability of large linear systems with unknown parameters. Engineering Mechanics Institute Conference 2018, Massachusetts Institute of Technology, USA, May 2018.

Observability and identification of damage-healing hysteretic model. World Conference on Structural Control and Monitoring, Qingdao, China, July 2018.

Robust computation of the observability of large linear systems with unknown parameters. The Sixth International Symposium on Life-Cycle Civil Engineering, Ghent, Belgium, October 2018.

An efficient algorithm to test the observability of rational nonlinear systems with unmeasured inputs. Engineering Mechanics Institute Conference 2019, California Institute of Technology, USA, June 2019.

Lie symmetries of nonlinear systems with unmeasured inputs. The XI International Conference on Structural Dynamics, Athens, Greece, November 2020.

Bibliography

[1] P. Andersen. *Identification of Civil Engineering Structures using Vector ARMA Models*. PhD thesis, Department of Building Technology and Structural Engineering, Aalborg University, 1997.

[2] M. Anguelova. *Observability and identifiability of nonlinear systems with applications in biology*. PhD thesis, Department of Mathematical Sciences, Division of Mathematics, Chalmers University of Technology and Goteborg University, 2007.

[3] M. Anguelova, J. Karlsson, and M. Jirstrand. Minimal output sets for identifiability. *Mathematical Biosciences*, 239:139–153, 2012.

[4] Arcadis. Design and Consultancy for natural and built assets. `www.arcadis.com`, 2020. Accessed: 1.9.2020.

[5] S.K. Au, C.T. Ng, H.W. Sien, and H.Y. Chua. Modal identification of a suspension footbridge using free vibration signatures. *International Journal of Applied Mathematics and Mechanics*, 1:55–73, 2005.

[6] S.K. Au, F.L. Zhang, and Y.C. Ni. Bayesian operational modal analysis: Theory, computation, practice. *Computers and Structures*, 126:3–14, 2013.

[7] E. August and A. Papachristodoulou. A New Computational Tool for Establishing Model Parameter Identifiability. *Journal of Computational Biology*, 16(6):875–884, 2009.

[8] J.P. Barbot, D. Boutat, and T. Floquet. An observation algorithm for nonlinear systems with unknown inputs. *Automatica*, 45:1970–1974, 2009.

[9] G. Baumann. *Symmetry Analysis of Differential Equations with Mathematica*. Springer: New York, USA, 2000.

[10] G. Bellu, M.P. Saccomani, S. Audoly, and L. D'Angio. DAISY: A new software tool to test global identifiability of biological and physiological systems. *Computer Methods and Programs in Biomedicine*, 88:52–61, 2007.

[11] G.W. Bluman and S.C. Anco. *Symmetry and Integration Methods for Differential Equations*. Springer: New York, USA, 2002.

[12] G.W. Bluman and J.D. Cole. *Similarity Methods for Differential Equations*. Springer-Verlag: New York, USA, 1974.

[13] G.W. Bluman and S. Kumei. *Symmetries and Differential Equations*. Springer-Verlag: New York, USA, 1989.

[14] C. Bouillaguet and C. Delaplace. Sparse Gaussian Elimination Modulo p: an Update. *International Workshop on Computer Algebra in Scientific Computing*, pages 101–116, 2016.

[15] F. Boulier, D. Lazard, F. Ollivier, and M. Petitot. Computing representations for radicals of finitely generated differential ideals. *Applicable Algebra in Engineering, Communication and Computing*, 20:73–121, 2009.

[16] R.P. Brent and H.T. Kung. Fast algorithms for manipulating formal power series. *Journal of the Association for Computing Machinery*, 25(4):581–595, 1978.

[17] R. Brincker and C.E. Ventura. *Introduction to Operational Modal Analysis*. John Wiley and Sons, 2015.

[18] R. Brincker, C.E. Ventura, and P. Andersen. Damping Estimation by Frequency Domain Decomposition. *Proceedings of IMAC 19: A Conference on Structural Dynamics, Orlando, USA*, 1:698–703, 2001.

[19] R. Brincker, L. Zhang, and P. Andersen. Modal identification of output-only systems using frequency domain decomposition. *Smart Materials and Structures*, 10:441–445, 2001.

[20] D.L. Brown, R.J. Allemang, R. Zimmerman, and M. Mergeay. Parameter Estimation Techniques for Modal Analysis. *SAE Transactions*, 88:828–846, 1979.

[21] J.M.W Brownjohn. Structural health monitoring of civil infrastructure. *Philosophical Transactions of the Royal Society*, 365:589–622, 2006.

[22] B.J. Cantwell. *Introduction to Symmetry Analysis*. Cambridge University Press, Cambridge, UK, 2002.

[23] E.N. Chatzi and A.W. Smyth. The Unscented Kalman filter and particle filter methods for nonlinear structural system identification with non-collocated heterogeneous sensing. *Structural Control and Health Monitoring*, 16(1):99–123, 2009.

[24] M.N. Chatzis and E.N. Chatzi. A discontinuous unscented Kalman filter for non-smooth dynamic problems. *Frontiers in Built Environment*, 3(56), 2017.

[25] M.N. Chatzis, E.N. Chatzi, and A.W. Smyth. An experimental validation of time domain system identification methods with fusion of heterogeneous data. *Earthquake Engineering and Structural Dynamics*, 44:523–547, 2015.

[26] M.N. Chatzis, E.N. Chatzi, and A.W. Smyth. On the observability and identifiability of nonlinear structural and mechanical systems. *Structural Control and Health Monitoring*, 22:574–593, 2015.

[27] M.N. Chatzis, E.N. Chatzi, and S.P. Triantafyllou. A discontinuous extended Kalman filter for non-smooth dynamic problems. *Journal of Mechanical Systems and Signal Processing*, 92:13–29, 2017.

[28] G. Chavent. Identification of Distributed Parameter Systems: About the Output Least Square Method, its Implementation, and Identifiability. *IFAC Proceedings Volumes*, 12(8):85–97, 1979.

[29] E.W.V. Chaves. *Notes on continuum mechanics*. Springer Netherlands, 2013.

[30] A.K. Chopra. *Dynamics of Structures: Theory and Applications to Earthquake Engineering*. Pearson Education, 2012.

[31] B De Jager. The use of symbolic computation in nonlinear control: is it viable? *IEEE Transactions on Automatic Control*, 40(1):84–89, 1995.

[32] A. De Sortis, E. Antonacci, and F. Vestroni. Dynamic identification of a masonry building using forced vibration tests. *Engineering Structures*, 27:155–165, 2005.

[33] V. Dertimanis, E. Chatzi, and S. Eftekhar Azam. Input-state-parameter estimation of structural systems from limited output information. *Mechanical Systems and Signal Processing*, 126:711–746, 2019.

[34] S.X. Ding. *Model-Based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*. Springer - Verlag Berlin Heidelberg, 2008.

[35] S. Diop and M. Fliess. On Nonlinear Observability. *Proceedings of the* 1^{st} *European Control Conference, Grenoble, France*, 1:152–157, 1991.

[36] S. Eftekhar Azam, V.K. Dertimanis, E.N. Chatzi, and C. Papadimitriou. Output-only schemes for joint input-state-parameter estimation of linear systems. *UNCECOMP 2015 -* 1^{st} *ECCOMAS Thematic Conference on Uncertainty Quantification in Computational Sciences and Engineering, Crete, Greece*, pages 497–510, 2015.

[37] D.J. Ewins and P.T. Gleeson. A method for modal identification of lightly damped structures. *Journal of Sound and Vibration*, 84:57–59, 1982.

[38] C.R. Farrar and K. Worden. An introduction to structural health monitoring. *Philosophical Transactions of The Royal Society A*, 365:303–315, 2007.

[39] M.R. Fernandez, J.A. Egea, and J.R. Banga. Novel metaheuristic for parameter estimation in nonlinear dynamic biological systems. *BMC Bioinformatics*, 7:483–500, 2006.

[40] G. Franco, R. Betti, and R.W. Longman. On the uniqueness of solutions for the identification of linear structural systems. *Journal of Applied Mechanics*, 73(1):153–162, 2005.

[41] D.M. Frangopol and M. Liu. Maintenance and management of civil infrastructure based on condition, safety, optimization, and life-cycle cost. *Structure and Infrastructure Engineering*, 3(1):29–41, 2007.

[42] B.R. Frieden. *Science from Fisher Information*. Cambridge University Press, Cambridge, UK, 2004.

[43] S. Gillijns and B. De Moor. Unbiased minimum-variance input and state estimation for linear discrete-time systems with direct feedthrough. *Automatica*, 43:934–937, 2007.

[44] R. Hermann and A.J. Krener. Nonlinear Controllability and Observability. *IEEE Transactions on Automatic Control*, AC-22(5):728–740, 1977.

[45] M. Hoshiya and E. Saito. Structural Identification by Extended Kalman Filter. *Journal of Engineering Mechanics*, 110(12):1757–1770, 1984.

[46] E. Hubert and A. Sedoglavic. Polynomial Time Nondimensionalisation of Ordinary Differential Equations via Their Lie point Symmetries. *HAL Archive*, pages 1–17, 2006.

[47] N.H. Ibragimob. *Handbook of Lie Group Analysis of Differential Equations*. CRC Press: Boca Raton, FL, USA, 1994.

[48] N.H. Ibragimov. *Transformation Groups Applied to Mathematical Physics*. Reidel Publishing Company: Dordrecht, the Netherlands, 1985.

[49] S.R. Ibrahim and E.C. Mikulcik. A method for the direct identification of vibration parameters from the free response. *Shock and Vibration Bulletin*, 4:183–198, 1977.

[50] A. Isidori. *Nonlinear Control Systems*. Springer-Verlag London, 1995.

[51] N.J. Jacobsen, P. Andersen, and R. Brincker. Using Enhanced Frequency Domain Decomposition as a Robust Technique to Harmonic Excitation in Operational Modal Analysis. *Proceedings of ISMA2006: International Conference on Noise and Vibration Engineering*, 1(1):2–13, 2006.

[52] S.J. Julier and J.K. Uhlmann. New extension of the Kalman filter to nonlinear systems. *Proceedings of Signal Processing, Sensor Fusion, and Target Recognition VI*, 3068:182–193, 1997.

[53] S.J. Julier and J.K. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92:401–422, 2004.

[54] R.E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(D):35–45, 1960.

[55] R.E. Kalman. On the General Theory of Control Systems. *Proceedings of the 1st International IFAC Congress on Automatic and Remote Control, Moscow, USSR*, 1:491–502, 1960.

[56] J. Karlsson, M. Anguelova, and M. Jirstrand. An efficient method for structural identifiability analysis of large dynamic systems. *Proceedings of the 16th International Federation of Automatic control, Brussels, Belgium*, 45:941–946, 2012.

[57] L.S. Katafygiotis and J.L. Beck. Updating models and their uncertainties. II: model identifiability. *Journal of Engineering Mechanics*, 124:463–467, 1998.

[58] J. Kirshenboim and D.J. Ewins. Method for the Derivation of Optimal Modal Parameters From Several Single-Point Excitation Tests. *Proceedings of the International Modal Analysis Conference*, 2:991–997, 1984.

[59] A. Kugi, K. Schlacher, and R. Novak. Symbolic computation for the analysis and synthesis of nonlinear control systems. *WIT Press Transactions on Engineering Sciences*, 22:255–264, 1999.

[60] J. Lardies and S. Gouttebroze. Identification of modal parameters using the wavelet transform. *International Journal of Mechanical Sciences*, 44(11):2263–2283, 2002.

[61] T.P. Le and P. Argoul. Continuous wavelet transform for modal identification using free decay response. *Journal of Sound and Vibration*, 277:73–100, 2004.

[62] Y.Y. Liu, J.J. Slotine, and A.L. Barabasi. Observability of complex systems. *Proceedings of the National Academy of Sciences of the United States of America*, 110(7):2460–2465, 2013.

[63] L. Ljung. *Theory and Practice of Recursive Identification*. MIT Press, Massachusetts, USA, 1983.

[64] L. Ljung. *System identification: theory for the user*. Prentice Hall, Division of Simon and Schuster One Lake Street Upper Saddle River, NJ, USA, 1986.

[65] L. Ljung and T. Glad. On global identifiability for arbitrary model parametrizations. *Automatica*, 30:265–276, 1994.

[66] E. Lourens, E. Reynders, G. De Roeck, G. Degrande, and G. Lombaert. An aug-

mented Kalman filter for force identification in structural dynamics. *Mechanical Systems and Signal Processing*, 27:446–460, 2012.

[67] K. Maes, M.N. Chatzis, and G. Lombaert. Observability of nonlinear systems with unmeasured inputs. *Mechanical Systems and Signal Processing*, 130:378–394, 2019.

[68] K. Maes, S. Gillijns, and G. Lombaert. A smoothing algorithm for joint input-state estimation in structural dynamics. *Mechanical Systems and Signal Processing*, 98:292–309, 2018.

[69] Maple. *Maplesoft, a division of Waterloo Maple Inc.* Maplesoft, Waterloo, Ontario, 2019.

[70] A. Martinelli. Extension of the observability rank condition to nonlinear systems driven by unknown inputs. *Proceedings of the 23rd Mediterranean Conference on Control and Automation (MED), Torremolinos, Spain*, pages 589–595, 2015.

[71] MATLAB. *version R2016a*. The MathWorks Inc., Natick, Massachusetts, 2016.

[72] S.V. Meleshko. *Methods for Constructing Exact Solutions of Partial Differential Equations*. Springer: New York, USA, 2005.

[73] B. Merkt, J. Timmer, and D. Kaschek. High-order Lie symmetries in identifiability and predictability analysis of dynamic models. *Physical Review E*, 92:1–9, 2015.

[74] H. Miao, X. Xia, A.S. Perelson, and H. Wu. On Identifiability of Nonlinear ODE Models and Applications in Viral Dynamics. *SIAM Rev. Soc. Ind. Appl. Math.*, 53(1):3–39, 2011.

[75] S. Mukhopadhyay, H. Lus, and R. Betti. Structural identification with incomplete instrumentation and global identifiability requirements under base excitation. *Structural Control and Health Monitoring*, 22:1024–1047, 2015.

[76] F. Oliveri. Lie Symmetries of Differential Equations: Classical Results and Recent Contributions. *Symmetry*, 2:658–706, 2010.

[77] P.J. Olver. *Applications of Lie Groups to Differential Equations*. Springer: New York, USA, 1986.

[78] P.J. Olver. *Equivalence, Invariants, and Symmetry*. Cambridge University Press: Cambridge, UK, 1995.

[79] L.V. Ovsiannikov. *Group Analysis of Differential Equations*. Academic Press: New York, USA, 1982.

[80] C. Papadimitriou and G. Lombaert. The effect of prediction error correlation on optimal sensor placement in structural dynamics. *Mechanical Systems and Signal Processing*, 28:105–127, 2012.

[81] H. Pohjanpalo. System Identifiability Based on the Power Series Expansion of the Solution. *Mathematical Biosciences*, 41:21–33, 1978.

[82] L.B. Rall. *Computational Solution of Nonlinear Operator Equations*. John Wiley & Sons, Inc., New York, 1969.

[83] A. Raue, C. Kreutz, T. Maiwald, J. Bachmann, M. Schilling, U. Klingmuller, and J. Timmer. Structural and practical identifiability analysis of partially observed dynamical models by exploiting the profile likelihood. *Bioinformatics*, 25(15):1923–1929, 2009.

[84] M. Ruzzene, A. Fasana, L. Garibaldi, and B. Piombo. Natural frequencies and dampings identification using wavelet transform: application to real data. *Mechanical Systems and Signal Processing*, 11(2):207–218, 1997.

[85] S.F. Schmidt. Application of state-space methods to navigation problems. *Advances in Control Systems*, 3:293–340, 1966.

[86] A. Sedoglavic. A Probabilistic Algorithm to Test Local Algebraic Observability in Polynomial Time. *Journal of Symbolic Computation*, 33:735–755, 2002.

[87] X. Shi and M.N. Chatzis. Lie symmetries, observability and model transformation of nonlinear systems with unknown inputs. *Proceedings of the XI International Conference on Structural Dynamics (Eurodyn2020), Athens, Greece*, 2020.

[88] X. Shi, M.N. Chatzis, S.P. Triantafyllou, and M.S. Williams. Observability and identification of damage-healing hysteretic model. *Proceedings of the Seventh World Conference on Structural Control and Monitoring (7WCSCM), Qingdao, China*, 2018.

[89] X. Shi, M.N. Chatzis, and M.S. Williams. Robust computation of the observability of large linear systems with unknown parameters. *Proceedings of the Sixth International Symposium on Life-Cycle Civil Engineering (IALCCE), Ghent, Belgium*, 2018.

[90] V. Shoup. *A Computational Introduction to Number Theory and Algebra*. Cambridge University Press, 2008.

[91] I. Shparlinski. *Finite fields: theory and computation: the meeting point of*

number theory, computer science, coding theory and cryptography. Springer Netherlands, 1999.

[92] D. Simon. *Optimal State Estimation: Kalman, H∞, and Nonlinear Approaches*. John Wiley & Sons, 2006.

[93] E.D. Sontag and Y. Wang. I/O Equations for Nonlinear Systems and Observation Spaces. *Proceedings of the 30th Conference on Decision and Control, Brighton, England*, pages 720–725, 1991.

[94] M. Spivak. *Calculus on Manifolds: A Modern Approach to Classical Theorems of Advanced Calculus*. Avalon Publishing, 1965.

[95] H. Stephani. *Differential Equations: Their Solutions Using Symmetries*. Cambridge University Press: Cambridge, UK, 1989.

[96] H.J. Sussmann. Single-input Observability of Continuous-time Systems. *Math. Systems Theory*, 12:371–393, 1979.

[97] S.P. Triantafyllou and E.N. Chatzi. A hysteretic multiscale formulation for nonlinear dynamic analysis of composite materials. *Computational Mechanics*, 54(3):763–787, 2014.

[98] S.P. Triantafyllou and M.N. Chatzis. A new damage-healing smooth hysteretic formulation for the modelling of self-healing materials. *Proceedings of the 8th GRACM International Congress on Computational Mechanics*, 2015.

[99] S.P. Triantafyllou and V.K. Koumousis. Hysteretic finite elements for the nonlinear static and dynamic analysis of structures. *Journal of Engineering Mechanics*, 140(6):04014025, 2014.

[100] F.E. Udwadia and D.K. Sharma. Some uniqueness results related to building structural identification. *SIAM Journal on Applied Mathematics*, 34(1):104–118, 1978.

[101] F.E. Udwadia, D.K. Sharma, and P.C. Shah. Uniqueness of damping and stiffness distributions in the identification of soil and structural systems. *Journal of Applied Mechanics*, 45(1):181–187, 1978.

[102] A. Urguplu. *Contributions to Symbolic Effective Qualitative Analysis of Dynamical Systems; Application to Biochemical Reaction Networks*. PhD thesis, Department of Computer Science, University of Lille, 2010.

[103] S. Vajda, K.R. Godfrey, and H. Rabitz. Similarity transformation approach to identifiability analysis of nonlinear compartemental models. *Mathematical Biosciences*, 93:217–248, 1989.

[104] H. Van der Auweraer and J. Leuridan. Multiple input orthogonal polynomial parameter estimation. *Mechanical Systems and Signal Processing*, 1:259–272, 1987.

[105] P. Van Overschee and B. De Moor. Subspace algorithms for the stochastic identification problem. *Automatica*, 29:649–660, 1993.

[106] P. Van Overschee and B. De Moor. N4SID: Subspace Algorithms for the Identification of Combined Deterministic-Stochastic Systems. *Automatica*, 30(1):75–93, 1994.

[107] P. Van Overschee and B. De Moor. *Subspace Identification for Linear Systems:*

Theory, Implementation, Applications. Kluwer Academic Publishers Group, 1996.

[108] P. Verboven. *Frequency Domain System Identification for Modal Analysis*. PhD thesis, VRIJE UNIVERSITEIT BRUSSEL, 2002.

[109] A.F. Villaverde, A. Barreiro, and A. Papachristodoulou. Structural Identifiability of Dynamic Systems Biology Models. *PLOS Computational Biology*, 12(10):1–22, 2016.

[110] Wolfram. *Mathematica*. Wolfram Research Inc., Champaign, Illinois, 2019.

[111] J.N. Yang, H. Huang, and S. Lin. Sequential nonlinear least square estimation for damage identification of structures. *International Journal of Nonlinear Mechanics*, 41(1):124–140, 2006.

[112] T. Yoneyama and Y. Fujii. Fabrication and erection of Tokyo Gate Bridge. *IABSE-JSCE Joint Conference on Advances in Bridge Engineering-III*, pages 268–277, 2015.

[113] F.L. Zhang, Y.C. Ni, S.K. Au, and H.F. Lam. Fast Bayesian approach for modal identification using free vibration data, Part I - Most probable value. *Mechanical Systems and Signal Processing*, 70-71:209–220, 2016.

[114] Y.C. Zhu. *Bayesian Modal Identification using Asynchronous Ambient Data*. PhD thesis, Department of Civil Engineering, University of Liverpool, 2018.