On the Localisation and Reconstruction of Structural Model Errors in Dynamic Systems

Dissertation
zur
Erlangung des Doktorgrades (Dr. rer. nat.)
der
Mathematisch-Naturwissenschaftlichen Fakultät
der
Rheinischen Friedrich-Wilhelms-Universität Bonn

von
Dominik Thilo Kahl
aus
Recklinghausen

Bonn, 06.11.2020

Angefertigt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Rheinischen Friedrich-Wilhelms-Universität Bonn.

1. Gutachter:	Prof. Dr. Reinhard Klein
2. Gutachter:	Prof. Dr. Maik Kschischo

Tag der Promotion:	02.07.2021
Erscheinungsjahr:	2021

Darum versenkt, wer im ungeschlichteten Zwist der Völker nach geistiger Ruhe strebt, gern den Blick in das stille Leben der Pflanzen und in der heiligen Naturkraft inneres Wirken; oder, hingegeben dem angestammten Triebe, der seit Jahrtausenden der Menschen Brust durchglüht, blickt er ahndungsvoll aufwärts zu den hohen Gestirnen, welche in ungestörtem Einklang die alte, ewige Bahn vollenden.

Alexander von Humboldt, Ansichten der Natur [1]

Abstract

Pursuing a deeper understanding of the behaviour of dynamic systems is a main task of modern sciences, from accurate weather forecasts to precise drug delivery, from electrical engineering to cell biology and the very fundamental laws of nature. Spurious knowledge or simply the lack of deeper physical insights lead to *structural model errors*, which cause discrepancies between the predictions from the theoretical model and the experimental data.

Significance of SEEDS Dynamic systems serve as models for systems or processes that evolve over time. To get a better understanding of the dynamic system of interest, a precise mathematical model is crucial inasmuch as erroneous models lead to false conclusions about the current state of the system, about the interactions within the system, and about its future development. In this thesis we offer a methodology for *Structural Error Estimation in Dynamic Systems* as a solution to this problem. As in the era of big data, classical procedures become infeasible, a theoretical framework is presented for the *Model Error Reconstruction Problem*: A data-driven approach to compute estimates for structural model errors. By this we quantify the model error, achieve a numerical description, and correct state estimates.

Results We exploit analytical, algebraic, and structural properties of dynamic systems and formulate conditions for the *invertibility of a dynamic system*. Invertibility guarantees a unique model error reconstruction. We subsequently investigate the network principles that favour the error reconstruction. We unveil the intrinsic *gammoid structure* of dynamic input-output systems and show that *invariable sparsity* is a well appropriate regularisation concept for non-invertible systems. In addition a new coherence measure will be introduced that makes a statement about the distinguishability of model error sources and about the redundancy of sensors. Finally, a LASSO-type optimisation technique will be presented as a variety of Dynamic Compressed Sensing. The resulting optimisation problem is capable of producing invariable sparse solutions.

Preface

The curiosity and strife to understand the world has inspired scientists throughout the ages to bring order into the vast complexity of phenomena of nature, subject to the zeitgeist and technical opportunities of their time. The works on theoretical information science and the development of semiconductor devices lead mankind into the *Digital Age.* Within seconds computers solve numerical problems which would have been inconceivable one hundred years ago. Exploiting the arising abilities and to enrich the sciences with computational methods in order to answer old and new questions, therefore, is the duty of our generation.

> *[...] so ging allen Naturforschern ein Licht auf. Sie begriffen, daß die Vernunft nur das einsieht, was sie selbst nach ihrem Entwurfe hervorbingt, daß sie mit Prinzipien ihrer Urteile nach beständigen Gesetzen vorangehen und die Natur nötigen müsse, auf ihre Fragen zu antworten [...] denn sonst hängen zufällige, nach keinem vorher entworfenen Plane gemachten Beobachtungen gar nicht in einem notwendigen Gesetze zusammen, welches doch die Vernunft sucht und bedarf.* [*] [3]

In the light of current research in the fields of *machine learning* in its broadest sense, Kant's *Critique of Pure Reason* seems to manifest as a Critique of Pure Data Science. The possibility to process *big data* enabled the development of a purely data driven methodology of forecasting. Such a methodology stands in contrast to the classical way of formulating theories together with mathematical models to describe the phenomena one aims to understand. As the classical scientist did not have the chance to make use of modern computers, it seems just fair to glue together the fragments of analytical understanding by modern methods.

But any purely data driven approach can only make statements that are limited to the given data and have no power to contradict other possible outcomes. Inasmuch as a trustworthy prediction needs rules of general validity, a *synthetic reasoning,* that is, the theoretical hence generally valid models tested and verified by experimental data, is the way any nature sciences shall follow. For the last centuries, the human mind and ability to observe his surroundings have been the main tools for this process. But as the data get big, bigger than a human mind can oversee, new approaches are needed. Ideas such as *causal machines* [4] show that this problem is indeed a subject of current discussion and may provide computational methods for this purpose in the future.

At the end we always have to face the same task: We need to bring together generally valid theories and apodictically true experimental results. Mathematical models, which are crucial for a quantitative theory, build on our limited understanding of its fundamental processes. Usually these models are only valid under certain simplifying conditions or in certain regimes and consequently

[*] [...] a light dawned on all those who study nature. They comprehended that reason has insight only into what it itself produces according to its own design; that it must take the lead with principles for its judgements according to constant laws and compel nature to answer its questions [...] for otherwise accidental observations, made according to no previous designed plan, can never connect up into a necessary law, which is yet what reason seeks and requires. [2]

show a discrepancy when compared to experimental data. These discrepancies may be negligible, or may be severe. This problematic results in what will be called a *structural model error*. So the following questions arise: When we observe such a discrepancy between theory and experiment, can there be a consistent treatment for structural model errors? Can we *localise* where the model is erroneous and can we quantify the error? Is it possible to *reconstruct* the model error?

> The aim of this thesis is to develop a theoretical framework which helps to localise and reconstruct such structural model errors that are inherent in many dynamic systems.

Starting with an introduction to the idea of *structural model errors*, the model class of *dynamic systems* is introduced in order to formulate the *Model Error Reconstruction Problem* and the key property of *invertibility*. For the characterisation of invertibility we exploit *state space*, *differential-algebraic*, as well as *structural* methods, which, respectively, lead to a set of (almost) equivalent conditions. The structural version will be used to investigate the *network principles* of invertibility. With the rigorous conditions as preliminaries, we turn our interest towards the practical problem of *invariable sparse error reconstruction*. Motivated by the classical idea of sparse and compressed sensing, we proof that the Model Error Reconstruction Problem gives rise to an intrinsic *gammoid* that allows to formulate conditions for the reconstruction of *invariable sparse* model errors. In order to compute estimates for such invariable sparse model errors, we introduce a variety of *dynamic compressed sensing* which is consistent with the new concept of invariable sparsity in form of a LASSO-type regularised dynamic optimisation problem. To bring the theory to a close, we derive several applications of the developed theory. Conditions for the concrete cases of *fault detection* in a network are provided, a *model correction* for an erroneous model of the Lorenz system is presented, and a finally a *clustering approach* is introduced which identifies indistinguishable model errors as well as redundant sensors.

This thesis is part of the SEEDS project, funded by *Deutsche Forschungsgemeinschaft* under project number 354645666. Earlier publications [5] and [6] already presented intermediate results of this thesis, furthermore an `R` implementation of a Model Error Reconstruction Algorithm was published [7] and provided via *CRAN* [8]; resemblances of the present thesis with these publications is consequential.

Contents

1 Structural Model Errors Emerge **1**
1.1 Dynamic Systems 4
1.2 Quantifying Model Errors 6
1.2.1 Exogenous Model Errors 7
1.2.2 Endogenous Model Errors 8
1.2.3 Endogenous or Exogenous? 9
1.3 Reconstruction of Model Errors 10
1.3.1 The Model Error Reconstruction Problem 11
1.3.2 Significance of SEEDS 11
1.3.3 Example 12
1.3.4 Invertibility of a Dynamic System 14

2 Differential-Algebraic Invertibility **17**
2.1 D-Algebra for Dynamic Systems 18
2.1.1 d-Rings and d-Fields and d-Ideals 19
2.1.2 From State Space to d-Algebra 23
2.2 Ordinary d-Algebraic Systems 24
2.2.1 Ordinary System 24
2.2.2 Generating Systems 27
2.2.3 Differential-Transcendence Degree of Ordinary Systems 30
2.3 Differential-Algebraic Invertibility 34

3 Structural Invertibility **37**
3.1 Structural Systems 38
3.2 Necessity of Structural Invertibility 40
3.3 Sufficiency of Structural Invertibility 41
3.4 A Note on the Connection between State Space, d-Algebraic, and Structural Systems 46

4 Invertibility Structures in Complex Networks **47**
4.1 Random Networks 48
4.1.1 Erdős-Rényi Network 48
4.1.2 Scale-Free Networks 49
4.1.3 Structural Invertibility Algorithm 50
4.2 Scenario I 51
4.2.1 Random Networks 52
4.2.2 Real Networks 53

4.3 Scenario II 54
4.3.1 The Role of Hubs 54
4.3.2 Sensor Node Placement Algorithm 55
4.3.3 Application to Real Networks 57
4.4 Scenario III 58
4.4.1 Asymmetric Node Degrees 58
4.4.2 Full Experimental Design 60
4.5 Principles of Structural Invertibility 60

5 Sparse Error Reconstruction 65
5.1 Error Reconstruction Example 65
5.2 Localisability of Model Errors 66
5.2.1 Input Localisation 68
5.2.2 Concepts of Sparse Sensing 68
5.3 Gammoid Interpretation of Structural Invertibility 69
5.3.1 Matroids 70
5.3.2 The Gammoid Structure of Dynamic Input-Output Systems 73
5.3.3 Uniqueness of the Sparsest Solution 78
5.4 Coherence of Input Nodes 79
5.4.1 Weighted Gammoids and the Transfer Function 81
5.4.2 Concatenation of Gammoids 83
5.4.3 Gramian 87
5.5 Spark Estimation via the Mutual Coherence 91
5.5.1 Eigenvalues of the Gramian 91
5.5.2 Strict Diagonal Dominance for Invertibility 93
5.5.3 Estimation of the Spark 99
5.5.4 Feed-Forward Graphs 100
5.6 Overview: Gammoids for Sparse Sensing of Model Errors 101

6 Dynamic Compressed Sensing 103
6.1 Spaces and Norms 106
6.1.1 Input Space 107
6.1.2 Output Space 112
6.2 Convex Optimisation 113
6.3 Restricted Isometry Property 113
6.4 Overview: Dynamic Compressed Sensing of Model Errors 123

7 Applications 125
7.1 Fault Detection 127
7.2 The Lorenz System 130
7.3 Cluster Approach 134
7.3.1 Caenorhabditis Elegans 134
7.3.2 Iterative Error Detection 137
7.3.3 Summary 139

8 A Concluding View on SEEDS 141
8.1 Result: A Theoretical Framework 142
8.1.1 How to: Model Error Reconstruction 143
8.2 Perspectives for Future Research 144
8.2.1 Non-linear Optimisation 144
8.2.2 Invertibility of Partial Differential Systems 145
8.2.3 Model Discovery 147

List of Figures 161

List of Tables 165

CHAPTER 1

Structural Model Errors Emerge

The *dynamic systems* we need to handle today, be it an ecosystem, a biochemical system like a living cell, mechanical, electric or even economical systems, are growing bigger, more complex and more interconnected than they have ever been before; see for instance [9, 10, 11, 12] for discussions about realistic modelling of biological systems and [13, 14, 15, 16] for databases about systems of current interest. The year this thesis is submitted, the *corona virus* brought the century's first pandemic. A minute virus that affects a human cell, spreads rapidly - especially by the help of air travel - all over the world, and causes inestimable economical losses everywhere. Given that, one must accept that a microscopic perturbation can force a giant system into entirely new and unpredictable regimes.

As the dynamic systems grow complex, our *mathematical models* must grow complex as well. But it seems that the current tendency is to go *big* rather then *complex*. Big models may consist of hundreds of equations and one might have a huge amount of data available to find the parameters of such a model. But complexity means that it takes more than a combination of simple mathematical functions to grasp all processes within system. Accepting complexity unavoidably means that the mathematical model we work with is merely a good guess about how things could work.

A Forest in Switzerland

Let us discuss the ecosystem forest as a vivid example. In a study about the reintroduction of the lynx in the Swiss Jura Forest in the 1970s, among many results, two plausible observations were made. In the years after the reintroduction of the lynx, the roe population decreased locally to almost extinction [17, 18]. The theory was formulated that the roe specimen were not used to predators. Later, the roes adapted to this new situation and learned to be more cautious [19]. As a consequence, the grown lynx population had to face starvation, they expanded their *home-range*, migrated to regions with better food supply, or changed their diet [20].

Assume the lucky case that we had an accurate mathematical model for a single lynx-roe interaction and one model that accurately describes the whole forest before the lynx was brought in. Still, a combination of these two models would not be capable of describing the new situation as the combination of the two systems changes the behaviour of the fundamental components, in this case, the animals. The models which have been accurate in isolation cease to be accurate in combination. There is a discrepancy between the naive combination of the two models and the unknown complex model that would correctly describe the merge of the systems. We will call this discrepancy an *endogenous model error*.

One plausible consequence of the shrunken roe population was the migration of the lynx. Assume we have an accurate model of the whole forest, including the lynx. As an exogenous observer, we might be able to define the boundaries of our forest more or less stringently. However, these are artificial mathematical boundaries of our model and will obviously not stop the starving animals from migrating into a region with better food supply. The forest is an *open system*, equation systems in their common formulation are *mathematically isolated*, hence are not capable of taking migration processes into account. How could an isolated model be able to depict an open system? A holistic model of the forest must respect its own insufficiency to describe anything beyond its boundaries. The discrepancies between the closed model and an accurate open model that arise from this issue will be called *exogenous model errors*. See also [21, 22, 6, 5] where the idea of endogenous and exogenous model errors has been discussed.

The Discovery of Uranus and Neptune

To formulate a theory, compare it with observations to detect inadequacies of the model, and develop a new, consistent theory, is the cycle of every nature science. One noteworthy example is the discovery of the planets Uranus and Neptune, see [23]:

In the year 1687 Isaac Newton published his famous theory for the gravitational attraction of planets, see [24] for a contemporary translation, and made it possible to predict their orbits on the basis of a generally valid law. In 1781, William Herschel discovered the planet Uranus. The observations of Uranus' orbit, however, did not match the theoretical predictions from Newton's law. Astronomers stated the hypothesis that another heavy planet might be the cause for Uranus' distraction. Urbain Le Verrier and others used the observation data to calculate the position of this new planet in 1846 and in the same year, Johann Gottfried Galle discovered the planet Neptune at the predicted location.

In this story, the astronomers started with a widely accepted, simple model, Newton's law of gravitation. But Uranus' movement around the sun is not a closed system. Neptune caused a perturbation that came from outside of the system boundaries, that is, Neptune caused an external model error. Le Verrier estimated this model error quantitatively and gave hints for where and when to make new observations so that Galle finally found the unknown planet that seemed to *perturb* Uranus' movement. This interplay of theoretical predictions and experimental observations enabled the astronomers to extend the model of the solar system.

What Model Errors Can Teach Us The emergence of model errors seems to be unavoidable once the system of investigations becomes too complex to fully understand it in theory or isolate it in practice. Figure 1.1 depicts the emergence of model errors. Subfigure (a) shows a graphical representation of an exemplary system, inspired by [21]. The blue-shaded region represents our knowledge about a dynamic system we want to investigate. This subsystem is embedded into a larger, complex system shaded in grey. The black dots represent the *entities*, and the arrows represent the *interactions* between them.

As in the example of the Swiss forest, we might know all the entities (roe and lynx) but we do not know how the interaction between them will manifest in reality. The green arrow shows an interaction between two known entities, but the interaction itself lies beyond the borders of the

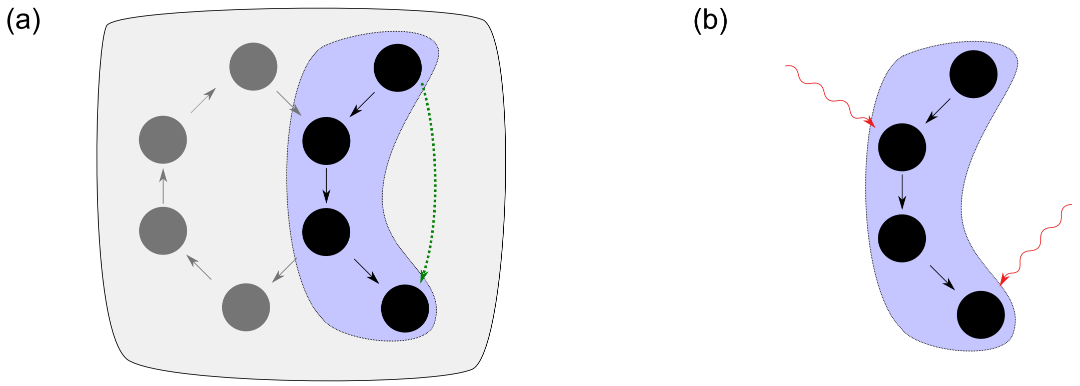

Figure 1.1: A visualisation of structural model errors. (a) A known subsystem (blue shaded region) is embedded into a larger unknown exo-system (grey shaded region). The subsystem interacts with the exo-system as indicated by the arrows. In addition, there exists one interaction (green dotted arrow) between two elements of the subsystem that leaves the known (blue) region. (b) A reformulation with model errors (red wiggly arrows). The interaction with the exo-system leads to an exogenous model error, the unknown interaction within the model leads to an endogenous model error.

known sub-system. As in the example of Uranus and Neptune, there might be influences from outside the borders of the known world. What we actually *see* is shown in subfigure (b): A system which is disturbed by *perturbations of unknown origin*, as represented by the wiggly arrows.

Aim of this Thesis

Model errors lead to false predictions and to a wrong estimation of the current state of the system. We call the endogenous and exogenous errors *structural* for they unveil inadequacies of the model that cannot be remedied by adjusting its parameters but which are related to the very structure of the system, meaning the existence and functional form of its interactions. As the systems of interest become complex and data become big, a classical workflow like in the story of Neptune and Uranus becomes infeasible.

In seminal papers Engelhardt et al. presented algorithms motivated by optimal control problems in order to estimate model errors [21, 22]. But these algorithms were heuristic and they do not give satisfaction to the need to investigate further into this issue. Until now, it lacks a theoretical understanding, if it is at all possible to estimate model errors, how much or which kind of data we need, and which algorithms are adequate to compute good estimates. The following chapters shall help to illuminate the problem of *Structural Error Estimation in Dynamic Systems (SEEDS).*

Conditions for Invertibility The possibility to infer model errors from given data will be treated under the name *invertibility*. We will search for necessary and sufficient conditions for the invertibility of a dynamic system.

Impact of the Network Structure It will be investigated, whether and how the *network structure* of a system can make a statement about its invertibility. We discuss scenarios for an *experimental design* which respects the invertibility conditions.

Localisability If we are not able to infer a quantitative description of the model error, we might still be able to *localise* with best precision within the framework of the given data. Intrinsic *independence* and *coherence* structures of a dynamic system are presented and utilised to learn more about the localisability of *invariable sparse model errors.*

Recovery Algorithm In order to give a practical algorithm that is capable of producing estimates for model errors, an optimisation problem is discussed, which can be seen as a variation of the LASSO-regularisation appropriate for erroneous dynamic systems.

1.1 Dynamic Systems

Having discussed the natural emergence of structural model errors, the first step must be a mathematically comprehensible characterisation of model errors in dynamic systems. In its broadest and abstract form, a dynamic system can be any set of *entities*, that *interact* according to specific, fundamental laws, giving rise to a development of the whole system over *time.* Though one could think of various ways to mathematically model a dynamic system, the formulation presented below is the standard form which is usually found in the literature, see for instance [25] for one of many textbooks on dynamic systems.

Time The main characteristic that renders a system dynamic is its time-development.

> Time is the spacial dimension that has only one direction. [*]

Mathematically, **time** can be characterised as the solution of the initial value problem

$$\frac{\mathrm{d}}{\mathrm{d}\tau}t(\tau) = 1 \quad , \quad t(0) = t_0 \,. \tag{1.1}$$

This defines the **time-domain** $\mathcal{T} \subseteq \mathbb{R}$ of a dynamic system. Without loss of generality the time domain is assumed to be

$$\mathcal{T} = [0, T] \,. \tag{1.2}$$

Our intuitive interpretation of time in contrast to any spacial coordinate, however, serves well for a vivid understanding of the time course of a system.

Space The current state of the i^{th} entity at a certain point in time $t \in [0, T]$ can usually be described by a real number $x_i(t) \in \mathbb{R}$. We call x_i the **state variable** of i. Say, the system comprises N entities, then the vector

$$\boldsymbol{x}(t) = \begin{pmatrix} x_1(t) \\ \vdots \\ x_N(t) \end{pmatrix} \tag{1.3}$$

represents the **state** of the whole system at time t. The **initial state** $\boldsymbol{x}_0$ of the system is the state of the system at the first point in time $\boldsymbol{x}(0)$. As there is by construction no time-development before $t = 0$, the initial state must be one given parameter in the formulation of a dynamic system.

[*] Prof. Ben Schweizer, in a lecture about PDEs at TU Dortmund.

Trajectory The vector of state variables

$$\boldsymbol{x} : [0, T] \to \mathbb{R}^N \tag{1.4}$$

can be interpreted as a curve through the **state space** $\mathbb{R}^N$. A curve through state space will be called a **trajectory** of the system. If we denote $\mathscr{X}_i$ the space of all possible time courses of state variable x_i, the compound space

$$\mathscr{X} := \mathscr{X}_1 \oplus \ldots \oplus \mathscr{X}_N \tag{1.5}$$

reflects all possible curves through state space, i.e., all possible trajectories. Here, we follow the convention from [26] and write $\mathscr{X}$ as a direct sum ($\oplus$) as this will be consistent with the idea of *invariable sparsity*, proposed at a later point, where $\mathscr{X}_i$ can be an arbitrary Banach space. For the scope of this thesis, the formulation as a Cartesian product ($\times$) is equivalent and more prominent in the literature, though.

Vector Field Which trajectory $\boldsymbol{x} \in \mathscr{X}$ is realised by the system is not ambiguous but governed by its *laws of interaction*. As before, these laws can principally be formulated in a variety of ways. In practice, however, the common form is that of ordinary differential equations

$$\dot{x}_i(t) = f_i(\boldsymbol{x}(t), t) \tag{1.6}$$

where $f_i : \mathbb{R}^N \times [0, T] \to \mathbb{R}$ is a Lipschitz function. In the engineering literature, see for instance [27], predominantly in optimal control problems, the system is steered into a desired state by applying external inputs $\boldsymbol{u} = (u_1, \ldots, u_M)^T$ and the differential equation takes the form

$$\dot{x}_i(t) = f_i(\boldsymbol{x}(t), \boldsymbol{u}(t)) \,. \tag{1.7}$$

As soon as one chooses the input $\boldsymbol{u}$ these two formulations are equivalent. A system without explicit time dependency is called **autonomous**. According to (1.1) the variable t itself can be considered the state variable of a dynamic system, hence it turns out that any system can be made autonomous. We will henceforth suppress explicit time dependencies as well as *known* external inputs from our notation. The vector valued function

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{pmatrix} f_1(\boldsymbol{x}) \\ \vdots \\ f_N(\boldsymbol{x}) \end{pmatrix} \tag{1.8}$$

is the **vector field** of the system.

Flow Among all possible trajectories $x \in \mathscr{X}$ of a system, only those can be realised, which are in accordance with the initial value

$$\boldsymbol{x}(0) = \boldsymbol{x}_0 \tag{1.9}$$

and with the governing laws of interaction

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t)) \,. \tag{1.10}$$

The classical *Picard-Lindelöf theorem*, see [28, 29] for the original works and well known in the literature, ensures the existence and uniqueness of such a trajectory. The trajectory which is realised by the system will be called the **flow** $\varphi_t(\boldsymbol{x}_0)$ of the system.

Observables When it comes to practical problems, usually it is not possible to measure each state variable $x_1,\dots,x_N$ individually. Instead, one has only access to a smaller number of **observables** $y_1,\dots,y_P$, which carry some information about the state of the system, as the state is mapped to the observables via a measurement function

$$\boldsymbol{c} : \mathbb{R}^N \to \mathbb{R}^P . \tag{1.11}$$

The measurement function can also be interpreted as a map

$$\boldsymbol{c} : \mathscr{X} \to \mathscr{Y} \tag{1.12}$$

where

$$\mathscr{Y} = \mathscr{Y}_1 \oplus \ldots \oplus \mathscr{Y}_P \tag{1.13}$$

represents all possible trajectories of the observables. We call $\mathscr{Y}$ the **output space** and $\boldsymbol{y}(t) = \boldsymbol{c}(\varphi_t(\boldsymbol{x}_0))$ the **output** of the system.

1.2 Quantifying Model Errors

Having discussed the basic notions of a dynamic system, we are ready to define the first mathematical model for dynamic systems.

Definition 1 *A system of the form*

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{f}(\boldsymbol{x}(t)) \\ \boldsymbol{x}(0) &= \boldsymbol{x}_0 \\ \boldsymbol{y}(t) &= \boldsymbol{c}(\boldsymbol{x}(t)) \end{aligned} \tag{1.14}$$

is called an ***ordinary state space system****.*

Nominal Model and Identification of Model Errors When one builds a mathematical model for a dynamic process of interest, one usually comes to the form of an ordinary state space system. From the introductory discussion about model errors, we expect this model to be erroneous hence it is not an adequate description of reality. But it usually represents the best model we have. In [21] the term **nominal model** was introduced to express exactly this situation of a model that proved useful with some limitations but is not yet an accurate description of the true dynamics of a system.

The state variables $x_1,\dots,x_N$ of a system are usually not accessible, but the experimentally accessible quantities are given by the observables $y_1,\dots,y_P$. It becomes clear that any detection or reconstruction of model errors must solely rely on the observables of the system and that it must be seen in relation to the nominal model.

1.2.1 Exogenous Model Errors

Exogenous model errors are understood as migration processes, external perturbations etc. that have their origin beyond the boundaries of the considered system. See also figure 1.2 for a schematic illustration. Subfigure (a) shows a forest with a multitude of interacting entities $\boldsymbol{z} = (z_1, \ldots)^T$, trees, mammals, insects, etc. which would be necessary to accurately describe this complex system. Since already a single tree is a sophisticated dynamic system itself we understand the forest as a *dynamic network* in the sense of [30], i.e., a network of dynamic systems which stay in contact but also show some intrinsic *node dynamics*. When we define the nominal system, e.g., the vicinity of a single tree, we draw a boundary around the tree to separate it from the **exo-system**, the parts which we are not interested in at the moment. However, this boundary is purely imaginary and not a physical boundary, i.e., does not prevent the plants and animals to interact across this border. Subfigure (b) shows a view on the tree; the node dynamics might now be modelled in more detail. The state variables $\boldsymbol{x} = (x_1, x_2, x_3)^T$ apparently constitute a standalone system. By construction we know that they stay in some complex contact with the exo-system. Let us see how exogenous model errors can be incorporated into an ordinary state space system.

Make the assumption that we have a complete and accurate description of a complex system, take again a forest for instance. Say, we are actually interested in a smaller subsystem, such as the vicinity of a single tree. The N state variables $\boldsymbol{z} = (z_1, \ldots, z_N)^T$ of the complex system behave according to

$$\dot{\boldsymbol{z}}(t) = \boldsymbol{g}(\boldsymbol{z}(t)) \tag{1.15}$$

where $\boldsymbol{g}$ is the vector field of the complex model. Without loss of generality, the subsystem of interest (the vicinity of the tree) corresponds only to the first n state variables $z_1, \ldots, z_n$. One can introduce $x_i := z_i$ for $i = 1, \ldots, n$ as the state variables of interest. The differential equation of x_i can be written as

$$\dot{x}_i(t) = g_i(x_1, \ldots, x_n, z_{n+1}, \ldots, z_N)\,. \tag{1.16}$$

If g_i turns out to be independent of $z_{n+1}, \ldots, z_N$, then we can simply interpret

$$f_i(\boldsymbol{x}) := g_i(\boldsymbol{x}, z_{n+1}, \ldots, z_N) \tag{1.17}$$

as one component of the vector field of the subsystem. If g_i does depend on the states $z_{n+1}, \ldots, z_N$ we can still formally extract the variables of the subsystem

$$g_i(x_1, \ldots, x_n, z_{n+1}, \ldots, z_N) = f_i(\boldsymbol{x}) + \left(g_i(\boldsymbol{x}, z_{n+1}, \ldots, z_N) - f_i(\boldsymbol{x})\right)\,. \tag{1.18}$$

Here, f_i is principally an arbitrary function.

That the vector field $\boldsymbol{f}$ of the subsystem is theoretically an arbitrary function becomes plausible as soon as one becomes aware of the fact that the subsystem is an artificial construction. The arbitrariness of the vector field reflects the arbitrariness in choosing the (mathematical) boundaries of the subsystem. Despite this caprice it seems intuitive that choosing $\boldsymbol{f}$ *close to* $\boldsymbol{g}$ is advantageous for the numerical estimation, as will be explained in the later chapters.

The correct behaviour of $\boldsymbol{x}$ is obtained by the differential equation

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t)) + (P\boldsymbol{g}(\boldsymbol{z}(t)) - \boldsymbol{f}(\boldsymbol{x})(t)) \tag{1.19}$$

where P is understood as the projection onto the first n components. Now let η and φ be the flows of the accurate model (1.15) and of the nominal model

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t)), \tag{1.20}$$

respectively. If the nominal model would already produce the correct behaviour we would find

$$P\eta_t(\boldsymbol{z}_0) = \varphi_t(P\boldsymbol{z}_0). \tag{1.21}$$

But if the nominal model suffers from model errors, the quantity

$$\boldsymbol{w}(t) := P\boldsymbol{g}\left(\eta_t(\boldsymbol{z}_0)\right) - \boldsymbol{f}\left(P\eta_t(\boldsymbol{z}_0)\right) \tag{1.22}$$

describes the discrepancy between the vector fields of the accurate and the nominal model along the true flow $\eta_t(\boldsymbol{z}_0)$ for each point in time. Thus, installing the additional input $\boldsymbol{w}$ we obtain an augmented system with differential equation

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t)) + \boldsymbol{w}(t). \tag{1.23}$$

Let the flow of this augmented system be denoted as $\tilde{\varphi}$. This time we find

$$\tilde{\varphi}_t(P\boldsymbol{z}_0) = P\eta_t(\boldsymbol{z}_0). \tag{1.24}$$

The augmented system produces exactly the projection of the accurate complex system onto the desired subsystem.

1.2.2 Endogenous Model Errors

Endogenous model errors are understood as errors emerging from a poor model, e.g., due to inaccuracy of model parameters or simply due to limited knowledge about the fundamental processes. Consider again figure 1.2. The tree, represented in (a) by z_1, has an intrinsic node dynamics and splits into (x_1, x_2, x_3) in the more detailed view (b), representing the treetop, trunk, and roots. When we formulate a dynamic system with these three variables, we will encounter endogenous model errors: In reality, each of the nodes will not be characterised by a real number, but again show some complex dynamics that cannot be grasp by the three-dimensional nominal system. Or, even if we knew all state variables that are sufficient to describe the real system accurately, a precise description of all interactions is unrealistic to achieve. This limited knowledge about the complex relations within the boundaries of the system lead to endogenous model errors.

Assume that

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{g}(\boldsymbol{x}(t)) \tag{1.25}$$

is a completely correct description of a system with N state variables and flow $\eta_t(\boldsymbol{x}_0)$ and let

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t)) \tag{1.26}$$

be the nominal model. The model error can be written as

$$\boldsymbol{w}(t) := \boldsymbol{g}\left(\eta_t(\boldsymbol{x}_0)\right) - \boldsymbol{f}\left(\eta_t(\boldsymbol{x}_0)\right) \tag{1.27}$$

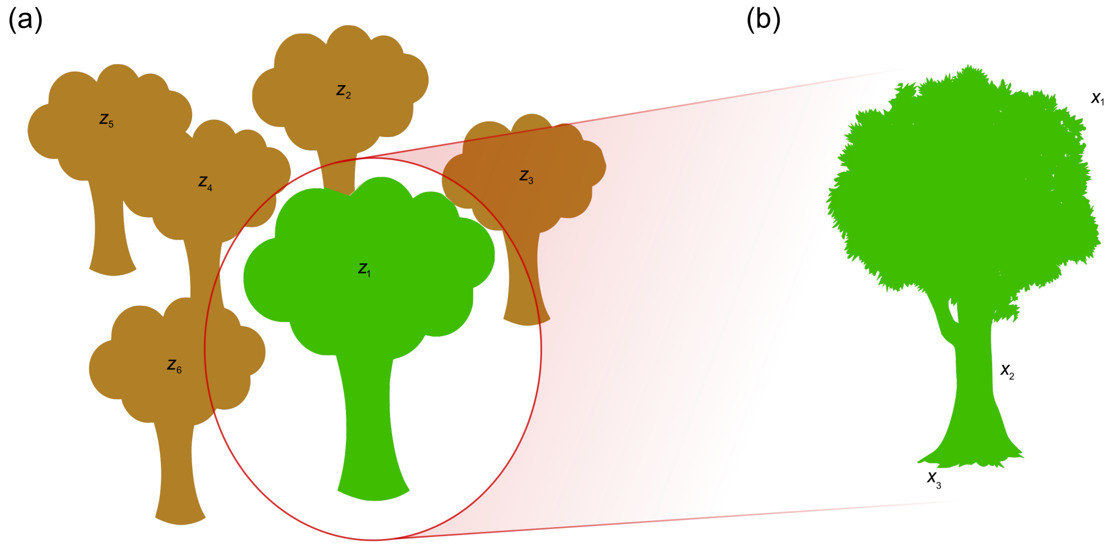

Figure 1.2: A vicinity of a tree becomes a nominal system.
Public domain (🅭∅) found at [31] were used. (a) A schematic illustration of the multiple state variables z_i of the complex system. We are interested in the vicinity of a single forest, illustratively described by the variables z_1, z_2, z_3. The boundary of the subsystem are drawn basing on our intuitive understanding, however, without mathematical reason. Furthermore, the boundary is an imaginary line but not a physical barrier. Hence, the interactions with the other state variables are not affected. (b) In the nominal model, the tree appears to be a standalone system.

such that we can again define an augmented model

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{w}(t)\,. \tag{1.28}$$

The flow of this augmented model again matches the correct flow $\eta_t(\boldsymbol{x}_0)$ exactly.

1.2.3 Endogenous or Exogenous?

In the discussion above it has been shown that endogenous and exogenous model errors both lead to an augmentation of the nominal model. This clearly implies the advantageous fact that both kinds of errors can be treated with the same methodology so that we can merely speak of structural model errors, disregarding their origin. However, it comes to the cost that we cannot distinguish endogenous from exogenous errors directly. In [21, 22] the analysis of correlations between the estimated model error and the state variables for the biochemical JAK-STAT system lead to the strong indication that the observed model error emerges from an unknown feedback loop in the system. Correlations, or anticipating the next chapter, differential-algebraic dependencies between the model error and the state variables may indicate its endogenous nature. For a large system and complex dependencies such an analysis might become practically infeasible, though. Also an exogenous model error can be assumed to be produced by some dynamic exo-system, as there is hardly any function, more precisely only the hyper-transcendental or differential-algebraically transcendental functions, which does not behave like any dynamic system, see for instance [32] for an early treatment. To finally understand the model error we would have to model the exo-system itself. Approaches to find the governing equations for observed state variables [33] could be applied to the numerical estimates for an exogenous model error.

The knowledge whether a model error is endogenous or exogenous is clearly informative for the design of new or more precise experiments. As well is a functional form for the so far only quantitatively estimated model error $\boldsymbol{w}$ beneficial in order to formulate an improved, predictive model. This thesis focuses on the reconstruction of model errors in the sense of finding an estimate for the unknown input function $\boldsymbol{w}$. The aforementioned analyses utilise methods which base on a trustworthy estimate for the time course of the errors and may therefore be subsequent goals which lie beyond the scope of this thesis.

1.3 Reconstruction of Model Errors

The discussion of endogenous and exogenous model errors above has shown that both can be remedied by augmentation of the nominal model with an additional **unknown input** $\boldsymbol{w} : \mathscr{T} \to \mathbb{R}^N$. Approaches for the same purpose have been presented in [34]. The augmentation of an ordinary system makes it loose its quality to be ordinary.

Definition 2 *A system of the form*

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{f}(\boldsymbol{x}(t)) + \boldsymbol{u}(t) \\ \boldsymbol{x}(t) &= \boldsymbol{x}_0 \\ \boldsymbol{y}(t) &= \boldsymbol{c}(\boldsymbol{x}(t)) \end{aligned} \tag{1.29}$$

is called a ***state space input-output system****.*

Originating in the theory of control systems in engineering and cybernetics, see the standard textbooks [27, 35] for a good overview over multi-input control, systems of the form above have been object of research for several decades.

Inputs The nomenclature of the unknown input that compensates model errors as $\boldsymbol{w}$ is chosen for historical reasons [21]. In the broader setting of input-output systems, we stick to the common terminology that $\boldsymbol{u}$ represents an input of the system. In analogy to the trajectory and output space of a dynamic system, we introduce the **input space**

$$\mathscr{U} = \mathscr{U}_1 \oplus \ldots \oplus \mathscr{U}_N \tag{1.30}$$

where $\mathscr{U}_i$ represents all allowed input functions which possibly augment the i^{th} component of the differential equation.

Input-Output Map An input-output system can be interpreted as map

$$\Phi : \mathscr{U} \to \mathscr{Y} \,. \tag{1.31}$$

If $\varphi_t^{\boldsymbol{u}}(\boldsymbol{x}_0)$ represents the flow of the system under input $\boldsymbol{u}$, the input-output map can be written as

$$\Phi(\boldsymbol{u})(t) = \boldsymbol{c}(\varphi_t^{\boldsymbol{u}}(\boldsymbol{x}_0)) \,. \tag{1.32}$$

1.3.1 The Model Error Reconstruction Problem

Utilising the interpretation of structural model errors in terms of dynamic input-output systems, we have everything ready to formulate the key problem of this thesis.

Model Error Reconstruction Problem (MERP) Let $\Phi : \mathcal{U} \to \mathcal{Y}$ be the input-output map of a state space input-output system and $\boldsymbol{y}^{\mathrm{data}} \in \mathcal{Y}$ be given data. Solve

$$\Phi(\boldsymbol{u}) = \boldsymbol{y}^{\mathrm{data}} \tag{1.33}$$

for $\boldsymbol{u}$.

If we find an input $\boldsymbol{w} \in \mathcal{U}$ that solves this problem, it can be seen as a numerical estimate for the model error of the nominal system.

In the last decades, the aim to solve what we call the Model Error Reconstruction Problem has been treated independently from different perspectives: The contribution to this subject from the engineering society is unmeasurable and only the publications with greater impact to the present research can be mentioned. Focusing on the geometrical and algebraic properties [36, 37, 38, 39, 40, 41, 42, 30, 43, 44, 45, 46, 47, 48, 49] present important developments of the field. Textbooks [39, 50] give an overview with the focus on fault detection for engineering. The frontier of generic and structural analysis has been pushed by [51, 52, 53, 54]. The application to biological systems has been discussed for instance in [55, 12, 10, 56, 57, 58].

1.3.2 Significance of SEEDS

In the light of the aforementioned, broad literature which deals with the MERP in diverse ways, the question arises in how much the present thesis contributes to the picture. To pick up the earlier formulated aims of this thesis, the *conditions for invertibility*, the algebraic approach introduced by Fliess [38] and the necessary structural condition by Wey [51] are refurbished and featured in a consistent and complete way. Furthermore, the lacking *proof for sufficiency* of the structural condition is provided.

As a new contribution, the Model Error Reconstruction Problem is treated with the methodology of network science, yielding insights into the *network principles* of invertibility. From the results of the network analysis, a practical sensor node placement algorithm is deduced.

A novel perspective is introduced by the finding that input-output systems are naturally equipped with an intrinsic *gammoid structure*. This new found structure paves the way for the *sparse sensing* of model errors.

The concept of *error localisability* is introduced as a relaxation of the error reconstruction. It is closely related to a measure of coherence between the model errors and between the sensors with which we observe the system.

Finally, a dynamic optimal control procedure which is inspired by classical optimal control problems is proven to be capable of producing model error estimates as a dynamic variation of *compressed sensing*.

1.3.3 Example

For the numerical computations and simulations we used `python` with the popular `numpy` and `scipy` packages. To see an example for the MERP consider figure 1.3. The equations of this toy model are inspired by cell biological pathways but simplified and designed to make the results clearer. The initial state is set as

$$\boldsymbol{x}_0 = 0\,. \tag{1.34}$$

One can read the equations of the nominal model directly from the graph in subfigure (a), since every arrow represents a linear interaction with strength $a = 0.75$ and every state variable has a self-loop of strength $b = -0.8$. E.g., the right hand side for x_5 is given by

$$f_5^{\text{nominal}}(\boldsymbol{x}) = a x_1 + a x_7 - b x_5\,. \tag{1.35}$$

The three measurable quantities $\boldsymbol{y} = (y_1, y_2, y_3)$ are the state variables x_2, x_4, and x_9. The initial value is $\boldsymbol{x}_0 = 0$.

Nominal Flow The nominal model reflects our a priori knowledge of the system and does not include the inputs $\boldsymbol{w} = (w_1, w_2, w_3)$. The flow of the nominal system can easily be deduced as

$$\varphi_t(0) = 0 \quad \forall t \in [0, T]\,. \tag{1.36}$$

So the nominal model rests in a fixed point and the outputs are constant zero

$$y_i(t) = 0 \quad \forall t \in [0, T]\,. \tag{1.37}$$

True Flow The *true* system is usually unknown to the scientist, otherwise the MERP would be pointless. To generate pseudo-experimental data, however, we have to define a ground truth. The true system differs from our nominal guess only in the right hand side for x_8, that is,

$$f_8^{\text{true}}(\boldsymbol{x}) = a x_5 + a x_7 + a x_{11} - b x_8\,. \tag{1.38}$$

Due to this, the flow of the true system differs from the constant zero nominal expectations. The data points in subfigure (b) shows output $\boldsymbol{y}^{\text{data}}$ of the true system *observed* at 10 discrete time points. We added Gaussian noise with a relative standard deviation of 5% of the measured value.

Flow Correction One will realise that the data $\boldsymbol{y}^{\text{data}}$ contradict the nominal model, as the measurements do not yield a constant zero output. We augment the system in order to fit the measurements. Let us, for simplicity, assume that we have three potential candidates for the model error, as indicated by the wiggly arrows in subfigure (a). In the augmented model, the right hand side for x_5 reads

$$f_5^{\text{augmented}}(\boldsymbol{x}, \boldsymbol{w}) = a x_1 + a x_7 - b x_5 + w_2\,. \tag{1.39}$$

Subfigure (b) shows the outputs of the augmented model for two choices of inputs. The two choices of inputs are shown in subfigures (c) and (d). The dotted lines (with inputs (c)) and the solid lines (with inputs (d)) show exactly the same results and match the data points perfectly. Thus, we have found two solutions of the MERP.

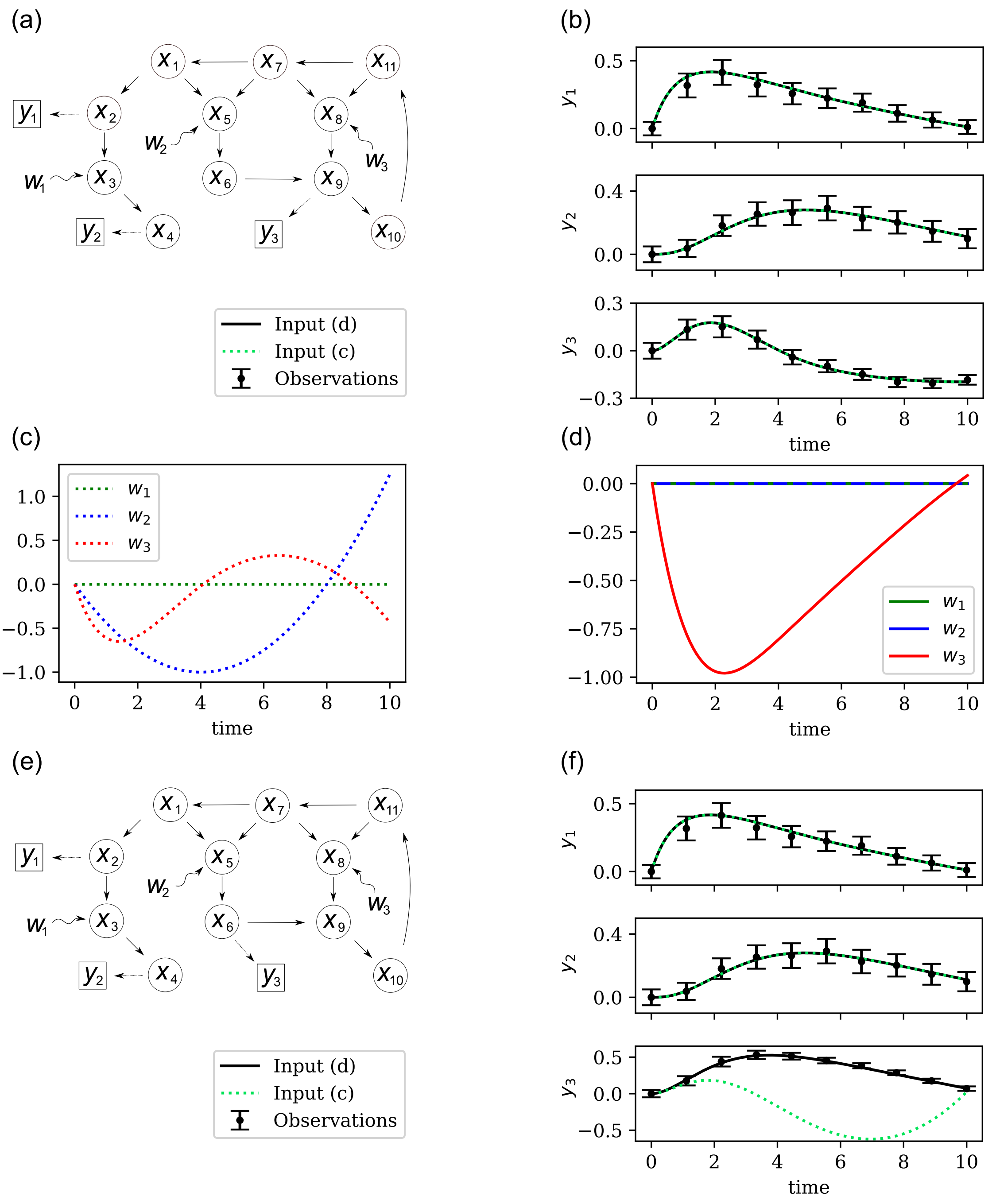

Figure 1.3: An exemplary model error reconstruction. (a) The considered system. Each node x_i represents one state variable, each arrow represents a linear interaction with coefficient one. The wiggly arrows indicate the additive inputs w_i, the square nodes represents the measured outputs y_i. The initial value is set to zero, $\boldsymbol{x}(0) = 0$. (b) The pseudo-experimental data of the system (a) and the corrected model with inputs from (c, dotted lines) and (d, solid lines). (c) One choice of input functions applied as w_1, w_2, w_3 to the systems (a) and (e). The corresponding trajectory of the system is shown as dotted lines in (b) and (f). (d) Another choice of input functions applied to the systems (a) and (e). The corresponding trajectory of the system is shown as solid lines in (b) and (f). (e) A variation of the system (a). It differs in the output y_3. (f) The pseudo-experimental data of the system (f) and the corrected model with inputs from (c, dotted lines) and (d, solid lines).

Replaced Outputs Now consider the model from subfigure (e). This model is identical to the model discussed above, with the only difference that we now measure the state variables x_2, x_4 and x_6. Again, the nominal model has constant zero outputs. Subfigure (f) shows the pseudo-experimental data points. The dashed and solid lines again show the outputs of the augmented model with the inputs from (c) and (d). One can see that the dashed and solid lines do not coincide this time. Output y_3 with inputs from (d) matches the data points exactly. With inputs (c), however, the output does not match the data points at all. Indeed, the input set from (d) is the only choice that is able to produce the desired output.

Discussion of the Error Estimates Let henceforth an asterisk, represent the correct description of the model error,

$$\Phi(\boldsymbol{w}^*) = \boldsymbol{y}^{\text{data}} . \tag{1.40}$$

A hat, for instance $\hat{\boldsymbol{w}}$, shall stand for our estimate of the model error. In the first model that was discussed above, there are two choices of inputs, that produce exactly the same output

$$\Phi(\boldsymbol{w}_{(c)}) = \Phi(\boldsymbol{w}_{(d)}) = \boldsymbol{y}^{\text{data}} . \tag{1.41}$$

Indeed, the theoretical results presented in a later chapter show, that there is an infinite dimensional function space of solutions. It is therefore highly questionable, whether $\boldsymbol{w}_{(c)}$ or $\boldsymbol{w}_{(d)}$ can serve as an estimate for the true model error $\boldsymbol{w}^*$ at all. In contrast to that, in the second model, subfigure (d) is the only input estimate that can correct the model flow

$$\Phi(\boldsymbol{w}_{(d)}) = \boldsymbol{y}^{\text{data}} \tag{1.42}$$

such that the outputs match the data points. Therefore, this estimate $\hat{\boldsymbol{w}} := \boldsymbol{w}_{(d)}$ must match the true model error $\boldsymbol{w}^*$, up to statistical uncertainties.

Whenever we know, that the estimate $\hat{\boldsymbol{w}}$ for the model error is unique, then it necessarily must match the true model error $\boldsymbol{w}^*$. And only the true model error leads to the correct estimate of the system state. Therefore we can formulate

$$\Phi(\hat{\boldsymbol{w}}) = \Phi(\boldsymbol{w}^*) \quad \Rightarrow \quad \hat{\boldsymbol{w}} = \boldsymbol{w}^* . \tag{1.43}$$

The uniqueness of solution for the MERP is therefore a crucial necessity for the credibility of the error estimate.

1.3.4 Invertibility of a Dynamic System

As soon as one has found an algorithm to solve the MERP, it is possible to compute an estimate $\hat{\boldsymbol{w}}$ for the model errors of the nominal system. One can then use this estimate to compute for example the correct state of the system at any time t via

$$\boldsymbol{x}(t) = \varphi_t^{\hat{\boldsymbol{w}}}(\boldsymbol{x}_0) . \tag{1.44}$$

But how trustworthy is this estimate? Say, there are two possible inputs $\boldsymbol{u}, \boldsymbol{v} \in \mathscr{U}$ that both solve

$$\Phi(\boldsymbol{u}) = \Phi(\boldsymbol{v}) = \boldsymbol{y}^{\text{data}} . \tag{1.45}$$

Which input should be considered the better estimate for the model error?

The first approach to answer this question is the rigorous claim for *uniqueness* of solution. Necessarily, this unique solution must correspond to the model error. The treatment introduced by Fliess [38] and Wey [51] yielded conditions which will be in the focus of the following two chapters. Therein as well as in the aforementioned literature, the term *invertibility of dynamic systems* has been established to express the uniqueness of solution.

The second approach is to introduce a *regularisation* strategy to obtain an ordering among all solutions such that we can choose the most preferable one. Engelhardt et. al [21, 22] have presented the *Dynamic Elastic Net* and the *Bayesian Dynamic Elastic Net* which both consider a dynamic version of the *Dynamic Elastic Net* which was originally presented to compute sparse and smooth estimates for static problems [59]. With the focus on *signal reconstruction* [60, 61] and *signal recovery* [62] regularisation strategies to get an *optimal representation* have been discussed.

Invertibility

Definition 3 *An input-output system is called* ***invertible****, if any solution of the Model Error Reconstruction Problem is unique.*

The definition above focuses in the intuitive understanding of invertibility as the counterpart of the problem of degenerated solutions of the MERP. It, however, deserves substantiation in form of testable conditions in order to decide whether a concrete system is invertible or not. So far, dynamic systems are formulated as *state space* models which can be interpreted as an input-output map $\Phi : \mathscr{U} \to \mathscr{Y}$. This brings us the first implementation of invertibility.

Definition 4 *Let Φ be the input-output map of a state space input-output system. The system is called* ***analytically invertible*** *if Φ is one-to-one.*

In the following chapter, two other representations of dynamic systems with their own varieties of invertibility, namely *differential-algebraic* and *structural invertibility*, will be discussed.

Robustness of the System Inversion In advance to a closer investigation of dynamic systems, a fundamental mathematical theorem unveils an issue intrinsic to the MERP. Hadamard [63] defined a *well-posed problem* by three conditions:

1. There is at least one solution.

2. There is at most one solution.

3. The solution depends continuously on the parameters.

Point one is silently assumed in this thesis. And this assumption is not an issue, since as soon as we see a deviation of the experimental data from the theoretical predictions, we deduce the existence of a model error. And as we have already seen that any model error can be understood as an additional input $\boldsymbol{w}$ to the system, we know that there exists at least one solution for the MERP for the given data. The second point is clearly equivalent to invertibility and hence in the focus of the following chapters. Point three has not been discussed, yet.

The input reconstruction problem belongs to the class of *inverse problems*. Let us assume, the operator Φ is linear. Given that the Hadamard conditions one and two are fulfilled, there exists an inverse map Φ^{-1} and the unknown input $\boldsymbol{w}$ can be inferred via

$$\Phi^{-1}(\boldsymbol{y}^{\mathrm{data}}) = \boldsymbol{w}\,. \tag{1.46}$$

A problem arises from the fact that $\Phi : \mathscr{U} \to \mathscr{Y}$ is a Hilbert-Schmidt operator [5], see also [64] for a comprehensive textbook, given that the vector field is sufficiently smooth. Therefore, $\Phi : \mathscr{U} \to \mathscr{Y}$ is a compact operator between infinite-dimensional spaces. It is a generally valid fact for such operators [64] that the inverse $\Phi^{-1} : \mathscr{Y} \to \mathscr{U}$ is not continuous. Hence, even if the operator Φ is linear and (left and right) invertible, we still have to face an ill-conditioned problem.

The issue of an ill-conditioned inverse problem is intrinsic to the model error estimation, as the input space and output space are both infinite dimensional. In praxis, the presence of measurement noise as well as numerical stiffness of the system make it hard to solve the inverse problem. Though a closer look on these issues is highly relevant before we can investigate the MERP in practical situations, there are still theoretical advances to make, thus let us for the moment assume the ideal, noiseless case.

CHAPTER 2

Differential-Algebraic Invertibility

Though many stones dœ bear greate price,
The whetstone is for exersice
As neadefull, and in woorke as straunge:
Dulle thinges and harde it will so chaunge,
And make them sharpe, to right good use:
All artesmen knowe, thei can not chuse,
But use his helpe: yet as men see,
No sharpenesse semeth in it to bee.

The grounde of artes did brede this stone
His use is greate, and moare then one.
Here if you lift your wittes to whette,
Moche sharpenesse therby shall you gette.
Dulle wittes hereby dœ greately mende,
Sharpe wittes are fined to their fulle ende.
Now proue, and praise, as you dœ finde,
And to your self be not unkinde.

Robert Recorde, The Whetstone of Witte [65]

Going back to the very outset of dynamic systems as *entities* and *interactions*, one will agree that the mathematical representations of dynamic systems can be divers. The *state space* form is without doubt the most prominent representation, for it works with the state variables x_i as functions of time and with the vivid geometrical understanding of trajectories through space. This chapter will employ the theory of *differential-fields* and formulate a dynamic system as a differential-algebraic problem. Differential-algebra was invented by Ritt [66] and mainly investigated by Ritt, Kolchin [67] and Kaplansky [68], who already made the statement [68]:

> Differential algebra is easily described: it is (99 per cent or more) the work of Ritt and Kolchin.

The differential-algebraic treatment of dynamic systems was established by Fliess [38], see also [69] for a later, comprehensive review by him. This chapter shall therefore be understood as an essay on a concise demonstration of a differential-algebraic toolbox adjusted to handle dynamic systems rather than a contribution to differential-field theory; especially for non-linear systems, when it is not possible to write down the input-output map Φ, or if parameters of the model are not known precisely, the differential-algebraic treatment offers powerful theorems and insights into dynamic systems. The novel contribution that utilises this toolbox will then be the result of chapter 3.

The Exponential As an illustrative example, let us briefly motivate the step from state space to differential-algebra. In state space, the laws that govern the system's behaviour are formulated as differential equations. For instance, *exponential growth* is described by

$$\dot{x}(t) = \alpha x(t)\,. \tag{2.1}$$

It is well known from the earlier mentioned Picard-Lindelöf theorem that there is one and only one solution for each initial value $x_0 \in \mathbb{R}$, here

$$x(t) = x_0 \mathrm{e}^{\alpha t} = x_0 \sum_{k=0}^{\infty} \frac{(\alpha t)^k}{k!}\,. \tag{2.2}$$

We rewrite equation (2.1) to achieve a more algebraic style by replacing the *function* x by an *indeterminate* X, so that the equation becomes

$$\frac{\mathrm{d}}{\mathrm{d}t} X = \alpha X\,. \tag{2.3}$$

The latter equation does not appear to add much to the understanding of the exponential function. Actually, it leads to the question how we can apply a derivation-operator, which actually stems from Analysis, to an algebraic indeterminate. With differential-algebra, it is possible to bring this two worlds together.

More than that, whereas equation (2.1) is usually understood as an equation in $\mathbb{R}$, the differential-algebraic version (2.3) can be formulated over any *differential-field.* Differential-fields are, for instance, all rational functions of time, or the *Laurent-polynomials.* Also equations from physical field theory, for instance from thermodynamics, fluid dynamics or Maxwell's theory of electromagnetism, see [70, 71, 72] for textbooks, respectively, can be written as (partial-)differential-polynomials. The solution of equation (2.1) with α not being a real number but an element of a differential-field leads to a completely different solution from the Analysis point of view. However, the algebraic structure of this equation and hence of the solution remains the same.

2.1 D-Algebra for Dynamic Systems

Henceforth, the attribute *differential* will be abbreviated *d.* The attribute *classical* emphasises the non-differential nature of some algebraic object. In several publications, Fliess utilised d-algebra in his treatment of dynamic control systems, consider his review [69] and references therein. The control of dynamic systems means to steer the system into a desired state by applying additional input signals. This is the point of view for many engineering-motivated applications, but obviously it is mathematically equivalent to structural model errors which also act like additional inputs to the system. Thus, the *reverse engineering* problem, i.e., the inference of the control inputs from the system's trajectory, is equivalent to the MERP.

Before we can make use of the rich results from the d-algebraic treatment of input-output systems, it is necessary to discuss the basic terminology of d-algebra and to see how a state space system is translated into the d-algebra setting.

2.1.1 d-Rings and d-Fields and d-Ideals

d-Rings

Definition 5 *A* ***ring*** $\Re$ *is understood as a set with* ***addition*** *"+" and* ***multiplication*** *"·" and with neutral elements* 0 *and* 1*, respectively, i.e.,*

$$a+0=a \quad , \quad 1\cdot a=a, \tag{2.4}$$

and such that the properties

1. *For* $a\in\Re$ *there is an* $(-a)\in\Re$ *with* $a-a:=a+(-a)=0$
2. $a+b=b+a$ *and* $ab=ba$
3. $a(b+c)=ab+bc$

hold. If additionally there is a ***derivation***

$$\partial:\Re\to\Re \tag{2.5}$$

that is linear and fulfils the Leibniz-rule

$$\partial(ab)=\partial ab+a\partial b \tag{2.6}$$

then $\Re$ *is called a* ***d-ring****.*

Example (Constant Rings)

Any (classical) ring $\Re$ can be equipped with a derivation for which we define $\partial a=0$ for each element $a\in\Re$.

In a given d-ring $\Re$ one can *define* the constants as those elements a which fulfil $\partial a=0$. Consider

$$\partial(1\cdot a)=\partial(1)\cdot a+1\cdot\partial a \tag{2.7}$$

to see that the unit element $1\in\Re$ is necessarily a constant.

Thus, every classical ring is also a (constant) d-ring, and every d-ring has a constant hence classical sub-ring.

This first example has shown that classical rings are a special case of d-rings. Subsequently it is plausible that many properties and theorems from classical algebra can be extended or modified for d-algebra. The second example demonstrates how we can compute maxima of a Laurent-polynomial without computing derivatives, but just through d-algebraic operations.

Example (Laurent Polynomials)

Consider the ring of Laurent-polynomials

$$p(s)=a_1\frac{1}{s}+a_2\frac{1}{s^2}+\ldots+a_n\frac{1}{s^n} \tag{2.8}$$

with real or complex coefficients a_i. The derivation $\partial := {}^{\mathrm{d}}/_{\mathrm{d}s}$ seems natural. But the derivation operator can also be written directly as multiplication

$$\partial : \frac{1}{s^n} \mapsto \frac{-n}{s}\frac{1}{s^n}\,. \tag{2.9}$$

d-Fields

Definition 6 *If $\mathbb{K}$ is a ring that has also a multiplicative inverse $a^{-1} \in \mathbb{K}$*

$$a \cdot a^{-1} = 1 \tag{2.10}$$

*for each element $a \neq 0$, then $\mathbb{K}$ is called a **field**. If $\mathbb{K}$ is equipped with a derivation, it is called a **d-field**.*

Example (Meromorphic Functions)

Consider the set of meromorphic functions of x, i.e., functions that are holomorphic except for poles. Meromorphic functions together with the derivation $\partial := {}^{\mathrm{d}}/_{\mathrm{d}x}$ form a d-field.

d-Indeterminates Now let $\mathfrak{R}$ be a d-ring with derivation ∂. If we consider an indeterminate X, this indeterminate has to be consistent with the differential structure of $\mathfrak{R}$. In d-algebra, a **d-indeterminate** X represents a whole family $\{X^{(0)}, X^{(1)}, \ldots\}$ which respects the differential structure through

$$\partial^k X = X^{(k)}\,. \tag{2.11}$$

Example (Exponential Growth)

Let $\mathfrak{R}$ be a d-ring, $\alpha \in \mathfrak{R}$ and X a d-indeterminate. Intuitively we would call the solution of

$$\partial X - \alpha X = 0 \tag{2.12}$$

the exponential function $\exp(\alpha)$.

For the case $\mathfrak{R} = \mathbb{R}$ and $\partial = {}^{\mathrm{d}}/_{\mathrm{d}t}$, we find the well-known exponential function

$$X = \mathrm{e}^{\alpha t} = 1 + \alpha t + \alpha^2 \frac{t^2}{2} + \alpha^3 \frac{t^3}{6} + \ldots \tag{2.13}$$

is a solution.

For the case of a non-constant d-field $\mathfrak{R}$, e.g., again the Laurent-polynomials we can define the exponential of a non-constant $\alpha \in \mathfrak{R}$. For $\alpha = 1/s$ we find

$$\exp\left(\frac{1}{s}\right) = s \tag{2.14}$$

and for $\alpha = 1/s^2$

$$\exp\left(\frac{1}{s^2}\right) = \mathrm{e}^{-\frac{1}{s}}. \tag{2.15}$$

When we consider several d-indeterminates $X_1, \ldots, X_N$, we prefer the notation as vector

$$\boldsymbol{X} = (X_1, \ldots, X_N). \tag{2.16}$$

We do this for two for two reasons: First, for consistency with the state space notation where the state $\boldsymbol{x}$ is also vector valued. So a trajectory $\boldsymbol{x}$ is one possible value for the d-indeterminate $\boldsymbol{X}$. Second, though the sorting of the d-indeterminates is arbitrary, in a concrete case it is convenient to have a fixed ordering.

d-Polynomials

Definition 7 *Let $\mathbb{K}$ be a d-field and X a d-indeterminate. A sequence $\alpha = (\alpha_0, \alpha_1, \ldots)$ in $\mathbb{N}_0$ that converges to zero can be understood as multi-index for the d-indeterminate. We call*

$$X^\alpha := X^{\alpha_0} (\partial X)^{\alpha_1} \left(\partial^2 X\right)^{\alpha_2} \left(\partial^3 X\right)^{\alpha_3} \ldots \tag{2.17}$$

*a **d-term** and*

$$cX^\alpha \tag{2.18}$$

*with $c \in \mathbb{K}$ a **d-monomial**. The **degree** of a d-term is*

$$\deg X^\alpha = \sum_{k=0}^{\infty} \alpha_k \tag{2.19}$$

*and the **order***

$$\operatorname{ord} X^\alpha = \max_{\alpha_k \neq 0} k. \tag{2.20}$$

Let $\mathscr{A}$ be a finite set of multi-indices and $c_\alpha \in \mathbb{K}$, then

$$p(X) := \sum_{\alpha \in \mathscr{A}} c_\alpha X^\alpha \tag{2.21}$$

*is a **d-polynomial** in X over $\mathbb{K}$. For a d-ring $\mathfrak{R}$ the d-polynomials in X are $\mathfrak{R}\{X\}$. The polynomials, d-polynomials of order zero, are denoted $\mathfrak{R}[X]$.*

The quotient field of $\mathfrak{R}[X]$ is $\mathfrak{R}(X)$. The quotient field of $\mathfrak{R}\{X\}$ is $\mathfrak{R}\langle X \rangle$.

Consider a differential equation

$$\dot{x}(t) = a_0 + a_1 x(t) + a_2 x^2(t) + \ldots + a_n x^n(t) \tag{2.22}$$

where in many practical cases a_i are real coefficients. The differential equation would then be autonomous. In the d-algebraic framework, the coefficients can come from a d-ring $\mathfrak{R}$. For instance, coefficients from the d-ring $\mathfrak{R} = \mathbb{R}\{t\}$ with derivation $\partial = {}^{\mathrm{d}}/_{\mathrm{d}t}$ make the differential equation non-autonomous. We can formulate the d-polynomial $p(X) \in \mathfrak{R}\{X\}$

$$p(X) = \partial X - (a_0 + a_1 X + a_2 X^2 + \ldots + a_n X^n) \tag{2.23}$$

and find, that if a function x solves (2.22), then it is a root of the d-polynomial, i.e., $p(x) = 0$.

d-Ideal

Definition 8 *Let $\mathfrak{R}$ be a d-ring and $r_1, \dots, r_N \in \mathfrak{R}$. The **d-ideal** generated by $r_1, \dots, r_N$ is the set*

$$\mathfrak{I}(r_1, \dots, r_N) := \mathfrak{R}\{r_1, \dots, r_N\} \backslash \mathfrak{R}, \tag{2.24}$$

*that is, the d-polynomials in $r_1, \dots, r_N$ without constant term. The **quotient ring***

$$\mathfrak{R}_{\sim r_1, \dots, r_N} := \mathfrak{R} / \mathfrak{I}(r_1, \dots, r_N) \tag{2.25}$$

is understood as the elements from $\mathfrak{R}$ together with the identification

$$a \sim b \overset{Def}{\Leftrightarrow} q = p + \omega \tag{2.26}$$

with $\omega \in \mathfrak{I}(r_1, \dots, r_N)$.

For simplicity we will introduce the sign $\mathfrak{R}_\sim$ whenever there is no confusion about how the d-ideal is generated. If a d-field is the quotient field of a ring, for instance $\mathbb{K}\langle X \rangle$ is the quotient field of the d-ring $\mathbb{K}\{X\}$, then $\mathbb{K}\langle X \rangle_\sim$ denotes the quotient field of the quotient ring $\mathbb{K}\{X\}_\sim$.

In the treatment of dynamic systems, d-ideals can be utilised to ensure that the calculations respect the dynamics of the system. For instance, consider the d-polynomials over $(\mathbb{R}, \mathrm{d}/\mathrm{d}t)$,

$$\begin{aligned} p(X, Y) &= X^{(1)} - Y^{(0)} \\ q(X, Y) &= Y^{(1)} + X^{(0)} \end{aligned} \tag{2.27}$$

and the d-ideal $\mathfrak{I}(p) \subset \mathbb{R}\{X, Y\}$. Since $p \in \mathfrak{I}(p)$, also $\partial p \in \mathfrak{I}(p)$. Hence in the quotient ring $\mathbb{R}\{X, Y\}_{\sim p}$

$$q(X, Y) \sim q(X, Y) + \partial p(X, Y) = X^{(2)} + X^{(0)}. \tag{2.28}$$

We have reformulated the d-polynomial q with respect to the dynamics dictated by p and by this reduced the number of d-indeterminates and d-polynomials needed to describe the system.

Insensitivity to Initial Values Let us continue with the d-polynomial $q(X)$ from above to discuss one characteristic of the d-algebraic approach, which can be advantageous but also disadvantageous under certain circumstances. The flow of a system is a solution of the governing differential equations (in state space) or, equivalently, a root of the corresponding d-polynomials.

We find, that both, $\sin(t)$ as well as $\cos(t)$ are roots of the d-polynomial $q(X)$. This is plausible since both functions solve the same differential equation and merely differ in their initial value. Still, to formulate the general solution of the differential equation

$$\ddot{x}(t) + x(t) = 0 \tag{2.29}$$

both must be considered.

In the treatment of model errors, this insensitivity leads to the advantage that the results obtained by d-algebra are independent of the initial state of the system. Hence they have a higher generality compared to the analytic invertibility of the input-output map Φ since

$$\Phi(\boldsymbol{u}) = \boldsymbol{c}(\varphi_t^{\boldsymbol{u}}(\boldsymbol{x}_0)) \tag{2.30}$$

might be one-to-one for some initial values but for others it may not.

Example (Pathological Initial Value)

Consider the small state space system

$$\begin{aligned} \dot{x}(t) &= u(t) - x(t) \\ y(t) &= x^2(t) \end{aligned} \tag{2.31}$$

with some initial value x_0. The input-output map for this system is

$$\Phi(u) = \left(x_0 \mathrm{e}^{-t} + \int_0^t \mathrm{e}^{\tau - t} u(\tau) \,\mathrm{d}\tau \right)^2 . \tag{2.32}$$

One will realise that if $x_0 = 0$, then $\Phi(u) = \Phi(-u)$ and the system is therefore analytically not invertible. However, for non-zero initial values $x_0 \neq 0$, the map is one-to-one, hence in most cases the system is analytically invertible.

At a later point we will introduce the notion of *d-algebraic invertibility*. The d-algebraic treatment leads to a d-polynomial in U over $\mathbb{R}\langle Y \rangle$

$$\begin{aligned} p(U) =& (U - U^{(1)})Y^{(3)} + (U - 2U^{(1)} + U^{(2)})Y^{(2)} + (2U - 3U^{(1)} + U^{(2)})Y^{(1)} \\ &+ (3U(U^{(1)})^2 - U^2U^{(1)} - 2U^2U^{(2)}) \end{aligned} \tag{2.33}$$

for which, with respect to the d-ideal of the system,

$$p(U) \sim 0 . \tag{2.34}$$

One will see that the d-algebraic approach does not cover the case of a pathological initial value, but it makes a general statement about the algebraic dependencies within the system.

2.1.2 From State Space to d-Algebra

Utilising the language developed so far, we have everything ready to formulate a dynamic system in d-algebraic language. State space systems are usually formulated over the real numbers $\mathbb{R}$, possibly with explicit time-dependence. Hence, we consider the d-field $\mathbb{R}\langle t \rangle$ and derivation $\partial := {}^{\mathrm{d}}/_{\mathrm{d}t}$.

In the naive reformulation the state space equation

$$\dot{x}_i(t) = f_i(t, \boldsymbol{x}(t), \boldsymbol{u}(t)) \tag{2.35}$$

is translated into

$$q_i(\boldsymbol{X}, \boldsymbol{U}) := X_i^{(1)} - p_i(\boldsymbol{X}, \boldsymbol{U}) . \tag{2.36}$$

At this point we make the assumption, that f_i is a polynomial in the variables $\boldsymbol{x}$ and $\boldsymbol{u}$, e.g., the vector field component

$$f_i(x,t) = \mathrm{e}^{-t}x(t) \tag{2.37}$$

is translated into a polynomial $p(X) \in \mathbb{R}\langle t\rangle\{X\}$

$$p(X) = aX \tag{2.38}$$

with a coefficient $a \in \mathbb{R}\langle t\rangle$. The case of non-polynomial f_i will be discussed subsequently. The observables of the system can be characterised through the polynomials

$$h_j(Y_j,\boldsymbol{X}) = Y_j - c_j(\boldsymbol{X}) \tag{2.39}$$

where c_j is a component of the state space measurement function. It turns out convenient and without loss of generality to interpret

$$\boldsymbol{Y} \subseteq \boldsymbol{X}\,. \tag{2.40}$$

For the d-algebraic formulation it makes no difference whether the state space system is autonomous or not, it might even be formulated over any other d-field.

2.2 Ordinary d-Algebraic Systems

The *ordinary state space system* has been introduced in chapter one as the standard form for nominal models. Reformulated in d-algebra language, the state space equations always lead to a very specific form of d-polynomials,

$$q_i(\boldsymbol{X}) = X_i^{(1)} - p_i(\boldsymbol{X}) \tag{2.41}$$

where p_i is a classical polynomial.

One might come to the conclusion that d-algebraic systems can be much more complicated, for instance including higher order derivatives with a higher degree. Soon it will be shown that we find a standard form for *ordinary d-algebraic systems* which serve as nominal models. We discuss the connection between ordinary systems and the d-transcendental degree of d-algebraic d-field extensions which will turn out to play a central role in the d-algebraic invertibility theory.

2.2.1 Ordinary System

Definition 9 *Let $\mathbb{K}$ be a d-field and $\boldsymbol{X} = (X_1,\ldots,X_N)$* ***state d-indeterminates****. A system of the form*

$$q_i(X_i;\boldsymbol{X}) = X_i^{(1)} - p_i(\boldsymbol{X}) \tag{2.42}$$

with $p_i \in \mathbb{K}[\boldsymbol{X}]$ and $i = 1,\ldots,N$ together with a set of observables $\boldsymbol{Y} \subseteq \boldsymbol{X}$ is called an ***ordinary d-algebraic system****.*

In the term above we introduce the notation $q_i(X_i;\boldsymbol{X})$, meaning it contains $X_i^{(1)}$ as leading term plus some classical polynomial expression in $\boldsymbol{X}$. To see that ordinary d-algebraic systems are indeed a standard form, the following paragraphs show how non-ordinary systems can be brought into the desired form.

Observables Assume we have an ordinary system with state indeterminates $\boldsymbol{X}$ and d-polynomials $q_1, \dots, q_N$, but the observable Y is characterised by

$$h = Y - c(\boldsymbol{X}) \tag{2.43}$$

where $c \in \mathbb{K}\{\boldsymbol{X}\}$. We can introduce a new d-indeterminate $\tilde{Y}$ and set $\tilde{\boldsymbol{X}} = (X_1, \dots, X_N, Y)$. Via

$$\tilde{q}(Y; \tilde{\boldsymbol{X}}) := \partial h = Y^{(1)} - \partial c(\tilde{\boldsymbol{X}}) \tag{2.44}$$

we can interpret Y as a state indeterminate and characterise the new observable by

$$\tilde{h} = \tilde{Y} - Y\,. \tag{2.45}$$

In the d-ideal $\Im(\tilde{h})$ we identify

$$\tilde{Y} \sim Y \tag{2.46}$$

hence we can equivalently write $\tilde{Y} \subseteq \tilde{\boldsymbol{X}}$. One will realise that the system is not yet ordinary due to the term $\partial c(\tilde{\boldsymbol{X}})$, which is not necessarily a classical polynomial, as required for ordinary systems.

d-Rational q_i As a first case, consider q_i a d-rational in $\mathbb{K}\langle \boldsymbol{X} \rangle$. Since $\mathbb{K}\langle \boldsymbol{X} \rangle$ is the quotient field of $\mathbb{K}\{\boldsymbol{X}\}$, there are d-polynomials $\hat{q}_i$ and $\tilde{q}_i$ such that

$$q_i = \frac{\hat{q}_i}{\tilde{q}_i} \tag{2.47}$$

Due to the absorption property of the d-ideal $\Im(q_i)$

$$q_i \sim q_i \tilde{q}_i = \hat{q}_i\,, \tag{2.48}$$

hence $q_i \sim \hat{q}_i$. This shows that the d-ideals $\Im(q_i)$ and $\Im(\hat{q}_i)$ are equivalent. Hence it is sufficient to consider d-polynomials.

Higher Order d-Polynomial q_i Now let $q \in \mathbb{K}\langle \boldsymbol{X} \rangle$ be a higher order d-polynomial. For better readability a single indeterminate X is considered; the extension to the multivariate case is straightforward. Let m be the order of $q(X)$. Without loss of generality we can assume that $X^{(m)}$ appears linearly in q since if

$$q(X) = \ldots + p(X)(X^{(m)})^n\,, \tag{2.49}$$

where p is of order at most $m-1$, then

$$\partial q(X) = \ldots + \partial p(X)(X^{(m)})^n + np(X)(X^{(m)})^{n-1} X^{(m+1)}\,. \tag{2.50}$$

The latter expression has a leading term $X^{(m+1)}$ of degree one. As a d-ideal $\Im$ is closed in the sense

$$q \in \Im \Rightarrow \partial q \in \Im\,, \tag{2.51}$$

the shift from q to ∂q causes no issue. For $k = 1,\dots,m-1$ each derivation $\partial^k X$ is going to be replaced by a new d-indeterminate Z_k. We set $\boldsymbol{Z} := (X, Z_1, \dots, Z_{m-1})$ and

$$q_0(X;\boldsymbol{Z}) := X^{(1)} - Z_1 \tag{2.52}$$

and for $k = 1,\dots,m-1$

$$q_k(Z_k;\boldsymbol{Z}) := Z_{k-1}^{(1)} - Z_k\,. \tag{2.53}$$

In $q(X)$ we replace each X^k with Z_k for $k = 1,\dots,m-1$ to get

$$q(\boldsymbol{Z}) = f(\boldsymbol{Z}) + g(\boldsymbol{Z})Z_{m-1}^{(1)} \tag{2.54}$$

where f and g have order zero. From this we define

$$\tilde{q}(Z_{m-1};\boldsymbol{Z}) = Z_{m-1}^{(1)} - p(\boldsymbol{Z})\,, \tag{2.55}$$

with $p(\boldsymbol{Z}) = -f(\boldsymbol{Z})/g(\boldsymbol{Z})$.

Rational Vector Field Consider

$$q(X;X) = X^{(1)} - f(X) \tag{2.56}$$

where $f(X) \in \mathbb{K}(X)$. The extension to the multivariate case is again straightforward. Since f is a rational, we can write

$$f(X) = \frac{p(X)}{g(X)} \tag{2.57}$$

with two polynomials $p, g \in \mathbb{K}[X]$. We define

$$\begin{aligned} q_1(X;X,Z) &:= X^{(1)} - p(X)Z \\ q_2(Z;X,Z) &:= Z^{(1)} + Z^2\left(\frac{\partial g}{\partial X}p(X)Z + (\partial g)(X)\right). \end{aligned} \tag{2.58}$$

Above we applied the finding that for a polynomial

$$g(X) = a_0 + a_1X + \ldots + a_nX^n \tag{2.59}$$

the derivative

$$\partial(g(X)) = \partial a_0 + \partial a_1 X + a_1\partial X + \ldots + \partial a_n X^n + a_n n X^{n-1}\partial X \tag{2.60}$$

can be written as

$$\partial(g(X)) = \frac{\partial g}{\partial X}\partial X + \underbrace{(\partial a_0 + \partial a_1 X + \ldots + \partial a_n X^n)}_{=:(\partial g)(X)}\,. \tag{2.61}$$

One will see that with respect to the d-ideal $\Im(q_1, q_2) \subseteq \mathbb{K}\langle X, Z\rangle$

$$Z \sim \frac{1}{g(X)} \tag{2.62}$$

and thus it is equivalent to the original d-ideal $\Im(f) \subseteq \mathbb{K}\langle X\rangle$.

d-Algebraic Function Finally, let $f : X \mapsto f(X)$ be a function that fulfils any differential equation over $\mathbb{K}$ as a classical field. For simplicity only the univariate case is demonstrated. The differential equation that f fulfils can be understood as a d-polynomial h in the d-field $(\mathbb{K}, \partial_X)$

$$h(L) = \sum_{\alpha} c_\alpha L^{\alpha_0} (\partial_x L)^{\alpha_1} \dots . \tag{2.63}$$

Let m be the order of this d-polynomial. Now we replace the derivative $\partial_X^k L$ by a new d-indeterminate Z_k so that we obtain a polynomial $\tilde{h}(\boldsymbol{Z}) \in \mathbb{K}[\boldsymbol{Z}]$. We also introduce

$$q_k := Z_k^{(1)} - Z_{k+1} X^{(1)} \tag{2.64}$$

for $k \in \mathbb{N}_0$ up to a sufficiently large number, which represents the chain rule

$$\partial f(X) = \big(\partial_X f(X)\big)\, \partial X . \tag{2.65}$$

We find that

$$\mathbb{K}\langle X\rangle_{\sim f} = \mathbb{K}\langle X, \boldsymbol{Z}\rangle_{\sim \tilde{h}, q_0, q_1, \dots} . \tag{2.66}$$

We can now bring $\tilde{h}$ into standard form as shown above so that we achieve an ordinary system that is equivalent to the original one.

2.2.2 Generating Systems

An important role is played by what will be called a *generating system*. The concept of generating systems is similar to the idea of *polynomial generators* which was discussed in [73]. However, the treatment therein is limited to the field $\mathbb{R}$. In the following, a more general version for arbitrary d-fields is presented. With respect to that we need to introduce the idea of algebraic and transcendental elements of field extensions first.

Definition 10 *Let the field $\mathbb{K}$ be a proper subset of another field $\mathbb{L}$. The symbol $\mathbb{L}/\mathbb{K}$ is called a **field extension**. An element $l \in \mathbb{L}$ is called **algebraic** over $\mathbb{K}$ if there is a polynomial $p(X) \in \mathbb{K}[X]$ such that*

$$p(l) = 0 . \tag{2.67}$$

*If l is not algebraic, it is called **transcendental**. If each element of $\mathbb{L}$ is algebraic, $\mathbb{L}/\mathbb{K}$ is called an **algebraic field extension**.*

*Moreover, $\mathbb{L}/\mathbb{K}$ is a **d-field extension** if $\mathbb{L}$ and $\mathbb{K}$ are d-fields. An element $l \in \mathbb{L}$ is **d-algebraic** if there is a d-polynomial $p(X) \in \mathbb{K}\{X\}$ such that*

$$p(l) = 0 . \tag{2.68}$$

*Else it is called **d-transcendental** and if each element is d-algebraic, the d-field extension is called a **d-algebraic d-field extension**.*

In classical algebra, there is a connection between finite and algebraic field extensions. In d-algebra, a single d-indeterminate represents infinitely many classical indeterminates. Hence, the concept of finite fields is not applicable. The following idea of generating system however seems to provide an appropriate alternative. Generating systems will enable us to proof d-algebraic versions of theorems which are known in classical algebra and whose proof actually utilises the connection between finite and algebraic field extensions.

Definition 11 *Let $\mathbb{L}/\mathbb{K}$ be a d-field extension and $y \in \mathbb{L}$. A system*

$$q_i(X_i; \boldsymbol{X}) = X_i^{(1)} - p_i(\boldsymbol{X}) \tag{2.69}$$

*with $i = 1, \dots, N$ and $X_1 \sim y$ is called a **generating system** for y.*

Proposition 1 *Let $\mathbb{L}/\mathbb{K}$ be a d-field extension. If $y \in \mathbb{L}$ is generated by a system over $\mathbb{K}$, then it is d-algebraic over $\mathbb{K}$.*

Proof Consider the generating system

$$q_i(X_i; \boldsymbol{X}) = X_i^{(1)} - p_i(\boldsymbol{X}) \tag{2.70}$$

for $i = 1, \dots, N$ and

$$y \sim X_1 . \tag{2.71}$$

At this point it is more convenient to replace the latter by

$$h = y - X_1 \tag{2.72}$$

and consider the d-ideal $\Im(q_1, \dots, q_N, h)$, for short denoted $\Im$. We understand

$$y \sim X_1 \tag{2.73}$$

as a replacement rule for X_1. The derivative

$$\partial h = y^{(1)} - X_1^{(1)} \tag{2.74}$$

is again in $\Im$. With respect to the d-ideal we can now replace $X_1^{(1)}$ with $p_1(\boldsymbol{X})$ and therein each appearance of X_1^k with y^k. This yields a d-polynomial $\tilde{h} \sim \partial h$ in $\mathbb{K}\langle y\rangle[X_2, \dots, X_N]$.

We now show, how to eliminate X_2 from this field to get a d-polynomial in $\mathbb{K}\langle y\rangle[X_3, \dots, X_N]$. As an initial step set $h_{[0]} := \tilde{h}$. By construction it is clear that $h_{[0]} \in \Im$ and that it has some degree r with respect to X_2. We will now give an constructive algorithm that reduces the degree. Assume $\deg(h_{[m]}) = r \neq 0$ and $h_{[m]} \in \Im$, so there are coefficients $c_k \in \mathbb{K}\langle y\rangle(X_3, \dots, X_N)$ such that

$$h_{[m]} = \sum_{k=0}^{r} c_k X_2^k . \tag{2.75}$$

The latter can be written as

$$X_2^r = h_{[m]} - \sum_{k=0}^{\tilde{r}} \frac{c_k}{c_r} X_2^k \tag{2.76}$$

with $\tilde{r} < r$. Applying the derivation to $h_{[m]}$ yields

$$\partial h_{[m]} = \sum_{k=0}^{r} \partial c_k X_2^k + \sum_{k=0}^{r} c_k \partial X_2^k . \tag{2.77}$$

Within the d-ideal we can replace any derivations ∂X_i with $p_i(\boldsymbol{X})$ so that we get a classical polynomial again.

Equation (2.76) gives evidence that any monomial of degree r or higher can be replaced by a polynomial of degree $\tilde{r}$, so

$$\partial h_{[m]} \sim \sum_{k=0}^{\tilde{r}} \tilde{c}_k X_2^k . \tag{2.78}$$

Since $\partial h_{[m]} \in \mathfrak{I}$, so is $(X_2 \partial h_{[m]})$ and the right hand side is of degree $\tilde{r}+1$. Multiplying with X_2 sufficiently often yields a polynomial of degree $r-1$. Therefore, there exists a

$$h_{[m+1]} \sim X_2^{r-1-\tilde{r}} \partial h_{[m]} , \tag{2.79}$$

with $h_{[m+1]} \in \mathfrak{I}$ of degree $r-1$.

We have constructed a sequence of polynomials in the d-ideal $\mathfrak{I}$ with degree decreasing by one in each step, until we reach degree zero. In this manner, we can eliminate $X_3, \ldots, X_N$ step by step and finally find some h^* which is in $\mathbb{K}\langle y \rangle$ and also in $\mathfrak{I}$. The denominator of $h^* = \hat{h}^* / \tilde{h}^*$ can be interpreted as a d-polynomial $\hat{h}^*(Y) \in \mathbb{K}\{Y\}$ for which we find

$$\hat{h}^*(y) \sim 0 \tag{2.80}$$

hence y is d-algebraic over $\mathbb{K}$.

■

Proposition 2 *Let $\mathbb{L}/\mathbb{K}$ be a d-field extension. If $y \in \mathbb{L}$ is d-algebraic over $\mathbb{K}$, then there is a system over $\mathbb{K}$ that generates y.*

Proof The proof was already presented before the generating system was introduced: Say there is some higher order d-polynomial $f(Y) \in \mathbb{K}\{Y\}$ with

$$f(y) = 0 . \tag{2.81}$$

We have seen that there is a set of order one d-polynomials

$$q_i(X_i; \boldsymbol{X}) = X_i^{(1)} - p_i(\boldsymbol{X}) \tag{2.82}$$

with $X_1 = y$ and a rational $p_i(\boldsymbol{X})$. Then we have shown that the rational p_i can be made polynomial.

■

The two propositions are already hinting towards a way of treating the invertibility problem: If we were able to proof that a model error, more precisely the d-indeterminate U_i which represents the i^{th} component of the input vector, is d-algebraic over some known d-field, then there existed a generating system with an estimate of the model error as its output. Say, the nominal model is formulated as an ordinary system in the state d-indeterminates $\boldsymbol{X}$ plus some input d-indeterminates $\boldsymbol{U}$ over a d-field $\mathbb{K}$, and $\boldsymbol{Y} \subseteq \boldsymbol{X}$ represent the output of the system. Then the d-field $\mathbb{K}\langle \boldsymbol{Y} \rangle$ incorporates all known or measured quantities. Hence the generating system for U_i does not have to be a system over $\mathbb{K}$, but we can use the observables and construct a generating system over $\mathbb{K}\langle \boldsymbol{Y} \rangle$. A comparable statement about the existence of left-inverse systems for invertible dynamic systems can be found in the seminal Fliess paper [38].

2.2.3 Differential-Transcendence Degree of Ordinary Systems

Ordinary systems serve as the standard form for d-algebraic systems. They are the direct translation of ordinary state space systems and serve as nominal models. In state space, the Picard-Lindelöf theorem ensures the existence and uniqueness of the flow of such a system: The behaviour of the system is completely determined by the initial value, and clearly by the governing differential equations.

The fact that there is nothing arbitrary in the behaviour of the system corresponds to a vanishing *d-transcendence degree* in the d-algebraic formulation. Vividly spoken, the transcendence degree can be seen as a *measure for arbitrariness.* Ordinary systems lack this arbitrariness as we will show below. The introduction of an unknown input U_i is accompanied by an increment of the d-transcendence degree while the introduction of an output Y_j can, but does not need to decrease the d-transcendence degree.

Simple d-Algebraic Field Extensions A first lemma in our proof that ordinary systems lack any algebraic transcendence is the following result about extensions of d-fields with an d-algebraic element. The classical version is well known, see for instance a standard textbook on algebra [74]. The proof facilitates the connection between finite and algebraic field extensions, which is missing in the d-algebraic setting. However, the fact that an d-algebraic element can be replaced by its generating system helps out.

Lemma 1 *Let l be a d-algebraic element of the d-field extension $\mathbb{L}/\mathbb{K}$. Then $\mathbb{K}\langle l \rangle/\mathbb{K}$ is d-algebraic.*

Proof Since l is d-algebraic, according to proposition 2 there is a generating system with d-indeterminates $\boldsymbol{X}$ and d-polynomials $q_1, \dots, q_N$ over $\mathbb{K}$ with

$$l \sim X_1 . \tag{2.83}$$

Now let

$$m \in \mathbb{K}\langle l \rangle . \tag{2.84}$$

Utilising equation (2.83) we replace l with X_1 to get a d-rational $r(X_1) \in \mathbb{K}\langle X_1 \rangle$. With respect to the d-ideal $\Im(q_1, \dots, q_N)$ we find that

$$p_0(\boldsymbol{X}) := \partial r(X_1) \tag{2.85}$$

can be written as a polynomial of order zero. We additionally introduce the d-indeterminate X_0 and

$$q_0 := X_0^{(1)} - p_0(\boldsymbol{X}) \tag{2.86}$$

and find that $m \sim X_0$ with respect to $\Im(q_0, \dots, q_N)$. We have constructed a system that generates m, so m is d-algebraic over $\mathbb{K}$ due to proposition 1.

■

Differential Tower Formula A **d-field tower** $\mathbb{M}/\mathbb{L}/\mathbb{K}$ is a nested d-field extension, i.e., $\mathbb{M}/\mathbb{L}$ and $\mathbb{L}/\mathbb{K}$ are d-field extensions. As an example, consider a d-field $\mathbb{K}$ and elements $a_1, \dots, a_n$ of some d-field extension $\mathbb{L}/\mathbb{K}$. The extension $\mathbb{K}\langle a_1, \dots, a_n\rangle/\mathbb{K}$ can also be interpreted as a tower of simple extensions

$$\mathbb{K}\langle a_1, \dots, a_n\rangle = \mathbb{K}\langle a_1, \dots, a_n\rangle/\mathbb{K}\langle a_1, \dots, a_{n-1}\rangle/\dots/\mathbb{K}\langle a_1, a_2\rangle/\mathbb{K}\langle a_1\rangle/\mathbb{K} \tag{2.87}$$

that is, $\mathbb{K}$ is first extended by a_1, then $\mathbb{K}\langle a_1\rangle$ is extended by a_2 and so forth. The following lemma proves the plausible fact that any tower of d-algebraic d-field extensions is again d-algebraic.

Lemma 2 *Let $\mathbb{M}/\mathbb{L}$ and $\mathbb{L}/\mathbb{K}$ be two d-algebraic d-field extensions. Then $\mathbb{M}/\mathbb{K}$ is also d-algebraic.*

Proof Let $m \in \mathbb{M}$. Since m is d-algebraic over $\mathbb{L}$, there are coefficients $d_\alpha \in \mathbb{L}$ with α from a finite set $\mathscr{A}$ of multi-indices, such that

$$p(M) = \sum_{\alpha \in \mathscr{A}} d_\alpha M^\alpha \tag{2.88}$$

has m as a root. Since each d_α is d-algebraic over $\mathbb{K}$, according to proposition 2 there is a generating system over $\mathbb{K}$ for each d_α. Let ${}_\alpha\boldsymbol{X} = ({}_\alpha X_1, \dots, {}_\alpha X_{N_\alpha})$ and ${}_\alpha q_1, \dots, {}_\alpha q_{N_\alpha}$ characterise the generating system for d_α. Inserting $d_\alpha \sim {}_\alpha X_1$ into $p(M)$ yields

$$\tilde{p}(M) := \sum_{\alpha \in \mathscr{A}} {}_\alpha X_1 M^\alpha \tag{2.89}$$

and with respect to the d-ideal induced by all the d-polynomials ${}_\alpha q_i$ of the generating systems, we find that

$$p_0({}_\alpha\boldsymbol{X} \,|\, \alpha \in \mathscr{A}) := \partial\tilde{p}(M) \tag{2.90}$$

can be written as a polynomial in $\mathbb{K}\langle {}_\alpha\boldsymbol{X} \,|\, \alpha \in \mathscr{A}\rangle\{M\}$. Introducing X_0 and

$$q_0 := X_0^{(1)} - p_0({}_\alpha\boldsymbol{X} \,|\, \alpha \in \mathscr{A}) \tag{2.91}$$

leads to a generating system over $\mathbb{K}$ for m. Hence, $m \in \mathbb{M}$ is d-algebraic over $\mathbb{K}$.

■

One will realise that the proof of the latter theorem has some overlap to the proof of lemmas 1 and 3, which clearly lies in the fact that whenever we encounter a d-algebraic element, we can replace it with its generating system.

One last lemma shall be presented before the d-tower formula is formulated. The following lemma helps to handle the case of simple field extensions with a d-transcendental element.

Lemma 3 *Let $\mathbb{M}/\mathbb{L}/\mathbb{K}$ be a d-field tower and $\mathbb{L}/\mathbb{K}$ a d-algebraic d-field extension. Let furthermore $b \in \mathbb{M}$ be d-transcendental over $\mathbb{K}$. Then $\mathbb{L}\langle b\rangle/\mathbb{K}\langle b\rangle$ is again a d-algebraic d-field extension.*

Proof Let

$$m \in \mathbb{L}\langle b\rangle \tag{2.92}$$

which can be written as

$$m = \frac{\hat{m}}{\tilde{m}} \tag{2.93}$$

where $\hat{m}$ and $\tilde{m}$ are d-polynomials of the form

$$\hat{m} = \sum_{\alpha\in\mathscr{A}} \hat{d}_\alpha b^\alpha \tag{2.94}$$

and

$$\tilde{m} = \sum_{\beta\in\mathscr{B}} \tilde{d}_\beta b^\beta \tag{2.95}$$

where $\hat{d}_\alpha, \tilde{d}_\beta \in \mathbb{L}$. The following argumentation holds analogously for $\tilde{m}$. Since each $\hat{d}_\alpha$ is d-algebraic over $\mathbb{K}$, according to proposition 2, there is a generating system with ${}_\alpha\hat{q}_i, i = 1,\dots,\hat{N}_\alpha$ and

$$\hat{d}_\alpha \sim_\alpha \hat{X}_1 . \tag{2.96}$$

We define the d-polynomial

$$u := \sum_{\alpha\in\mathscr{A}} {}_\alpha\hat{X}_i b^\alpha \tag{2.97}$$

which is a d-polynomial in $\mathbb{K}\langle b, {}_\alpha\hat{X}_1 | \alpha\in\mathscr{A}\rangle$. Analogously we find a

$$v = \mathbb{K}\langle b, {}_\beta\tilde{X}_1 | \beta\in\mathscr{B}\rangle . \tag{2.98}$$

So for $w := u/v$ we have

$$w \in \mathbb{K}\langle b, {}_\alpha\hat{\boldsymbol{X}}, {}_\beta\tilde{\boldsymbol{X}} | \alpha\in\mathscr{A}, \beta\in\mathscr{B}\rangle . \tag{2.99}$$

We can formulate the ordinary d-algebraic system over the field $\mathbb{K}\langle b\rangle$ consisting of all ${}_\alpha\hat{q}_i, {}_\beta\tilde{q}_i$ and

$$m \sim w , \tag{2.100}$$

which, according to proposition 1, is sufficient to say that there exists a d-polynomial $f(M) \in \mathbb{K}\langle b\rangle\{M\}$ with

$$f(m) \sim 0 . \tag{2.101}$$

Hence, m is d-algebraic over $\mathbb{K}\langle b\rangle$.

■

So far we have seen that the simple d-field extension $\mathbb{K}\langle a\rangle/\mathbb{K}$ with an d-algebraic element a can only produce new d-algebraic elements. If b is d-transcendental over $\mathbb{K}$ it is clear that $\mathbb{K}\langle b\rangle/\mathbb{K}$ is not d-algebraic. However, we have also seen that if in the d-algebraic extension $\mathbb{K}\langle a\rangle/\mathbb{K}$ both sides are extended with b to $\mathbb{K}\langle a,b\rangle/\mathbb{K}\langle b\rangle$, the whole d-field extension remains d-algebraic. The d-transcendence degree helps to quantify the problem of transcendences.

Definition 12 *Let $\mathbb{L}/\mathbb{K}$ be a d-field extension and $\boldsymbol{l} = (l_1,\dots,l_n)$ with $l_i \in \mathbb{L}$. The elements of $\boldsymbol{l}$ are called* ***d-algebraically dependent*** *over $\mathbb{K}$ if there is a (non-zero) d-polynomial $p \in \mathbb{K}\{L_1,\dots,L_n\}$ such that*

$$p(l_1,\dots,l_n) = 0 . \tag{2.102}$$

If no such d-polynomial exists, $\boldsymbol{l}$ is called ***d-algebraically independent****.*

If $\boldsymbol{l} = (l_1,\dots,l_n)$ is the largest d-algebraically independent set, that is, if any $\boldsymbol{l}' = (l'_1,\dots,l'_n,l'_{n+1})$ is d-algebraically dependent, then $\boldsymbol{l}$ is called a ***d-transcendence basis*** *and n the d-transcendental* *degree of the d-field extension,*

$$\text{d-tr-d}\,\mathbb{L}/\mathbb{K} = n\,. \tag{2.103}$$

Vividly spoken, the d-transcendence indicates how many simple field extensions are necessary to make a d-field extension d-algebraic, as

$$\mathbb{L}/\mathbb{K}\langle l_1,\dots,l_n\rangle \tag{2.104}$$

is d-algebraic if and only if $(l_1,\dots,l_n)$ is a d-transcendence basis [66, 67].

The following theorem shall close the d-algebraic toolbox and finally allow us to tackle d-algebraic invertibility.

Theorem 1 *Let* $\mathbb{M}/\mathbb{L}/\mathbb{K}$ *be a d-field tower, then*

$$\text{d-tr-d}\,\mathbb{M}/\mathbb{K} = \text{d-tr-d}\,\mathbb{M}/\mathbb{L} + \text{d-tr-d}\,\mathbb{L}/\mathbb{K}\,. \tag{2.105}$$

Proof Let $A = \{a_1,\dots,a_R\}$ be a d-transcendence basis of $\mathbb{M}/\mathbb{K}$, $B = \{b_1,\dots,b_S\}$ a d-transcendence basis of $\mathbb{M}/\mathbb{L}$ and $C = \{c_1,\dots,c_T\}$ a d-transcendence basis of $\mathbb{L}/\mathbb{K}$.

We know that $\mathbb{M}/\mathbb{L}\langle B\rangle$ is d-algebraic. Also $\mathbb{L}/\mathbb{K}\langle C\rangle$ is d-algebraic, and due to lemma 3, $\mathbb{L}\langle B\rangle/\mathbb{K}\langle B,C\rangle$ is d-algebraic. From lemma 2 we know that $\mathbb{M}/\mathbb{K}\langle B,C\rangle$ is algebraic. Since a d-transcendence basis A is the smallest set such that $\mathbb{M}/\mathbb{K}\langle A\rangle$ is d-algebraic, we find

$$R \le S + T\,. \tag{2.106}$$

Now it is clear that C is a d-algebraically independent set over $\mathbb{K}$: Assume C is d-algebraically dependent over $\mathbb{K}$. Without loss of generality, say, $\mathbb{K}\langle C\rangle/\mathbb{K}\langle c_2,\dots,c_T\rangle$ is d-algebraic. But $\mathbb{L}/\mathbb{K}\langle C\rangle$ is d-algebraic by construction, so lemma 2 tells us that $\mathbb{L}/\mathbb{K}\langle c_2,\dots,c_T\rangle$ is d-algebraic. This again shows that any d-transcendence basis of $\mathbb{L}/\mathbb{K}$ has at most $T-1$ elements. But that would contradict the initial assumption that C is a d-transcendence basis.

Now assume $C \cup \{b_1\}$ would be d-algebraically independent. Then b_1 would be d-algebraic over $\mathbb{K}\langle C\rangle \subseteq \mathbb{L}$. But b_1 belongs to a d-transcendence basis over $\mathbb{L}$ and we have already seen, that B must be d-algebraically independent over $\mathbb{L}$. This implies that b_1 must be d-transcendental over $\mathbb{L}$. Consequently, $B \cup C$ is a d-algebraically independent set over $\mathbb{K}$ and therefore cannot be larger than any d-transcendence basis of $\mathbb{M}/\mathbb{K}$, hence

$$S + T \le R\,. \tag{2.107}$$

Therefore

$$R = S + T \tag{2.108}$$

or in terms of d-transcendence degrees

$$\text{d-tr-d}\,\mathbb{M}/\mathbb{K} = \text{d-tr-d}\,\mathbb{M}/\mathbb{L} + \text{d-tr-d}\,\mathbb{L}/\mathbb{K}\,. \tag{2.109}$$

■

Application to Ordinary Systems Let us now apply the lemmas and theorems that we have developed so far to prove that ordinary systems are indeed free of transcendence.

Theorem 2 *The transcendence degree of an ordinary d-algebraic system is zero.*

Proof Say the ordinary d-algebraic system over $\mathbb{K}$ comprises $\boldsymbol{X} = (X_1, \dots, X_N)$ and d-polynomials $q_1, \dots, q_N$. This system can be interpreted as generating system for X_1, or after redefining the index, as generating system for any $X_i \in \boldsymbol{X}$. From proposition 1 it follows, that each X_i is d-algebraic over $\mathbb{K}$, with respect to $\Im(q_1, \dots, q_N)$. Write the d-field extension $\mathbb{K}\langle \boldsymbol{X} \rangle_\sim / \mathbb{K}$ as a tower

$$\mathbb{K}\langle \boldsymbol{X} \rangle_\sim / \mathbb{K} = \mathbb{K}\langle X_1, \dots, X_N \rangle_\sim / \mathbb{K}\langle X_1, \dots, X_{N-1} \rangle_\sim / \dots / \mathbb{K}\langle X_1, X_2 \rangle_\sim / \mathbb{K}\langle X_1 \rangle_\sim / \mathbb{K}. \tag{2.110}$$

and find

$$\text{d-tr-d} \frac{\mathbb{K}\langle X_1, \dots, X_k \rangle_\sim}{\mathbb{K}\langle X_1, \dots, X_{k-1} \rangle_\sim} = 0 \tag{2.111}$$

since each X_k is d-algebraic. Utilising the tower formula 1 we find

$$\text{d-tr-d} \frac{\mathbb{K}\langle \boldsymbol{X} \rangle_\sim}{\mathbb{K}} = 0. \tag{2.112}$$

■

2.3 Differential-Algebraic Invertibility

The elaborated discussion of ordinary d-algebraic systems has yielded tools to handle dynamic systems in an algebraic way. Let us now apply these tools to tackle the problem of unknown inputs.

Definition 13 *Let $\boldsymbol{X} = (X_1, \dots, X_N)$ be state-indeterminates, $\boldsymbol{U} = (U_1, \dots, U_M)$ input-indeterminates and $\boldsymbol{Y} \subseteq \boldsymbol{X}$ output-indeterminates, and*

$$q_i(X_i; \boldsymbol{X}, \boldsymbol{U}) = X_i^{(1)} - p_i(\boldsymbol{X}, \boldsymbol{U}) \tag{2.113}$$

*with $p_i(\boldsymbol{X}, \boldsymbol{U}) \in \mathbb{K}[\boldsymbol{X}, \boldsymbol{U}]$. This constitutes a **d-algebraic input-output system**.*

The reason why the d-field extension $\mathbb{K}\langle \boldsymbol{X} \rangle_{\sim q_1, \dots, q_N} / \mathbb{K}$ (which one can see as a compact notation for an ordinary system) has a vanishing d-transcendence degree lies in the d-ideal $\Im(q_1, \dots, q_N)$. The introduction of a d-indeterminate, consider for instance $\mathbb{K}\langle X_1 \rangle / \mathbb{K}$, increases the d-transcendence degree by one. The governing equation, expressed by $q_1(X_1; \boldsymbol{X})$, reduces the transcendence degree again. An input-output system has M d-indeterminates more than it has governing equations, hence it becomes already clear, that

$$\text{d-tr-d} \frac{\mathbb{K}\langle \boldsymbol{U}, \boldsymbol{X} \rangle_\sim}{\mathbb{K}} = M. \tag{2.114}$$

One will now understand the role of the outputs from the d-algebraic point of view: The outputs $\boldsymbol{Y}$ serve as *known* elements to reduce the degree of transcendence.

Definition 14 *An d-algebraic input-output system is called* ***d-algebraically invertible*** *if*

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{U},\boldsymbol{X}\rangle_{\sim}}{\mathbb{K}\langle \boldsymbol{Y}\rangle} = 0\,. \tag{2.115}$$

Fliess deduced an alternative condition for d-algebraic invertibility [38]. His theorem is briefly presented in a consistent notation.

Theorem 3 (Fliess) *A system is invertible if and only if*

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{Y}\rangle_{\sim}}{\mathbb{K}} = M\,. \tag{2.116}$$

Proof For the tower $\mathbb{K}\langle \boldsymbol{U},\boldsymbol{X},\boldsymbol{Y}\rangle_{\sim}/\mathbb{K}\langle \boldsymbol{Y}\rangle_{\sim}/\mathbb{K}$ we find

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{U},\boldsymbol{X},\boldsymbol{Y}\rangle_{\sim}}{\mathbb{K}} = \text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{U},\boldsymbol{X},\boldsymbol{Y}\rangle_{\sim}}{\mathbb{K}\langle \boldsymbol{Y}\rangle_{\sim}} + \text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{Y}\rangle_{\sim}}{\mathbb{K}}\,. \tag{2.117}$$

We also find for the tower $\mathbb{K}\langle \boldsymbol{U},\boldsymbol{X},\boldsymbol{Y}\rangle_{\sim}/\mathbb{K}\langle \boldsymbol{U}\rangle/\mathbb{K}$ that

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{U},\boldsymbol{X},\boldsymbol{Y}\rangle_{\sim}}{\mathbb{K}} = \text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{U},\boldsymbol{X},\boldsymbol{Y}\rangle_{\sim}}{\mathbb{K}\langle \boldsymbol{U}\rangle} + \text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{U}\rangle}{\mathbb{K}}\,. \tag{2.118}$$

An input-output system over $\mathbb{K}$ is ordinary over $\mathbb{K}\langle \boldsymbol{U}\rangle$, hence

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{U},\boldsymbol{X},\boldsymbol{Y}\rangle_{\sim}}{\mathbb{K}\langle \boldsymbol{U}\rangle} = 0\,. \tag{2.119}$$

It is also clear that

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{U}\rangle}{\mathbb{K}} = M \tag{2.120}$$

as this is just a d-field extension with M d-indeterminates. Altogether we find

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{U},\boldsymbol{X},\boldsymbol{Y}\rangle_{\sim}}{\mathbb{K}\langle \boldsymbol{Y}\rangle_{\sim}} + \text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{Y}\rangle_{\sim}}{\mathbb{K}} = M\,. \tag{2.121}$$

By definition

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{U},\boldsymbol{X},\boldsymbol{Y}\rangle_{\sim}}{\mathbb{K}\langle \boldsymbol{Y}\rangle_{\sim}} = 0 \tag{2.122}$$

if and only if the system is d-algebraically invertible, thus

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{Y}\rangle_{\sim}}{\mathbb{K}} = M \tag{2.123}$$

if and only if the system is d-algebraically invertible.

■

CHAPTER 3

Structural Invertibility

Recalling the very abstract understanding of a dynamic system as entities and interactions between them, one will come to the probably most minimalistic model to describe this system: The representation as *structural system* frees itself from the question *how* the entities of a system interact and merely takes the *influence structure* into account.

Network Structure The structure of equations (2.1) and (2.3) which describe the exponential function takes a very simple form:

$$x \circlearrowleft$$

which expresses, that only the current value of x influences its change $\dot{x}$. The structural approach leaves indeed much space for interpretation. If the loop is intensifying, for instance, we get a vicious circle, whose amplitude grows exponentially, or, if the loop is mitigating, it represents an exponential decay. The exponential growth and decay present two qualitatively different phenomena, however, the influence structure is one and the same. Given that, the question might arise, whether the structural representation contains enough information about the system to make mathematically rigorous statements at all.

Structural Properties Starting in 1974 with Lin's proof that *structural controllability* [75] is indeed able to make useful statements about dynamic control systems, the exploration of structural properties pops up in the literature every now and then, see [35] for an engineering textbook including discussions of structural observability and controllability. The currently increasing interest on network science lead to a transfer of structural properties, originally proven for dynamic systems, to general networks, see for instance [76, 77, 78, 79] for overviews, also [80, 81], and [82, 83, 84]. For a critical discussion on the significance of structural properties see also [85, 86] and [30].

Structural Invertibility The main result of this chapter will show that the structural and algebraic properties of a dynamic system are closely connected, inasmuch as we show that the d-transcendence degree of an input-output system can be deduced from its influence structure. It can therefore be seen as a key result that connects the analytical and d-algebraic conditions, which are mathematically rigorous but hard to verify in practice, with conditions on the structural or network level. This connection also builds the basis of chapters 4 and 5.

3.1 Structural Systems

The structural approach employs the language of *graph theory*, see [87] for a textbook. In the graph theoretical framework, the i^{th} entity of the system corresponds to a *node* $\mathtt{i}$ of a graph. This node plays the role of the state variable x_i in state space or of the d-indeterminate X_i in the d-algebraic representation. Assume that in state space we find

$$\frac{\partial f_j}{\partial x_i} \neq 0 \tag{3.1}$$

or equivalently in the d-algebraic system

$$\frac{\partial p_j}{\partial X_i} \neq 0\,. \tag{3.2}$$

This means that the value $x_i(t)$ has a direct influence on the change $\dot{x}_j(t)$. The structural approach lacks an equation to describe this interaction but is restricted to the statement

$$\mathtt{i} \to \mathtt{j} \tag{3.3}$$

meaning $\mathtt{i}$ has a direct influence on $\mathtt{j}$.

Definition 15 *A* ***graph*** $\mathtt{g} = (\mathtt{N},\mathtt{E})$ *is understood as a set of* ***nodes*** $\mathtt{N}$ *together with a set of* ***edges*** $\mathtt{E} \subseteq \mathtt{N} \times \mathtt{N}$. *An edge* $\mathtt{e} \in \mathtt{E}$ *takes the form* $\mathtt{e} = (\mathtt{a} \to \mathtt{b})$ *with* $\mathtt{a},\mathtt{b} \in \mathtt{N}$.

Let furthermore $\mathtt{S},\mathtt{Z} \subseteq \mathtt{N}$ *be two sets, the* ***input set*** *and* ***output set****, respectively. We understand the doublet*

$$(\mathtt{g},\mathtt{Z}) \tag{3.4}$$

as an ***ordinary structural system****. The triple*

$$(\mathtt{S},\mathtt{g},\mathtt{Z}) \tag{3.5}$$

is understood as an ***structural input-output system****.*

Whereas the state space formulation yields the geometric interpretation of the system's time course as trajectory through space, and the d-algebraic formulation yields the classification as transcendental or algebraic model errors, the structural approach equips invertibility problem with a new picture of the scene. The representation of a dynamic system in form of an *influence graph* is less common in pure mathematics but mainly exploited in the engineering literature, see for instance [35].

Information Flow The structural approach lacks a quantitative description of the interactions in a system but it rather suggests the following interpretation: Some *information* (for instance about the model error) enters the system at the input nodes $\mathtt{s}_1,\ldots,\mathtt{s}_M \in \mathtt{S}$. This information is passed by the interactions, i.e., along the edges from one node to the following nodes and by this percolates through the system. We assimilate the information output at the nodes $\mathtt{z}_1,\dots,\mathtt{z}_P \in \mathtt{Z}$. The question is now: Can we reconstruct *paths* over which the information flowed through the system to infer the input set S?

Definition 16 *Let* $\mathtt{g} := (\mathtt{N},\mathtt{E})$ *be a graph. A* ***path*** π *is understood as*

$$\pi = (\mathtt{a}_0 \to \dots \to \mathtt{a}_L) \tag{3.6}$$

where for $l = 1,\dots,L$ *each* $\mathtt{a}_l \in \mathtt{N}$ *is a node and each* $(\mathtt{a}_{l-1} \to \mathtt{a}_l) \in \mathtt{E}$ *is an edge of* $\mathtt{g}$*. The* ***length*** *of this path is*

$$\operatorname{len}\pi = L, \tag{3.7}$$

the ***initial node***

$$\operatorname{in}\pi = \mathtt{a}_0 \tag{3.8}$$

and the ***terminal node***

$$\operatorname{ter}\pi = \mathtt{a}_L\,. \tag{3.9}$$

Let $\mathtt{a},\mathtt{b} \in \mathtt{N}$ *be two nodes. The set of* ***paths of length k from*** $\mathtt{a}$ ***to*** $\mathtt{b}$ *is understood as*

$$\mathscr{P}_k(\mathtt{a},\mathtt{b}) := \{\pi \,|\, \operatorname{len}\pi = k, \operatorname{in}\pi = \mathtt{a}, \operatorname{ter}\pi = \mathtt{b}\} \tag{3.10}$$

and all ***paths from*** $\mathtt{a}$ ***to*** $\mathtt{b}$

$$\mathscr{P}(\mathtt{a},\mathtt{b}) := \bigcup_{k=0}^{\infty} \mathscr{P}_k(\mathtt{a},\mathtt{b})\,. \tag{3.11}$$

Let $\pi = (\mathtt{a}_0 \to \dots \to \mathtt{a}_L)$ *and* $\rho = (\mathtt{b}_0 \to \dots \to \mathtt{b}_M)$ *be two paths with* $\operatorname{ter}\pi = \operatorname{in}\rho$*. The* ***concatenation*** *of* π *and* ρ *yields a new path*

$$\pi \circ \rho := (\mathtt{a}_0 \to \dots \to \mathtt{a}_L \to \mathtt{b}_1 \to \dots \to \mathtt{b}_M)\,. \tag{3.12}$$

Structural Invertibility Exploiting the structural system terminology, let us first define *structural invertibility* on a purely graph theoretical level before we proof the (almost) equivalence of structural and d-algebraic invertibility. *Linked sets* in a graph have earlier been connected to input reconstruction [53, 51].

Definition 17 *Let* $\mathtt{g} = (\mathtt{N},\mathtt{E})$ *be a graph and* $\mathtt{S},\mathtt{Z} \in \mathtt{N}$*. We say* $\mathtt{S}$ ***is linked in*** $\mathtt{g}$ ***to*** $\mathtt{Z}$*, if there is a family* Π *of node-disjoint paths with* $\operatorname{in}\Pi = \mathtt{S}$ *and* $\operatorname{ter}\Pi = \mathtt{Z}$*. We say* $\mathtt{S}$ ***is linked in*** $\mathtt{g}$ ***into*** $\mathtt{Z}$*, if there is a family* Π *of node-disjoint paths with* $\operatorname{in}\Pi = \mathtt{S}$ *and* $\operatorname{ter}\Pi \subseteq \mathtt{Z}$.

The concept of linked sets can be treated as a purely graph theoretical problem, see for instance [88] for an early paper on the subject and the references in chapter 5. With the interpretation of $(\mathtt{S},\mathtt{g},\mathtt{Z})$ as structural system, the linkage of the input and output set can be reinterpreted as invertibility.

Definition 18 *Let* $(\mathtt{S},\mathtt{g},\mathtt{Z})$ *be a structural system. The system is called* ***structurally invertible****, if* $\mathtt{S}$ *is linked in* $\mathtt{g}$ *into* $\mathtt{Z}$.

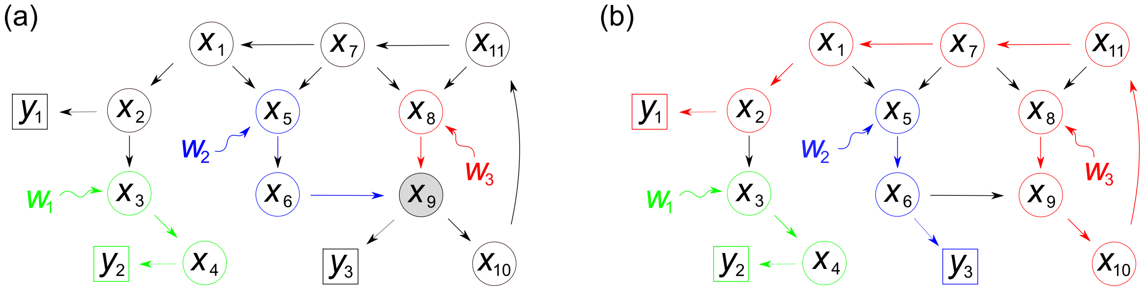

Figure 3.1: Structural invertibility analysis for the two systems from figure 1.3. (a) A system with three inputs w_1, w_2, w_3 and three outputs y_1, y_2, y_3. All paths from w_2 and w_3 to the outputs intersect at node x_9. A node-disjoint linking of inputs and outputs is not possible and the system structurally non-invertible. (b) After replacing the output y_3, there exists a node-disjoint linking as indicated by the colouring. The system is structurally invertible.

3.2 Necessity of Structural Invertibility

The following two theorems show that structural invertibility is a necessary condition for d-algebraic ivertibility. They have first been formulated by Wey [51]. These theorems seem to be the only results on this subject. The first theorem is given to give a better picture of Wey's results. However, it is of less importance for the purposes of this thesis and therefore presented without proof. The second theorem proofs the necessity of structural invertibility. Here, a new proof is given which utilises the d-algebraic tools presented so far.

Theorem 4 (Wey I) *Consider a dynamic input-output system with differential output rank*

$$\rho := \text{d-tr-d}\,\mathbb{K}\langle \boldsymbol{Y}\rangle_{\sim}/\mathbb{K}\,. \tag{3.13}$$

Let g *be the influence graph of the system with input set* S *and output set* Z *and let* ρ^* *denote the maximum amount of node-disjoint paths from* S *to* Z*. Then*

$$\rho \leq \rho^*\,. \tag{3.14}$$

Theorem 5 (Wey II) *Consider an input-output system in d-algebraic form*

$$q_i(X_i;\boldsymbol{X},\boldsymbol{U}) = X_i^{(1)} - p_i(\boldsymbol{X},\boldsymbol{U}) \tag{3.15}$$

and structural form

$$(\mathsf{S},\mathsf{g},\mathsf{Z})\,. \tag{3.16}$$

The system is d-algebraically invertible only if it is structurally invertible.

Proof First, note that by Menger's theorem [88] the number of node-disjoint paths between two sets equals the size of the smallest *separator* of these sets.

Assume the system is structurally not-invertible. That means, the maximal number of node-disjoint paths from S to Z is $m < M$ where $M = \operatorname{card} S$. This indicates that we can separate the nodes into $\mathtt{N} = \mathtt{N}_1 \dot{\cup} \mathtt{N}_2 \dot{\cup} \mathtt{N}_3$ such that $\mathtt{S} \subseteq \mathtt{N}_1$, $\mathtt{Z} \in \mathtt{N}_3$ and such that $\mathtt{N}_2$ is the separator of size m. There is no edge $\mathtt{i} \to \mathtt{j} \in \mathtt{E}$ with $\mathtt{i} \in \mathtt{N}_1$ and $\mathtt{j} \in \mathtt{N}_3$.

Let $\boldsymbol{X}_\alpha = (X_k \mid k \in \mathtt{N}_\alpha)$ be the vectors of d-indeterminates that correspond to the three node sets. We find that for $\mathtt{i} \in \mathtt{N}_2 \dot{\cup} \mathtt{N}_3$ the d-polynomials have the form

$$q_i(X_i; \boldsymbol{X}_2, \boldsymbol{X}_3) = X_i^{(1)} - p_i(\boldsymbol{X}_2, \boldsymbol{X}_3) \tag{3.17}$$

and

$$\boldsymbol{Y} \subseteq \boldsymbol{X}_3 \,. \tag{3.18}$$

So $\mathscr{N}_2 \dot{\cup} \mathscr{N}_3$ is a subsystem. More than that, this subsystem is ordinary over $\boldsymbol{X}_2$ hence

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{Y} \rangle_\sim}{\mathbb{K}} = m < M \,. \tag{3.19}$$

■

3.3 Sufficiency of Structural Invertibility

While the necessity of structural invertibility has been proven earlier, its sufficiency seems to be an open question until today. For this purpose, the technique of a *propagating transcendence basis* is suggested.

In the interpretation of an information flow, we have said, that some information enters the system at the input set S, flows through the graph g and is then observed at the output set Z. Note that a d-algebraic input-output system over $\mathbb{K}$ can also be interpreted as an ordinary system over $\mathbb{K}\langle \boldsymbol{U} \rangle$, indicating that the input set $\boldsymbol{U}$ is a d-transcendence basis of the system. The example below shows, how this d-transcendence basis propagates along the edges of the graph.

Example (d-transcendence basis propagation)

Consider the d-algebraic system $\mathfrak{S}_0$

$$\begin{aligned} q_1(X_1; U_1) &= X_1^{(1)} - p_1(U_1) \\ q_2(X_2; U_2) &= X_2^{(1)} - p_2(U_2) \\ q_3(X_3; X_1, X_2) &= X_3^{(1)} - p_3(X_1, X_2) \\ q_4(X_4; X_3) &= X_4^{(1)} - p_4(X_3) \\ q_5(X_5; X_3) &= X_5^{(1)} - p_5(X_3) \end{aligned} \tag{3.20}$$

over a d-field $\mathbb{K}$. The observables are given by

$$(Y_1, Y_2) = (X_4, X_5) \,. \tag{3.21}$$

Find the influence graph of this system on the right hand side in figure 3.2.

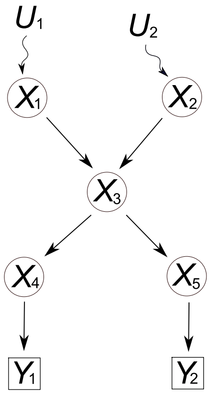

Figure 3.2: The influence graph of the exemplary system 3.20 to demonstrate the propagation of a d-transcendence basis.

Since $\mathfrak{S}_0$ is ordinary over $\mathbb{K}\langle U_1, U_2\rangle$,

$$\boldsymbol{B}_0 := \{U_1, U_2\} \tag{3.22}$$

is a d-transcendence basis of the system.

Now consider the following calculations: We find

$$\partial p_1(U_1) = \sum_k \partial a_k U_1^k + \sum_k a_k k U_1^{k-1} \partial U_1 \tag{3.23}$$

and

$$\frac{-\partial q_1}{\sum_k a_k k U^{k-1}} = U_1^{(1)} - X_1^{(2)} + \sum_k \partial a_k U_1^k \,. \tag{3.24}$$

We set

$$\tilde{p}(X_1, U_1) := X_1^{(2)} - \sum_k \partial a_k U_1^k \tag{3.25}$$

and

$$\tilde{q}_1(U_1; X_1, U_1) := U_1^{(1)} - \tilde{p}_1(X_1, U_1) \,. \tag{3.26}$$

Since q_1 and

$$\tilde{q}_1 = \frac{-\partial q_1}{\sum_k a_k k U^{k-1}} \,, \tag{3.27}$$

only differ by a differentiation and multiplication, the elements q_1 and $\tilde{q}_1$ generate equivalent d-ideals. The system

$$\mathfrak{S}_1 := \tilde{q}_1(U_1; X_1, U_1), q_2(X_2; U_2), q_3(X_3; X_1, X_2), q_4(X_4; X_3), q_5(X_5; X_3) \tag{3.28}$$

is ordinary over $\mathbb{K}\langle X_1, U_2\rangle$, hence

$$\boldsymbol{B}_1 := \{X_1, U_2\} \tag{3.29}$$

a d-transcendence basis.

In complete analogy we find a d-polynomial

$$\tilde{q}_2(U_2; X_2, U_2) = U_2^{(1)} - \tilde{p}_2(X_2, U_2) \tag{3.30}$$

and for

$$\mathfrak{S}_2 := \tilde{q}_1(U_1; X_1, U_1), \tilde{q}_2(U_2; X_2, U_2), q_3(X_3; X_1, X_2), q_4(X_4; X_3), q_5(X_5; X_3) \tag{3.31}$$

the d-transcendence basis

$$\boldsymbol{B}_2 := \{X_1, X_2\} \,. \tag{3.32}$$

We can proceed with the polynomial for X_3,

$$q_3(X_3; X_1, X_2) = \sum_k c_k X_1^k \tag{3.33}$$

where $c_k \in \mathbb{K}\langle X_2 \rangle$. With

$$\tilde{p}_3(X_1, X_2, X_3) := \frac{X_3^{(2)} - \sum_k \partial b_k X_1^k}{\sum_k b_k k X_1^{k-1}} \tag{3.34}$$

and

$$\tilde{q}_3(X_1; X_1, X_2, X_3) := X_1^{(1)} - \tilde{p}_3(X_1, X_2, X_3) \tag{3.35}$$

we find that

$$\mathfrak{S}_3 := \tilde{q}_1(U_1; X_1, U_1), \tilde{q}_2(U_2; X_2, U_2), \tilde{q}_3(X_1; X_1, X_2, X_3), q_4(X_4; X_3), q_5(X_5; X_3) \tag{3.36}$$

is ordinary over $\mathbb{K}\langle X_2, X_3 \rangle$, so

$$\boldsymbol{B}_3 := \{X_2, X_3\} \tag{3.37}$$

is a d-transcendence basis.

The same procedure for q_4 leads to

$$\tilde{q}_4(X_3; X_3, Y_1) = X_3^{(1)} - \tilde{p}_4(X_3, Y_1) \tag{3.38}$$

and to the d-transcendence basis

$$\boldsymbol{B}_4 := \{X_2, Y_1\} \tag{3.39}$$

for

$$\mathfrak{S}_4 := \tilde{q}_1(U_1; X_1, U_1), \tilde{q}_2(U_2; X_2, U_2), \tilde{q}_3(X_1; X_1, X_2, X_3), \tilde{q}_4(X_3; X_3, Y_1), q_5(Y_2; X_3) \tag{3.40}$$

where Y_1 and Y_2 were already inserted for X_4 and X_5.

At this point there is no possibility propagate the d-transcendence basis any further.

The idea which is used repetitively throughout the example above is the following: Say X_i and X_j are two d-indeterminates of an input-output system and there is a governing d-polynomial $q_i(X_i; \boldsymbol{X})$ that renders X_i d-algebraic, whereas X_j remains d-transcendent. By a sent calculation we find that, with respect to the d-ideal of the system, we can transform q_i to $\tilde{q}_i(X_j; \boldsymbol{X})$. We have swapped the roles of X_i and X_j, such that we now have a d-polynomial that makes X_j d-algebraic but in return X_i became d-transcendent. However, this swapping of two d-indeterminates is not always possible.

Direct Dependence First, the polynomial p_i must depend on X_j, i.e.

$$\frac{\partial p_i}{\partial X_j} \neq 0\,. \tag{3.41}$$

If not, $q_i(X_i; \boldsymbol{X})$ does not receive information about X_j directly and the reformulation into $\tilde{q}_i(\tilde{X}_j; \boldsymbol{X})$ cannot be done. As the condition above is exactly the condition for the existence of an edge

$$\mathtt{i} \to \mathtt{j} \tag{3.42}$$

from the structural point of view, one can see this as a *propagation* of the d-transcendence basis along the edges of the structural system.

Sufficient Number of Node-disjoint Paths Second, take a look at $\tilde{q}_4(X_3; X_3, Y_1)$ and $q_5(Y_2; X_3)$ from the example above, where we propagated the d-transcendence basis until $\boldsymbol{B}_4 = (X_2, Y_1)$. Ideally, we want to have the observables (Y_1, Y_2) to be a d-transcendence basis. The only way to put Y_2 into the d-transcendence basis, however, goes via $q_5(Y_2; X_3)$. But swapping X_3 and Y_2 in the described manner would yield $\tilde{q}_5(X_3; X_3, Y_2)$ which is redundant as we already have $\tilde{q}_4(X_3; X_3, Y_1)$ as d-polynomial for X_3. More than that, $\tilde{q}_4(X_3; X_3, Y_1)$ shows that X_3 is d-algebraic over $\mathbb{K}\langle Y_1\rangle$ and $q_5(Y_2; X_3)$ makes Y_2 d-algebraic over $\mathbb{K}\langle X_3\rangle$, always with respect to the generated d-ideal. Lemma 1 and 2 tell us, that consequently Y_2 is d-algebraic over $\mathbb{K}\langle Y_1\rangle$. Hence (Y_1, Y_2) are d-algebraically dependent and cannot belong to the same d-transcendence basis. This issue that we would get two redundant d-polynomials, here $\tilde{q}_4(X_3; X_3, Y_1)$ and $\tilde{q}_5(X_3; X_3, Y_2)$, goes back to the fact, that X_3 is a separator of between the initial d-transcendence basis (U_1, U_2) and the desired one (Y_1, Y_2). Take a look at figure 3.2 to see this.

Pathological Case Third, a pathological case where it is not possible to construct a d-transcendence basis arises when the d-polynomials of the system are d-algebraically dependent. Assume, for instance, (X_1, X_2) is a d-transcendence basis, or a subset of a d-transcendence basis, and

$$\begin{aligned} q_3(X_3; X_1, X_2) &= X_3^{(1)} - f(X_1, X_2) \\ q_4(X_4 : X_1, X_2) &= X_4(1) - g(X_1, X_2) \end{aligned} \tag{3.43}$$

where f and g are d-linearly dependent over $\mathbb{K}\langle X_1, X_2\rangle$. With d-linear dependence we mean that there are two elements in the d-ring of derivation operators $\delta_1, \delta_2 \in \mathbb{K}\langle X_1, X_2\rangle[\partial]$ such that

$$\delta_1(f) + \delta_2(g) = 0\,. \tag{3.44}$$

Then

$$\delta_1(q_3) + \delta_2(q_4) \sim 0 \tag{3.45}$$

is clearly still in the d-ideal and

$$\delta_1(q_3) + \delta_2(q_4) = \delta_1(X_3^{(1)}) + \delta_1(X_4^{(1)}) - \underbrace{(\delta_1(f) + \delta_2(g))}_{=0}\,. \tag{3.46}$$

One will see, that

$$r(X_3, X_4) := \delta_1(X_3^{(1)}) + \delta_1(X_4^{(1)}) \tag{3.47}$$

is a d-polynomial over $\mathbb{K}$ that implies the d-algebraic dependence of (X_3, X_4). Hence, (X_3, X_4) cannot be part of the same d-transcendence basis.

***Almost* Equivalence** The first point can be interpreted as the *propagation* of the d-transcendence basis along the edges of the influence graph. The general formulation will follow in the theorem below. The second point handles the case that there exists a *separator* of a cardinality smaller than the number of inputs. As due to Menger [88] the size of the separator equals the number of node-disjoint paths, the propagation is possible only if the number of node-disjoint paths is large enough. One will realise that this reproduces the necessary condition, theorem 5. The third point remains a pathological case and is responsible for the designation that structural and d-algebraic invertibility are *almost* equivalent.

Theorem 6 *Consider a d-algebraic dynamic system in* $\boldsymbol{X} = (X_1, \ldots, X_N)$ *and* $\boldsymbol{B} = (B_1, \ldots, B_M)$ *over the d-field* $\mathbb{K}$

$$q_i(X_i; \boldsymbol{X}, \boldsymbol{B}) = X_i^{(1)} - p_i(\boldsymbol{X}, \boldsymbol{B}) \tag{3.48}$$

with $p \in \mathbb{K}[\boldsymbol{X}, \boldsymbol{B}]$ *and*

$$\frac{\partial p_1}{\partial B_1} \neq 0 . \tag{3.49}$$

If $\boldsymbol{B}$ *is a d-transcendence basis and if* X_1 *is a d-transcendental element of the d-field extension* $\mathbb{K}\langle X_1, B_2, \ldots, B_M\rangle_\sim / \mathbb{K}\langle B_2, \ldots, B_M\rangle$, *then* $(X_1, B_2, \ldots, B_M)$ *is a d-transcendence basis of the system.*

Proof By assumption p_1 is a non-constant polynomial in $\mathbb{K}\langle \boldsymbol{X}, B_2, \ldots, B_M\rangle[B_1]$, so

$$p_1(B_1) = \sum_{k=0}^{K} a_k B_1^k \tag{3.50}$$

with $a_k \in \mathbb{K}\langle \boldsymbol{X}, B_2, \ldots, B_M\rangle$. Defining

$$\tilde{q}_1 := \frac{-\partial q_1}{\sum_{k=0}^{K} a_k k B_1^{k-1}} \tag{3.51}$$

and

$$\tilde{p}_1 := \frac{\sum_{k=0}^{K} \partial a_k B_1^k}{\sum_{k=0}^{K} a_k k B_1^{k-1}} - X_1^{(2)} \tag{3.52}$$

yields a d-polynomial

$$\tilde{q}_1(B_1; \boldsymbol{X}, \boldsymbol{B}) := B_1^{(1)} - \tilde{p}_1(\boldsymbol{X}, \boldsymbol{B}) . \tag{3.53}$$

One will realise that due to the terms ∂a_k and $X_1^{(2)}$, $\tilde{p}_1$ is not of order zero. However, it is of order zero in B_1. Higher orders in $X_2, \ldots, X_N$ can be remedied by inserting $q_2, \ldots, q_N$. Higher orders in $X_1, B_2, \ldots, B_M$ cause no issue, as we now find the system $\tilde{q}_1, q_2, \ldots, q_N$ to be ordinary over the d-field $\mathbb{K}\langle X_1, B_2, \ldots, B_M\rangle$. The d-ideals induced by $q_1, \ldots, q_N$ and $\tilde{q}_1, q_2, \ldots, q_N$ are the same, hence we can use the d-tower formula

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{X}, \boldsymbol{B}\rangle_\sim}{\mathbb{K}} = \underbrace{\text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{X}, \boldsymbol{B}\rangle_\sim}{\mathbb{K}\langle X_1, B_2, \ldots, B_M\rangle}}_{=0} + \text{d-tr-d}\,\frac{\mathbb{K}\langle X_1, B_2, \ldots, B_M\rangle}{\mathbb{K}} = M \tag{3.54}$$

to see $(X_1, B_2, \ldots, B_M)$ is indeed a d-transcendence basis of the system.

■

3.4 A Note on the Connection between State Space, d-Algebraic, and Structural Systems

The different kinds of models, *state space*, *d-algebra*, and *structural* should not be seen as competing model classes. It should rather be emphasised, that they are different perspectives upon one and the same system. Starting from a state space model, it is always possible to deduce a d-algebraic form, and from the d-algebraic from it is always possible to obtain a structural system. Both steps, however, make a generalisation: The d-algebraic form does not take the initial value into account. The structural system does not incorporate a quantitative description about the interactions.

Hence, there are many state space systems, namely one for each initial value, that will be mapped to the same d-algebraic system. In the same way, many d-algebraic systems will be mapped to the same structural system $\mathtt{Sys}$.

Example: Path Lengths As one example how a property of a system substantiates in the different representations, consider the following connection between *path lengths* in the structural system and *perturbation theory* in state space.

If an edge $(\mathsf{a} \to \mathsf{b})$ represents a direct influence from a to b, then a path $\pi \in \mathscr{P}_n(\mathsf{a},\mathsf{b})$ represents an indirect influence. For simplicity assume there is only one path π between these two nodes. For a discrete-time system one will see that an input on node a influences the behaviour of b after n time steps. Consider also the seminal work by Kalman [89, 90] where this can easily be deduced for linear systems from the *Kalman controllability matrix*.

In continuous-time systems that are described by ordinary differential equations without delay terms or other special elements, also indirect influences happen instantaneously. Meaning that an input starts to perturb a at a time t_0, then this perturbation can be detected at b at time $t_0 + \epsilon$, for any $\epsilon > 0$.

However, say x_b is the state variable that corresponds to the node b and has a Taylor-expansion

$$x_\mathsf{b}(t_0+\epsilon) = x_\mathsf{b}(t_0) + \ldots + \frac{1}{n!}\left.\frac{\mathrm{d}^n x_\mathsf{b}}{\mathrm{d}\,t^n}\right|_{t=t_0} \epsilon^n + \ldots . \tag{3.55}$$

We indeed find that the perturbation appears first at order $\mathcal{O}(\epsilon^n)$ with $n = \operatorname{len}\pi$. If the underlying d-field $\mathbb{K}$ is actually a classical field, it appears only at this order.

CHAPTER 4

Invertibility Structures in Complex Networks

In contrast to state space and d-algebraic models, the influence graph of a system carries the information *which* entities of the system interact but not *how* they interact. Any dynamic system formulated in state space or d-algebraically therefore contains an underlying structural system. In fact, it is likely to happen that two dynamic systems, e.g., a predator-prey like model

$$\begin{aligned} \dot{x}_1(t) &= x_1 - x_1 x_2 \\ \dot{x}_2(t) &= x_1 x_2 - x_2 \end{aligned} \tag{4.1}$$

and a chemical reaction model

$$\begin{aligned} \dot{x}_1(t) &= -x_1 x_2 \\ \dot{x}_2(t) &= +x_1 x_2 \end{aligned} \tag{4.2}$$

describe two completely different problems but with an identical influence structure. The systems are then said to be **topologically equivalent** or simply to have the same structure.

The main result of the former chapter has shown, that the influence structure of a system makes a statement about its invertibility. But this result can also be viewed from another perspective: The structural system is the fundamental model and has numerous *realisations* in d-algebra or state space. All these realisations are topologically equivalent.

Working with a dynamic system, one can take advantage of *structural properties* which are actually inherited from the underlying structural system. Unfortunately, structural properties are often veiled by stiff numerics or apparently complicated vector fields. It is therefore worth an investigation, in how much the exploration of the influence structure of a system can already aid in our aim to reconstruct model errors. The seminal paper [75] was the first hint that structural properties build the bridge between dynamic systems and a rather new field of research: *Network science*. However, network science is applicable to a broad variety of tasks and in no way restricted to dynamic systems any more. In the same way the investigation of the *network principles* (compare [80] and references given in chapter 3) that support or mitigate structural invertibility might turn out to be of greater validity than for the reconstruction of model errors in dynamic systems.

Networks To emphasise the more general scope of this chapter, the *network* character shall be put into focus. Two important classes of networks (or graphs) are the *Erdős-Rényi networks* [91] and *Scale-Free networks* [92]. Both are *random networks* and have been discussed many times throughout the literature, e.g., connected to *structural observability* and *structural controllability*, see [80] for an overview. These two types of networks will help to understand the importance of density, homogeneity, and the role of hubs and satellites in a network. To complement these results, a choice of real networks from various fields is also considered. From these we will learn in how much the principles of structural invertibility are realised in reality.

Experimental Design As the discussion of network structures and invertible configurations is to manifold to give a complete treatment, only one line of reasoning will be demonstrated, which can be considered as three steps towards an *experimental design* and might therefore be of practical relevance.

Scenario I represents a pure *a posteriori* analysis. Meaning, we assume to have a model and data for a certain dynamic system but the data assimilation was done without respecting invertibility at all. In this scenario we choose the input and output nodes *randomly* to simulate a not-designed experiment, and check whether the obtained network is structurally invertible. In this scenario we will also learn about the issues arising from network *hubs* and *satellites*.

In Scenario II a *sensor placement algorithm* will be presented, building on the new gained knowledge. We will see how this sensor node placement algorithm is capable of increasing the probability, that a system is structurally invertible.

Scenario III finally presents the case of a full experimental design, meaning, we can not only place the sensors but also the model errors. This scenario might not often be the case, as one usually does not have control about the model errors, but it can be seen as lucky case and therefore yield an upper bound for realistic situations.

4.1 Random Networks

A *random graph* or *random network* is a graph-valued random variable. We will present two important classes of random graph, *Erdős-Rényi (ER)* and *Scale-Free (SF)* networks and examine their structural invertiblity behaviour.

Definition 19 *Let* $\mathtt{g} = (\mathtt{N}, \mathtt{E})$ *be a graph. For a node* $\mathtt{n} \in \mathtt{N}$, $\mathtt{Pa}(\mathtt{n}) \subseteq \mathtt{N}$ *denotes the set of* ***parent nodes****. A parent node of* $\mathtt{n}$ *is a node* $\mathtt{m}$ *for which the edge* $\mathtt{m} \to \mathtt{n} \in \mathtt{E}$ *exists. The* ***in-degree*** $\mathtt{d}_{\mathtt{in}}(\mathtt{n})$ *is the cardinality of* $\mathtt{Pa}(\mathtt{n})$*. In the same way* $\mathtt{Ch}(\mathtt{n})$ *denotes the* ***child nodes*** $\mathtt{n} \to \mathtt{m} \in \mathtt{E}$ *and* $\mathtt{d}_{\mathtt{out}}(\mathtt{n})$ *the* ***out-degree****. The* ***degree*** *of* $\mathtt{n}$ *is understood as the total degree* $\mathtt{d}(\mathtt{n}) = \mathtt{d}_{\mathtt{in}}(\mathtt{n}) + \mathtt{d}_{\mathtt{out}}(\mathtt{n})$*. The* ***average degree*** $\bar{\mathtt{d}}(\mathtt{g})$ *of the graph is straightforwardly defined as the average of the degrees of all nodes.*

4.1.1 Erdős-Rényi Network

An ER-network [93] can be created by the following algorithm.

Algorithm (Erdős-Rényi Network)

```
Let $0 < p < 1$ and let $\mathtt{N} = \{1, \dots, N\}$ be a set of nodes. Initialise the graph $\mathtt{g} = (\mathtt{N}, \mathtt{E} = \emptyset)$.
For each pair $\mathtt{i}, \mathtt{j} \in \mathtt{N}$ with $\mathtt{i} \neq \mathtt{j}$, add the edge $\mathtt{i} \to \mathtt{j} \in \mathtt{E}$ with a probability of $p$.
```

For a node $\mathtt{n}$, there could be $(N-1)$ outgoing edges and the same number of incoming edges. The edge $(\mathtt{n} \to \mathtt{n})$ is irrelevant for the structural invertibility behaviour and therefore neglected. As each of these $2(N-1)$ edges exists with a probability of p, the expectation value for the degree of $\mathtt{n}$ turns out to be

$$\mathbb{E}\,\mathtt{d}(\mathtt{n}) = 2(N-1)p\,. \tag{4.3}$$

As this equation holds for each node, we find the same result for the average degree,

$$\mathbb{E}\overline{\mathtt{d}}(\mathtt{g}) = 2(N-1)p\,, \tag{4.4}$$

and also

$$\frac{1}{2}\mathbb{E}\,\mathtt{d}(\mathtt{n}) = \mathbb{E}\,\mathtt{d}_{\mathtt{in}}(\mathtt{n}) = \mathbb{E}\,\mathtt{d}_{\mathtt{out}}(\mathtt{n})\,. \tag{4.5}$$

The algorithm above takes only two parameters, the node number $N \in \mathbb{N}_0$ and an edge probability $0 \le p \le 1$, where the cases $N = 0$, $p = 0$ or $p = 1$ are allowed but trivial. From the latter equations one will see that we could use the node number N and the average degree $0 \le \bar{d} \le 2(N-1)$ as parameters of the algorithm. A graph that is constructed by this algorithm will be homogeneous, meaning there is no special structure in this network. ER-networks are therefore well suited to understand the impact of the *size* of the network, via the node number N, and the impact of the *density* or *connectivity*, via the average degree $\bar{d}$ or the edge probability p. To simulate ER-networks, the function `gnp_random_graph` provided by the `python networkx` package was used.

4.1.2 Scale-Free Networks

The *degree distribution* $\mathtt{P}$ of a network (or graph) $\mathtt{g} = (\mathtt{N},\mathtt{E})$ counts the number of nodes with a certain degree

$$\mathtt{P}(k) := \operatorname{card}\{\mathtt{n} \in \mathtt{N}\,|\,\mathtt{d}(\mathtt{n}) = k\}\,. \tag{4.6}$$

If the degree distribution of a network behaves like a power law

$$\mathtt{P}(k) \approx k^{-\gamma} \tag{4.7}$$

it is called **scale-free** [92]. The following algorithm implements a result presented in [94] and was realised in `python` using the `networkx` package.

Algorithm (Scale-Free network)

> Let $0 < \alpha < 1$ and let $\mathtt{N} = \{1, \ldots, N\}$ be a set of nodes. Let furthermore $0 < \overline{d} < 2(N-1)$ be given. Initialise the graph $\mathtt{g} = (\mathtt{N}, \mathtt{E} = \emptyset)$. Assign a weight of
>
> $$w_{\mathtt{n}} := \mathtt{n}^{-\alpha} \tag{4.8}$$
>
> to each node $\mathtt{n} \in \mathtt{N}$, where $\mathtt{n}$ is understood as numerical value and a probability
>
> $$p(\mathtt{n}) := \frac{w_{\mathtt{n}}}{\sum_{\mathtt{n}\in\mathtt{N}} w_{\mathtt{n}}}\,. \tag{4.9}$$
>
> According to this probability distribution, we draw pairs of nodes $\mathtt{i}$ and $\mathtt{j}$, $\mathtt{i} \neq \mathtt{j}$, and add an edge $(\mathtt{i} \to \mathtt{j})$ to $\mathtt{E}$ as long as the average degree of the graph does not exceed the desired value,
>
> $$\overline{\mathtt{d}}(\mathtt{g}) \le \bar{d}\,. \tag{4.10}$$

For N large enough one will approximate the desired average degree

$$\overline{\mathtt{d}}(\mathtt{g}) \approx \bar{d}. \tag{4.11}$$

The network generated by the algorithm above fulfils a power law with exponent

$$\gamma = \frac{1+\alpha}{\alpha}. \tag{4.12}$$

Due to the obtained degree distribution $\mathtt{P}(k) \sim k^{-\gamma}$, there are many nodes with a small node degree and very few nodes with a large node degree. The former are called **satellites** while the latter ones will be referred to as **hubs** of the network. While the node number and average degree allow to compare the size and density of ER- and SF-networks, the exponent γ acts like a parameter for inhomogeneity. SF-networks are thus useful to understand the consequences of input or output nodes being hubs or satellites of the network.

4.1.3 Structural Invertibility Algorithm

The great advantage of structural invertibility is that one can make use of efficient algorithms developed by graph theorists to solve the invertibility question for dynamic systems. The direct search for a family of node-disjoint paths in a structural system is not efficient yet. Indeed, it is mainly a combinatoric problem and we cannot give a deterministic algorithm with polynomial runtime. But structural invertibility does not require to explicitly find a family of node-disjoint paths, the mere existence of such a family is sufficient. Luckily, here we face a problem where it is indeed possible to formulate another graph theoretical problem which is equivalent to the existence proof: Due to the Max-Flow-Min-Cut-Theorem [88, 52, 95] the existence of a family of node-disjoint paths can be reformulated as a flow problem.

Flow Graph Let $(\mathtt{S}, \mathtt{g}, \mathtt{Z})$ be a structural input-output system with graph $\mathtt{g} = (\mathtt{N}, \mathtt{E})$. We will now construct the **flow graph** $\tilde{\mathtt{g}} = (\tilde{\mathtt{N}}, \tilde{\mathtt{E}})$. First for each node $\mathtt{n} \in \mathtt{N}$ there are two nodes $\mathtt{n}^+, \mathtt{n}^- \in \tilde{\mathtt{N}}$ to separate ingoing and outgoing edges (see Fig. 4.1). Meaning that if $(\mathtt{i} \to \mathtt{j}) \in \mathtt{E}$, then

$$(\mathtt{i}^+ \to \mathtt{j}^-) \in \tilde{\mathtt{E}}. \tag{4.13}$$

In addition to that for each pair $\mathtt{n}^+, \mathtt{n}^- \in \tilde{\mathtt{N}}$ there is an edge

$$(\mathtt{n}^- \to \mathtt{n}^+) \in \tilde{\mathtt{E}}. \tag{4.14}$$

Second, we introduce a **source node** $\sigma \in \tilde{\mathtt{N}}$ with edges

$$(\sigma \to \mathtt{n}^-) \in \tilde{\mathtt{E}} \tag{4.15}$$

for each $\mathtt{n} \in \mathtt{S}$ as well as a **sink node** $\zeta \in \tilde{\mathtt{N}}$ with edges

$$(\mathtt{n}^+ \to \zeta) \in \tilde{\mathtt{E}} \tag{4.16}$$

for each $\mathtt{n} \in \mathtt{Z}$. We assign a weight of 1 to each edge.

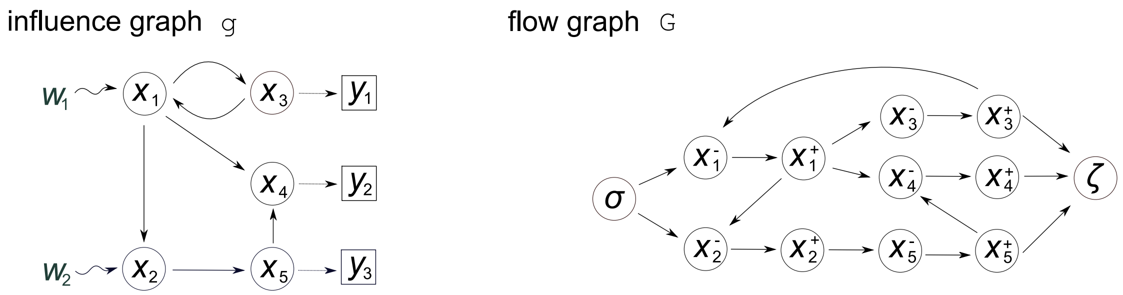

Figure 4.1: The transformation from influence graph $\mathtt{g}$ to flow graph $\mathtt{G}$ for an exemplary system.

Max-Flow Problem It is a fact from graph theory which will not be proven here, that the size of the largest family of node-disjoint paths in $\mathtt{g}$ from $\mathtt{S}$ to $\mathtt{Z}$ equals the maximum flow through $\tilde{g}$ from the source σ to the sink ζ, see, e.g., [95] for an overview and further literature. The Goldberg-Tarjan algorithm [96] is one of several efficient algorithms provided by the `networkx` package to solve maximum flow problems. To analyse the computational complexity of the structural invertibility algorithm, let N and E be the number of nodes and edges in the original graph. The number of input nodes is M and P is the number of output nodes. In directed graphs the number of edges is limited by $E \leq N^2$. For large networks we can assume $M \approx P << N$. To create the flow graph G, $n = 2N + 2$ nodes and $e = N + E + M + P$ edges are created. On $(\sigma, \tilde{\mathtt{g}}, \zeta)$ the Goldberg-Tarjan algorithm has a running time scaling like $\mathcal{O}(n^2\sqrt{e})$. All together we find that the structurally invertibility algorithm has a running time of $\mathcal{O}(N^3)$. On a single node of an `Intel® Xeon® Processor E5-2690 v2` an average time of 0.15 ± 0.03 seconds is needed for a network with $N = 10^3$ nodes and 2.4 ± 0.4 seconds for $N = 10^4$ nodes to check the existence of a node-disjoint path family.

4.2 Scenario I

The first scenario is motivated by pure *a posteriori* analysis. If we do an experiment, we usually have some theoretical idea about what will happen, the nominal model. We consider ER- and SF-networks as network topologies for the nominal model. As we intentionally do not want to respect the system's network topology, the input and output sets are chosen randomly without any preference. As the concrete realisation of the network and the input and output sets are random, we get a probability for structural invertibility, depending on the parameters. E.g., for an ER-network we have a size of N, an edge probability of p, and an input and output set size of M. For each choice of parameters we draw $n = 100$ structural input-output graphs and take

$$\rho(N, p, M) := \frac{\#\,\text{invertible systems}}{\#\,\text{non-invertible systems}} \tag{4.17}$$

as the empirical probability that a system with these parameters is structurally invertible. Please note that the number of invertible systems is binomially distributed, hence each value of ρ has an empirical standard deviation of

$$\sigma(\rho) = \sqrt{n\rho(1-\rho)}\,. \tag{4.18}$$

For the sake of a clearer visualisation we drop the error bars in the following figures, keeping this in mind. We have found that the probability ρ converges as the network gets larger,

$$N \to \infty. \tag{4.19}$$

We fix the node number to $N = 10^3$ as such networks seem to not differ significantly from this limit and at the same are computationally feasible. For simplicity, we consider the same number of input and output nodes. We do this, as the case $\operatorname{card} \mathrm{S} > \operatorname{card} \mathrm{Z}$ will always lead to non-invertibility. More precisely, considering the ratio

$$r := \frac{\operatorname{card} \mathrm{Z}}{\operatorname{card} \mathrm{S}}, \tag{4.20}$$

we know that $r < 1$ means non-invertibility. Increasing this ratio with all the other parameters fixed will monotonically increase the probability ρ but will not lead to any qualitatively new behaviour.

4.2.1 Random Networks

Our first point of interest is the dependence on the density of the network. In a very dense network, there are many interactions between all state variables. Therefore, a perturbative influence on one state variable perculates quickly through the network.

Figure 4.2 shows the probability ρ for ER-networks (a) and SF-networks (b) as a function of the average degree $\bar{d}$ and the input and output number M. One will directly see from both subfigures how the probability monotonically increases with the average degree, indicating that dense networks have a higher tendency to be structurally invertible. Furthermore, the probability decreases with increasing input and output size M. It is plausible that adding input nodes will decrease the probability, whereas adding output nodes must increase the probability to reconstruct the inputs.

It is important to note that the ratio $r = 1$ stays fixed in our simulations. We must deduce that the negative impact of additional input nodes outdoes the positive effect of additional outputs. Here we find an early indication that the number of input nodes has a strong impact on the whole invertibility behaviour. In a later chapter, we will see that we can take advantage of this finding as we formulate a regularisation based on the *sparsity* of the input set.

Looking at subfigure (a) one will realise that for a fixed number M of input and output nodes, the probability ρ as a function of the average node degree undergoes a kind of transition. Starting in a region of non-invertibility, $\rho \approx 0$, there comes a relatively narrow transition zone leading to a region of invertibility, $\rho \approx 1$. Thus, as soon as we know that a system has the structure of an ER-network, we can simply compute the average degree of the influence graph. Due to the narrow transition zone, it is very likely that we directly get an almost certain statement ($\rho \approx 0$ or $\rho \approx 1$) whether the system is invertible or not. Structural invertibility of an influence graph is (almost) equivalent to the analytic invertibility of the original system. This result yields a remarkably simple way to decide whether the (in general non-linear) input-output map Φ of this system is one-to-one. Note, that the ER-networks we consider represent systems of $N = 10^3$ coupled differential equations and that it takes less than a second to compute the average degree of such a network.

> In a homogeneous (ER) network the average degree carries significant information about the invertibility of the whole system.

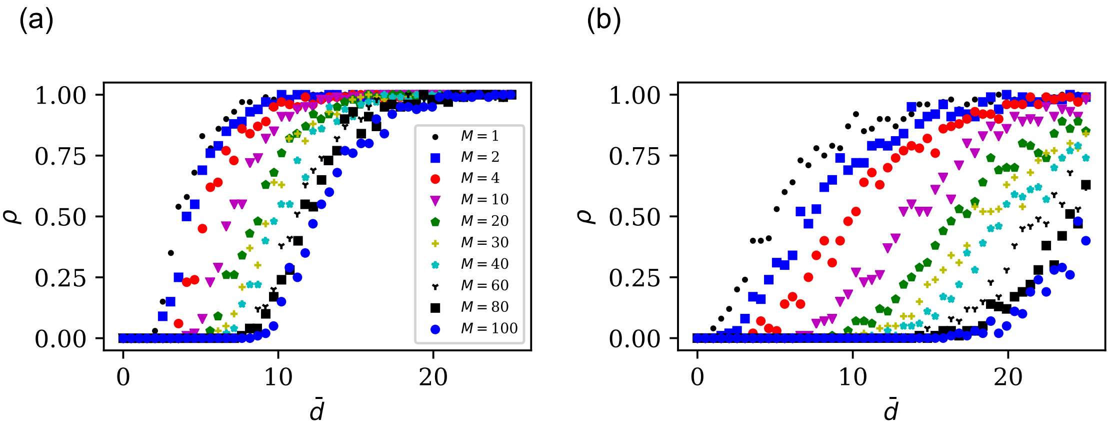

Figure 4.2: The probability ρ for an invertible ER- or SF-network with random input and output sets as a function of the average degree $\bar{d}$. The network size is fixed to $N = 1000$, the input and output sets have a size of M. For each value of $\bar{d}$, ρ is computed as the relative number of structurally invertible networks among 100 network realisations. Each data point has an empirical standard deviation of $\sqrt{N\rho(1-\rho)}$ which is not plotted for better readability. (a) ER-networks. (b) SF-networks with a scaling parameter $\gamma = 2.8$.

Subfigure (b) shows the results for SF-networks with $\gamma = 2.8$. We focus on this choice of γ as it produces networks with SF-behaviour but without *super-hubs* or too many *isolated satellites*. The comparison between ER- and SF-networks (for the same M and $\bar{d}$) leads to the result that inhomogeneity shrinks the probability ρ for the random network to be structurally invertible. Furthermore, the transition from non-invertibility to invertibility is blurred over a wider range. Consequently, the average degree is a less significant indicator for invertibility.

> Dense and homogeneous networks have a better invertibility behaviour than sparse and inhomogeneous networks.

In SF-networks, paths from an input node to an output node are much smaller compared to ER-networks, compare [97] for estimates of network diameters. From the d-algebraic point of view, where a path family from the input nodes to the output nodes can be understood as the propagation of a d-transcendence basis, one might come to the hypothesis that short paths are beneficial. But the network analysis has shown that this is false; instead, metaphorically spoken, the *basis gets stuck* at the hubs and bottlenecks which are already occupied. In dense and homogeneous networks, the basis has a better chance to find a detour around the occupied nodes and consequently a better chance to reach the output nodes.

4.2.2 Real Networks

With the results above in mind, a selection of 17 real networks from various fields was studied under scenario I. Biological systems can handle perturbations remarkably well, think of the immune system that not only detects a virus but somehow *gathers* information about its shape and produces specific anti-bodies, see also the references in the first chapter. Here, the virus can be interpreted as the unknown input and the immune reaction takes precise information about the virus into account. Not in the sense of an intelligent being but in the sense of a smart mechanism, the immune system has to *reconstruct* the form of the unknown input in order to start an appropriate reaction. One might come to the hypothesis that biological systems are *designed*, meaning evolutionarily encoded in the DNA or naturally grown, such that their structure supports the detection of unknown inputs, i.e., that they tend to be structurally invertible.

Some properties of the real networks can be found in table 4.1. Given a real network, we distribute M input and output nodes randomly. For each network and each M this was done $n = 100$ times and again we took the fraction of structurally invertible networks as empirical probability ρ. As explained above, we again drop the error bars in all figures. The resulting probability ρ is plotted in figure 4.6 (a) as a function of the input and output number M. Except for two networks (*WikiVote* and *Consulting*) the probability tends to zero already for small $M < 10$, directly leading to a clear rejection of our primary hypothesis under scenario I.

> Under scenario I, real networks are hardly ever invertible.

This result must be interpreted in a modest way and is not in a too stark contrast to the hypothesis stated above, as soon as we remind ourselves that model errors can be of various nature. One might rather add the hypothesis that invertibility under scenario I is not the mechanism that explains the robustness of biological networks against external perturbations.

4.3 Scenario II

In contrast to the latter scenario which represented an *a posteriori* analysis, we now want to make the first step towards an *experimental design*. More precisely, we want to place the sensors in the network in a way to make the system more likely to be invertible. As before, we have no control about the input nodes, i.e., they will be distributed randomly over the network.

The results of scenario I show, that of two networks with the same size, one ER and one SF, with identical number of input nodes and the same average degree, the ER-networks has a much higher chance to be structurally invertible. Hence, not only the average degree of the network but the degree distribution, i.e., the node degrees the individual nodes have a significant impact on the invertibility.

4.3.1 The Role of Hubs

To understand the role of hubs, random SF-networks were generated as before, but with two selection preferences for the input and output nodes:

A Choose the hubs as output nodes, distribute the input nodes randomly over the remaining nodes.

B Choose the hubs as input nodes, distribute the output nodes randomly over the remaining nodes.

Let us first focus on case A. The network size was again set as $N = 1000$ and the power law exponent to $\gamma = 2.8$. The input and output set size was chosen as $M = 10$. It turned out that choosing the $M = 10$ largest hubs as output set often leads to specific non-invertible structures. To avoid such pathological constellations it makes sense to soften the selection preferences. Instead of choosing the $M = 10$ largest hubs directly as output nodes, the set of the $M' = 50$ largest hubs was taken.

Among these $M' = 50$ nodes, the output set was then chosen randomly. We found $M' = 50$ to be an appropriate number given the parameters of the network, meaning that these $M' = 50$ nodes have an over average node degree. Placing the input nodes randomly over the remaining nodes of the network consequently yields an input set that consists mainly of satellites. Again, it makes sense to not choose the satellites with the lowest node degrees, as these satellites tend to be isolated and therefore again lead to pathological constellations. As before, for each parameter configuration 100 networks were generated and ρ_{HubsOut} represents the fraction of structurally invertible ones. See figure 4.3 for the results. To have a reference, ρ shows the results for randomly chosen input and output sets, thus basically scenario I. From the comparison of ρ_{HubsOut} and ρ one will directly see that the preferential choice of output nodes with a high node degree leads to an increase of the probability for the network to be structurally invertible.

High degree output nodes improve the structural invertibility behaviour.

Before we come to the Sensor Node Placement Algorithm, let us now focus on case B. This case works analogous to the former one. The only difference is that in case B directly choosing the $M = 10$ largest hubs as inputs, i.e., not randomising over the largest $M' = 50$ hubs, won't lead to pathological cases. The resulting probability ρ_{HubsIn} can be found in figure 4.3. One can see that case B increases the probability for structural invertibility again. We will come back to this result in the discussion of scenario III.

4.3.2 Sensor Node Placement Algorithm

The preferential selection of high hubs as output nodes increases the probability that a network is structurally invertible. The advise for any sensor placement is clearly to monitor those state variables which are the hubs of the network. In practice, however, there are often many more restrictions. For example in
cell-biological experiments it is often technically not possible to measure each desired protein concentration [21]. Or the measurement of one specific variable is costly and should be avoided if possible. Many sensor placing algorithms, for instance [53], work constructively. Meaning, they start with a small and insufficient set of sensor and add a sensor that increases the information content of the measurements while keeping the costs small. We have already learned that real networks often have a nearly vanishing chance to be invertible. Starting a constructive sensor node placement therefore comes with the risk of a huge number of iterations and at the end come to the result that one needs to measure almost each state variable. If there are state variables which cannot be observed, the algorithm might even run for a long time without any good sensor placement because there does not even exist a solution that renders the system invertible.

As an alternative approach a destructive algorithm is proposed. As an initial step, we define a maximal output set $\mathtt{Z}_0$, we will call it the *output ground set*, of all state variables or nodes which can principally be observed. Thus, already from the initial step we know whether a solution exists. Then step by step the redundant outputs are removed from the ground set.

Algorithm (Sensor Node Placement)

Let $\mathtt{g} = (\mathtt{N}, \mathtt{E})$ be a given graph, $\mathtt{S} \subseteq \mathtt{N}$ of size M and $\mathtt{Z}_0 \in \mathtt{N}$.

1. If $\mathtt{S}$ is not linked into $\mathtt{Z}_0$, stop. Else proceed.

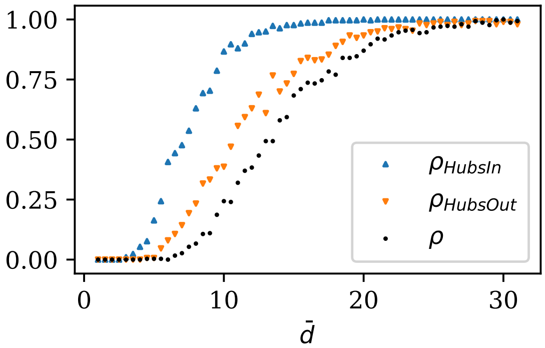

Figure 4.3: The probability that an SF-network is structurally invertibility under preferential input and output placement. For each average node degree $\overline{d}$, ρ is the relative number of structurally invertible networks among 100 realisations of Scale-Free networks with $N = 1000$ nodes and $\gamma = 2.8$, where the input and output sets of size $M = 10$ were drawn uniformly. For ρ_{HubsIn} and ρ_{HubsOut} we took the 50 nodes with the highest node degree and randomly drew a subset of size $M = 10$ as input and output set, respectively. The output and input set, respectively, were drawn randomly from the remaining nodes of the network.

2. Sort Z_0 by ascending node degree.
3. Delete redundant sensors:

```
$i = k = 0$
While $|Z_k| > M$
  If $S$ is linked into $Z_k \backslash Z_k[i]$
    $Z_{k+1} := Z_k \backslash Z_k[i]$
  Else
    $Z_{k+1} := Z_k$
    $i++$
  $k++$
```

Some comments on this algorithm are in turn. First, if step 1 is not successful, then there is no chance to make the system structurally invertible and one has to choose a different ground set Z_0. Second, given that step 1 is successful, this algorithm always results in a set Z^* of size M and such that S is linked into Z^*. This means, the algorithm always produces a minimal output set that renders the system structurally invertible. Third, step two is not mandatory for the algorithm to work, but it will shorten the runtime. We call the algorithm above **satellite-deletion**. We have also considered variations in which the output ground set Z_0 is present with decreasing node degree, called **hub-deletion**, or in random order, **random-deletion**. Figure 4.4 shows the size P of the current output set Z_i depending on the number of iterations. If M is the size of the input set and P_0 the size of the output ground set Z_0, then the number n^* of iterations until the algorithms converge is bounded by

$$P_0 - M \leq n^* \leq P_0 . \tag{4.21}$$

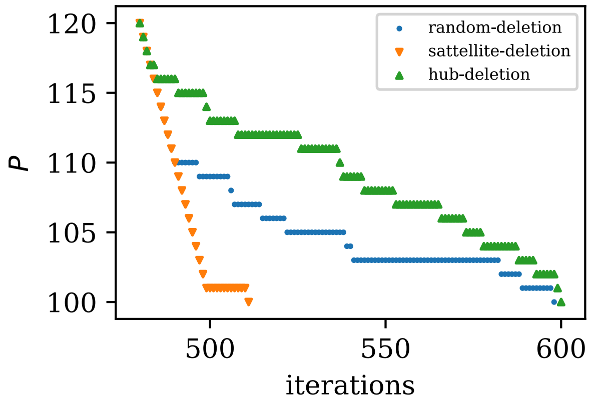

Figure 4.4: Convergence of the Sensor Node Placement Algorithm. The size P of the current output set as a function of the number of iterations for the satellite-deletion, hub-deletion and random-deletion versions of the algorithm for an exemplary ER-system with $N = 1000$ and $M = 100$ inputs. In this example, the minimal number of iterations is 500 and the maximal number of iterations 600.

One will quickly realise this: If step one was successful, then the output ground set Z_0 of size P_0 contains necessarily a minimal set Z^* of size M. Hence Z_0 contains $P_0 - M$ redundant nodes. In the lucky case, Z_0 is sorted such that it starts with the redundant elements. Then in each iteration we delete one element and converge after $P_0 - M$ steps. In the unlucky case, Z_0 is sorted such that the first M elements are a minimal set Z^*. Then the algorithm first performs M checks whether one of these nodes can be deleted, before it takes another $P_0 - M$ steps to delete the redundant nodes. Figure 4.4 shows that the satellite-deletion converges the fastest and nearly after the minimal number of steps. Hub-deletion and random-deletion need almost the maximal number of steps. The fast convergence of the satellite-deletion underlines again that satellites in the role of output nodes are usually redundant. This is also in agreement with the former result on hubs as efficient output nodes.

Finally, it must be noted that there is a clear dependency on the input set S. In practice, one might have prior knowledge or an educated guess about this input set. If, however, there is no plausible way to define this set of input nodes, one can only come back to the result that taking high degree nodes as outputs increases the probability that the system is invertible.

4.3.3 Application to Real Networks

The application of the Sensor Node Placement Algorithm yields a significant improvement of the invertibility behaviour. For the real networks from table 4.1 we have distributed M input nodes over the network and applied the Sensor Node Placement Algorithm. Doing so for 100 times again yields a probability ρ that the system is structurally invertible. The result can be found in figure 4.5. Subfigure (a) shows that the probability decreases to nearly zero as the number of inputs exceeds $M \approx 30$. It is noteworthy that the Sensor Placement Algorithm always works, given that the initial step is successful. That means, if one realisation turns out to be non-invertible, then there indeed exists not one set of sensors that renders the system structurally invertible. This again emphasises that these real systems have a structure that clearly not supports the structural invertibility under

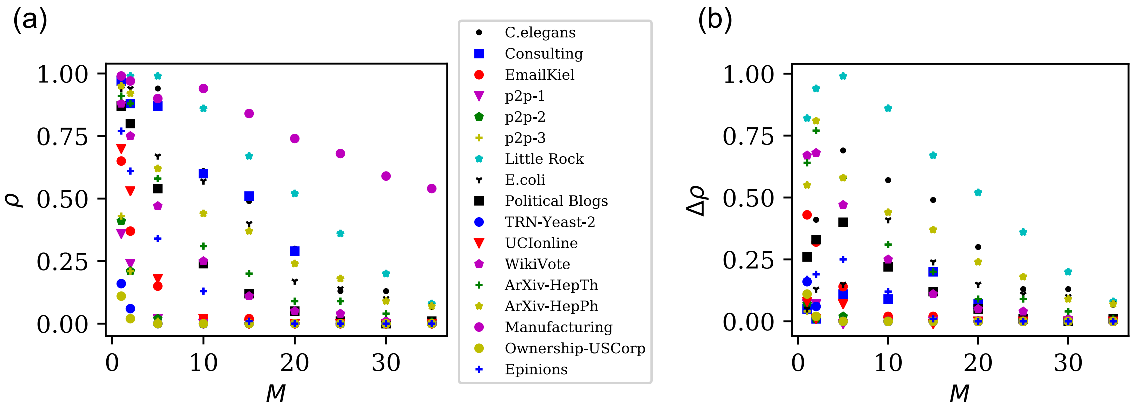

Figure 4.5: Enhanced structural invertibility of real networks. (a) The probability ρ that the networks from table 4.1 are structurally invertible. The input set of size M is drawn randomly, the output set is obtained by the Sensor Node Placement Algorithm. For each M, ρ is the relative number of structurally invertible networks among 100 realisations. Each value is equipped with a standard deviation of $\sqrt{N\rho(\rho-1)}$ which s again not plotted for better readability. (b) The difference in the probabilities between the results of scenario I, see figure 4.6 (a) and scenario II, see subfigure (a).

scenario II. However, subfigure (b) only shows the difference $\Delta\rho$ between scenario I and scenario II, that is, how much the probability increases by a better sensor node placement. Up to a number of $M \approx 30$ input nodes, the probability increase ranges from 0 to 100%, with many networks easily reaching 25 – 50%.

> For up to $M \approx 30$ input nodes, the Sensor Placement Algorithm in able to improve the invertibility behaviour significantly.

4.4 Scenario III

4.4.1 Asymmetric Node Degrees

We have seen that real networks between a few hundreds and a hundred thousand nodes have a vanishing probability to be structuraly invertible under both, the completely random scenario I and even under scenario II with improved sensor placement. It is especially remarkable that even the real networks which show a SF degree distribution have a much worse invertibility behaviour than the SF random networks. To understand the reason for this behaviour, we do the simulations for real networks under scenario I again, but this time we also apply a randomisation to the real networks. If the bad invertibility behaviour is due to very specific network topologies, these special structures should be destroyed. For instance, a network with *feed-forward* structure (discussed at a later point) will not be feed-forward after randomisation.

Algorithm (Degree Preserving Randomisation)

Let $\mathtt{g} = (\mathtt{N}, \mathtt{E})$ be a graph and $0 < p < 1$ a fixed probability.

1. Randomly choose two edges $(\mathtt{a} \to \mathtt{b}), (\mathtt{c} \to \mathtt{d}) \in \mathtt{E}$.

Name	N	|E|	$\bar{d}$	SF	γ	Brief Description	Database
Regulatory							
TRN-Yeast-2 [98]	688	1079	3.14	True	2.29	Transcriptional regulatory network of S.cerevisiae	Uri Alon Lab [99]
Ownership-USCorp [100]	7253	6726	1.85	True	2.45	Ownership network of US corporations	Pajek [14]
Trust							
WikiVote [101]	7115	103689	29.13	False		Who-vote-whom network of Wikipedia users	snap Stanford [16]
Epinions [102]	75888	508837	13.41	True	1.73	Who-trust-whom network of Epinions.com users	snap Stanford [16]
Food Web							
Little Rock [103]	183	2494	27.26	False		Food Web in Little Rock lake	Mount Sinai [104]
Metabolic							
E.coli [105]	1039	5802	11.17	True	2.61	Network of the metabolic reactions of the E. coli bacteria	BiGG [106]
Neuronal							
C.elegans [107]	297	2345	15.79	True	2.15	Neural network of C.elegans	Network Repository [13]
Citation							
ArXiv-HepTh [108]	27770	352807	25.41	False		Citation networks in HEP-TH category of Arxiv	snap Stanford [16]
ArXiv-HepPh [108]	34546	421578	24.41	False		Citation networks in HEP-PH category of Arxiv	snap Stanford [16]
WWW							
Political blogs [109]	1224	19025	31.09	True	1.04	Hyperlinks between weblogs on US politics	Moreno [110]
Internet							
p2p-1 [111]	10876	39994	7.36	False		Gnutella peer-to-peer file sharing network	snap Stanford [16]
p2p-2 [111]	8846	31839	7.2	False		Same as above (at different time)	snap Stanford [16]
p2p-3 [111]	8717	31525	7.2	False		Same as above (at different time)	snap Stanford [16]
Social Communication							
UCIonline [112]	1899	20296	21.38	True	1.33	Online message network of students at UC, Irvine	Opsahl [113]
EmailKiel [114]	57194	103731	3.63	True	1.77	Email network of traffic data collected at University of Kiel, Germany	Barabasi [77]
Intraorganizational							
Manufacturing [115]	77	2228	3.14	False		Social network from a manufacturing company	Opsahl [113]
Consulting [115]	46	879	38.22	False		Social network from a consulting company	Opsahl [113]

Table 4.1: A compilation of networks from various fields, also examined by other authors ([77]). Here N is the number of nodes and $|\mathtt{E}|$ the number of edges. The column *SF* indicates whether the degree distribution shows a power law and if so, the power law exponent γ was computed.

2. With probability p replace the edges with $(\mathtt{a} \to \mathtt{d})$ and $(\mathtt{c} \to \mathtt{b})$.

We choose $p \approx 0.5$ and perform around N (number of nodes in the network) iterations.

The Degree Preserving Randomisation leaves the in- and out-degrees of each node invariant, it just *rewires* the network. Consequently, any special structures might be destroyed, but any overall asymmetry in the degree distribution will be conserved. Figure 4.6 (b) shows the probabilities ρ that were already shown in subfigure (a) against the probabilities ρ_{rand} of the randomised versions of the networks. Except for very few data points one finds

$$\rho \approx \rho_{\text{rand}} . \tag{4.22}$$

The still veiled issue which causes the bad invertibility behaviour of real networks does not vanish under the Degree Preserving Randomisation. Thus, it must lie within the distributions of the in- and out-degrees.

It makes sense to take a closer look into the degree asymmetry of a real network. Let us consider the *E.coli* metabolism network for instance. The *E.coli* is a real and experimentally observed network with a degree distribution that obeys a power law. In figure 4.7 (a) one can see the probability for structural invertibility of this network and of a SF-network of the same parameters. We again see the already known behaviour that the random network has a higher probability to be structurally invertible. Figures 4.7 (b) and (c) show the in- and out-degrees of each node in the real and in the random network, respectively. One can see that both networks have very few hubs (in the upper right corner of the figures) and the node degrees are comparable in magnitude. The obvious difference is that in the real network, subfigure (b), the in- and out-degree of a hub can differ significantly. In contrast to that, the hubs of the random network, subfigure (c), are

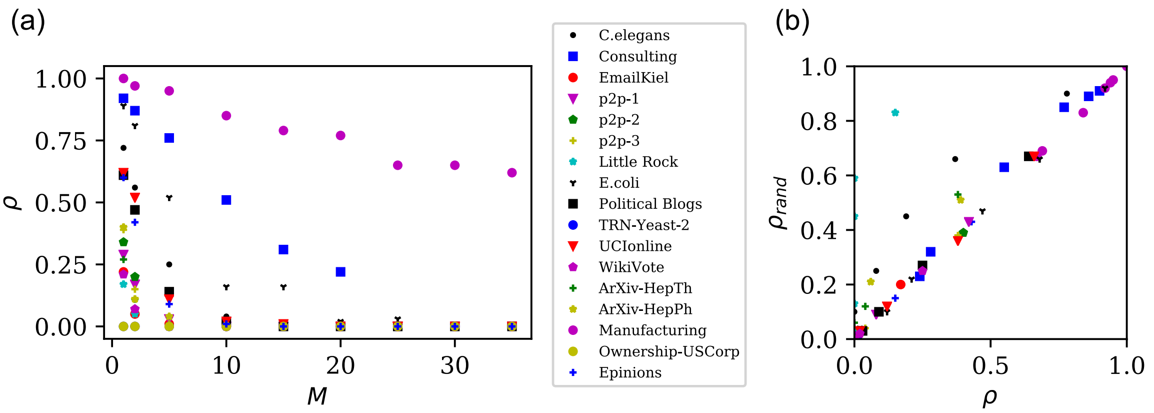

Figure 4.6: Structural invertibility of real networks. (a) The probability ρ that the networks from table 4.1 with randomly placed inputs and outputs is structurally invertible as a function of the size M of the input and output sets. For each value of M one hundred realisations of input and output sets were simulated and ρ gives the fraction of structurally invertible networks. Each data point has an empirical standard deviation of $\sqrt{N\rho(\rho-1)}$ which is not plotted for better readability. (b) The probability from (a) is plotted against the probability computed analogously after a Degree Preserving Randomisation.

symmetrically positioned. It makes sense to introduce the notions of *in-hubs* and *out-hubs*, the former of which lie below the symmetry line while the latter lie above this line.

4.4.2 Full Experimental Design

Out-hubs have the property that they are the starting point of many paths that reach almost all other nodes of the network, whereas in-hubs have the property that many paths terminate there. It clearly supports the existence of a node-disjoint linking between two sets S and Z, if the input nodes S lie on the out-hubs and if the output nodes Z lie on the in-hubs. While in scenario II we have only chosen the output nodes on the hubs, we now want to place the output as well as the input nodes. Under scenario III, we place M inputs on the largest out-hubs and M outputs on the largest in-hubs. This process contains no stochastic elements, consequently, instead of a probability ρ we get a maximum M^* for each network. Any number of in- and outputs $M \leq M^*$ is surely structurally invertible, whereas $M > M^*$ unavoidably leads to non-invertibility. The precise numbers computed for the real networks can be found in figure 4.8. We indeed find that the maximum M^* lies easily two orders of magnitude above the results for scenarios I and II, namely, it is now possible two reconstruct up to hundreds and thousands of inputs.

To improve the invertibility of a system, use in-hubs as outputs and out-hubs as inputs.

4.5 Principles of Structural Invertibility

Figure 4.8 shows a comparison of the probability for structural invertibility for the three scenarios. Starting with a pure *a posteriori* analysis, scenario I is the case in which the input and output nodes are randomly distributed over the system, meaning, not taking any invertibility conditions into account. We have found, that homogeneity and density are the main supporters for structural

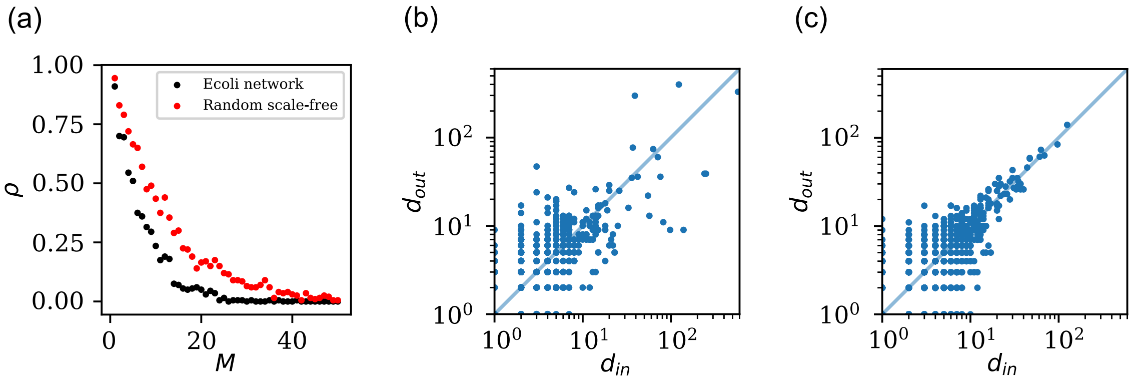

Figure 4.7: Impact of asymmetry. (a) The probability for structural invertibility for the *E.coli* network and a randomly generated SF-network with the same properties. Analogous to scenario I, see figure 4.6, for each M, ρ is the relative number of structurally invertible systems among 100 realisations. (b) The in- and out-degrees of the nodes of the *E.coli* network. (c) The in- and out-degrees of the nodes of a randomly generated Scale-Free network.

invertibility. The existence of hubs and satellites diminishes the probability that a system is invertible. Also, real networks have a strong tendency to be non-invertible *a posteriori*. Except for the *Manufacturing* network which shows an exceptionally good invertibility behaviour and the *E.coli* and *Consulting* networks with a moderate invertibility behaviour, one will easily see that the probability tends to zero already for an input and output set size of $M \lessapprox 3$.

We proceeded with scenario II, where the experimenter is aware of possible model errors and wants to find a sensor placement that increases the chance to infer these model errors from data he or she will collect. It turned out that the hubs of a network can indeed be utilised to improve the invertibility behaviour if chosen preferentially as the outputs of the system. The Sensor Node Placement Algorithm increases the number of inputs that can be observed to $M \lessapprox 10$. However, some networks, e.g., *TRN-Yeast-2*, *Ownership-USCorp* and *p2p-1*, *-2* and *-3* do not benefit from the Sensor Placement Algorithm and remain non-invertible even for small numbers of inputs.

Scenario III finally represents the case, where we have in some sense control over the input nodes. Clearly, inasmuch as model errors are unknown *a priori*, one cannot choose which nodes are targeted by an unknown input. However, in the act of model building one comes to the point to choose which state variables should be included in or excluded from the model. And usually, these are then also the state variables on the *boundaries* of the model, meaning, if they are included, they are likely to be targeted by an exogenous model error, or, if they are excluded, they cause the external model error as they weakly interact with the state variables within the borders of the model. Our investigation has show that one should make the decision based on the in-degree and out-degree of the corresponding nodes. In the sense of invertibility of dynamic input-output systems, we call scenario III a **full experimental design**. As one can see, the full experimental design is indeed able to make, e.g., the *TRN-Yeast-2* network, invertible for input and output sets of size $M \leq 100$, which is remarkable two orders of magnitude higher compared to scenario I and II. Systems with a moderate inhomogeneity due to the existence of hubs benefit from the Sensor Placement Algorithm, but then the additional benefit from the full experimental design is comparably small. For instance, the network *Little Rock* has a nearly vanishing probability

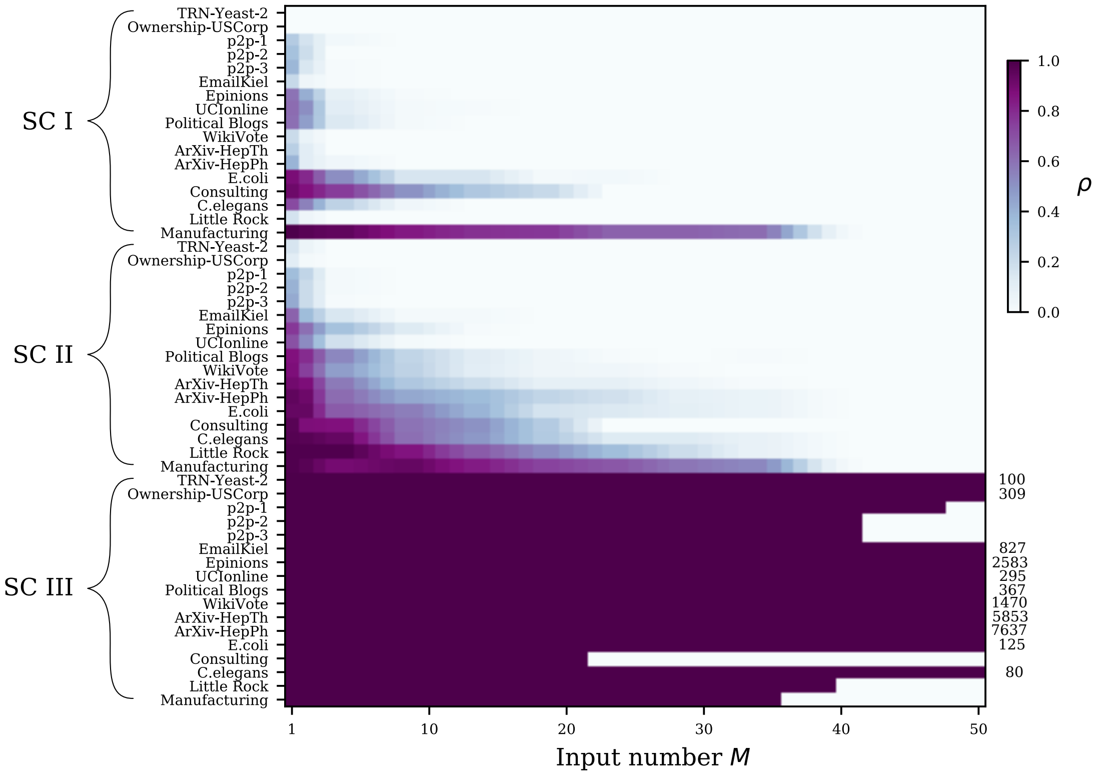

Figure 4.8: A comparison of the probability for structural invertibility of real networks for the three considered scenarios.

already for $M = 1$ under scenario I, but a non-vanishing probability for $M \lessapprox 25$ under scenario II. For scenario III we find a non-vanishing probability for $M \lessapprox 40$, i.e., in the same order of magnitude as in scenario II. Systems with a strong inhomogeneity due to an asymmetry in the degree distributions, for instance *TRN-Yeast-2*, hardly benefit from scenario II at all but show a completely different behaviour under scenario III.

To summarise, we have found several properties, the density, homogeneity, asymmetry and clearly the number of input and output nodes which affect the structural invertibility behaviour of networks. The density of the network and the sizes of the input and output sets lead to a very clear result. The denser the network, the more node-disjoint paths can exists. And the smaller the input set, the smaller is the number of node-disjoint paths we need for invertibility. It should be kept in mind that the number M of input nodes and the number P of output nodes were assumed to be equal, $M = P$. The results for the case of few inputs and many outputs will be qualitatively comparable but quantitatively less frustrating. It is remarkable, though, how homogeneity and asymmetry have a strong inhibitive effect under scenario I, but under scenario II and III we can use them to even improve the invertibility behaviour of a system. As this impact can be both, positive or negative, the network principle cannot solely lie in this property, but one must always consider homogeneity and asymmetry in combination with the input and output placement.

Apart from the concrete results about the network topology and placement of input and output nodes, at this point we actually find that not the influence graph $\mathtt{g} = (\mathtt{N}, \mathtt{E})$ alone is responsible for the structural invertibility of a system, but also the input set $\mathtt{S}$ and output set $\mathtt{Z}$. The triplet

$$(\mathtt{S}, \mathtt{g}, \mathtt{Z}) \tag{4.23}$$

must always be considered together. The conglomeration of all these triplets form a *gammoid* and will be in the focus of the next chapter where they turn out to be an essential tool for the introduction of a *dynamic compressed sensing* of model errors.

CHAPTER 5

Sparse Error Reconstruction

The preceding chapter has lead to the pessimistic result that, more often than not, real systems are not invertible and it is therefore questionable whether we can trust the results of a model error reconstruction in practice. Despite this issue, let this chapter begin with an exemplary error reconstruction.

5.1 Error Reconstruction Example

Nominal Model Look at figure 5.1. Subfigure (a) shows the influence graph of an exemplary $N = 30$ node system. For simplicity and to avoid numerical issues, the system is linear and each edge has weight of one. The initial state is $\boldsymbol{x}_0 = 0$. The equation system for this example can be found in chapter 7, where another approach to the same system is discussed. This model serves as nominal model. A linear system with initial value $\boldsymbol{x}_0 = 0$ should stay zero for all times, so the output if the nominal model behaves trivially

$$\boldsymbol{y}^{\text{nominal}}(t) = \Phi(0)(t) = 0 \quad \forall t \in [0, T]. \tag{5.1}$$

Data To generate pseudo-experimental data for this system, we put an additional input w^* to the state variable x_6 and took ten outputs $y_1, \ldots, y_{10}$ as indicated in the figure. The time course of the simulated output signals can also be found in subfigure (b) as y_i^{data}. A Gaussian noise with a relative standard derivation of 5% of the current output magnitude was added to simulate measurement noise.

Recovery Algorithm We now try to recover the input w^* without any prior knowledge, only using the nominal model and the data. Without prior knowledge, each node could potentially be an input node. That is, we consider the input set

$$\mathrm{S} = \{1, \ldots, 30\}. \tag{5.2}$$

An input set of size $M = 30$ and an output set of size $P = 10$ unavoidably leads to non-invertibility. Ignoring this fact, we augment the nominal model with an input vector $\boldsymbol{w} = (w_1, \ldots, w_{30})^T$ and solve the following optimisation problem to compute an estimate $\hat{\boldsymbol{w}}$ that hopefully approximates the true input w^*.

Algorithm (Recovery Algorithm Δ)

> Let $\Phi : \mathcal{U} \to \mathcal{Y}$ be an input-output map and $\boldsymbol{y}^{\text{data}} \in \mathcal{Y}$ given data over a time domain $\mathcal{T}$. Let M be the number of inputs and P the number of outputs. The **cost functional** $J : \mathcal{U} \to \mathbb{R}$ is defined as
>
> $$J[\boldsymbol{u}] := \sum_{i=1}^{P} \int_{\mathcal{T}} \left| y_i^{\text{data}}(t) - \Phi_i(\boldsymbol{u})(t) \right|^2 \mathrm{d}\,t + \sum_{j=1}^{M} \left(\int_{\mathcal{T}} \left| u_j(t) \right|^3 \mathrm{d}\,t \right)^{1/3} . \tag{5.3}$$
>
> The input is recovered via
>
> $$\Delta : \boldsymbol{y}^{\text{data}} \mapsto \hat{\boldsymbol{u}} \text{ such that } J[\hat{\boldsymbol{u}}] \leq J[\boldsymbol{u}] . \tag{5.4}$$

The recovery map Δ acts as the inversion of the input-output map Φ of the augmented nominal model but is of rather heuristic nature at this point. It fits the output of the augmented model ($\Phi(\boldsymbol{u})$) to the data ($\boldsymbol{y}^{\text{data}}$). The 3-norm on the inputs u_i makes the results smoother. Applying a 2-norm or any other will produce comparably good results.

Results Figure 5.1(b) shows the data and the fit for the output signals. Recovery algorithm Δ fitted the model well to the data. Subfigure (c) shows the estimated input vector. One can see that $\hat{w}_6$ which is the input component that targets state variable x_6 approximates w^* while the other components have a nearly vanishing amplitude.

Conclusion The system is highly non-invertible. More precisely, anticipating a later result, this system has nullity of $P - M = 20$ which vividly spoken means that there is an arbitrariness in the input estimate $\hat{\boldsymbol{w}}$ isomorphic to the function space

$$\underbrace{L^3(\mathcal{T}) \times \ldots \times L^3(\mathcal{T})}_{20 \text{ times}} . \tag{5.5}$$

In other words: We could distribute 20 $L^3(\mathcal{T})$-functions arbitrarily over the thirty components of an input vector $\boldsymbol{u}$ and we are still able to adjust the remaining components such that $\boldsymbol{u}$ reproduces the data. In such a highly under-determined setting it is remarkable that a simple optimisation procedure like Δ is capable of producing an estimate which approximates the ground truth very well.

5.2 Localisability of Model Errors

In the example above we were able to successfully compute an estimate of the model error in a non-invertible system. However, one will realise that the ground truth w^* was very sparse, meaning, of 30 potential input components only one did not vanish.

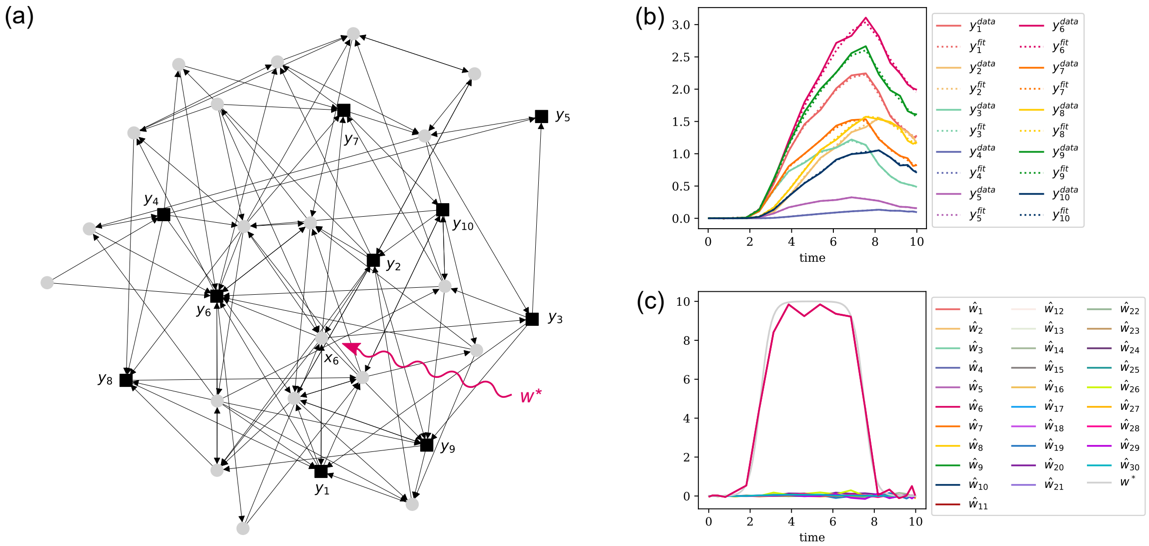

Figure 5.1: Example for a successful model error reconstruction. (a) The influence graph of an exemplary system of size $N = 30$. Each node represents one state variable. The black squares represent the $P = 10$ state variables that are considered as the observables $y_1, \dots, y_{10}$. The red wiggly arrow indicates that an additional input w^*, representing a model error, affects state variable x_6. (b) Pseudo-experimental and predictions of the augmented model with the input shown in (c) (c) The estimated input vector $\hat{\boldsymbol{w}} = (\hat{w}_1, \dots, \hat{w}_{30})^T$ is shown together with the ground truth w^*.

Example (Malfunction)

> Let us discuss another scenario. An electronic circuit works properly for some time. At time t_1 it starts to behave differently. We deduce that some component of the circuit broke at t_1 causing this malfunction. Unless there is a direct causal connection, we would not expect that several disconnected components of the circuit all broke simultaneously. When we try to localise this error, we would therefore first try to find the one broken component, repair or substitute it, and then switch on the electricity again.
>
> The electronic circuit can be seen as a network of electronic components $\mathtt{N}$ and wires $\mathtt{E}$. One node $\mathtt{n} \in \mathtt{N}$ is the broken component, the source of the model error, or in other words, the input node for the input w^*. As we have no prior knowledge about the location of model error within the system, each electrical component must be seen as *potential* input node. Let us call the set of all possible input nodes the *input ground set* L. The true input set, here $\mathsf{S} = \{\mathtt{n}\}$ however, is only a very small subset $\mathsf{S} \subseteq \mathsf{L}$.

One can think of many other examples in which we have good reason to assume the true input set to be very sparse but where we lack knowledge about the location of the input set. This was also the situation in the introductory example, where we successfully recovered the input set $\mathsf{S} = \{6\}$ from a large input ground set $\mathsf{L} = \{1, \dots, 30\}$.

5.2.1 Input Localisation

Instead of directly tackling the full MERP with a large input ground set L, the problem can be split into two parts: First, localise the correct input set S ⊆ L, and second, check whether the system

$$(\mathrm{S}, \mathrm{g}, \mathrm{Z}) \tag{5.6}$$

is invertible.

Model Error Localisation Problem (MELP) Consider a dynamic system with model error $\boldsymbol{w}$ and input ground set L. Find the minimal input set S ⊆ L with

$$w_i(t) = 0 \quad \forall\, t \in [0, T] \quad \Leftrightarrow \quad i \notin S\,. \tag{5.7}$$

Since the true input set S is usually much smaller that the input ground set L, we have a much better chance for invertivility if we solve the MERP for S instead of L. The remainder of this chapter will present a gammoid theoretical approach to the MELP in dynamic systems.

5.2.2 Concepts of Sparse Sensing

The fact that the true input vector $\boldsymbol{w}^*$ had only one non-vanishing component allows to localise the input set, in the example above $\mathrm{S}^* = \{6\}$. Similar findings are known from the literature about sparse and compressed sensing, see [116, 117] for comprehensive overviews and [118] for a textbook on the subject.

Varieties of Sparsity The definition of sparsity is based on our physical understanding of the considered system [119, 117]. For instance, an $\mathbb{R}^N$ vector $\boldsymbol{x} = (x_1, \dots, x_N)^T$ will be considered *sparse* if many components equal zero. The idea of sparsity is with respect to the canonical basis

$$\boldsymbol{x} = x_1 \boldsymbol{e}_1 + \ldots + x_N \boldsymbol{e}_N \tag{5.8}$$

with $\boldsymbol{e}_i = (0, \dots, 0, 1, 0, \dots, 0)^T$ where the one is in the i^{th} component. As another example in an infinite dimensional space take a periodic function like

$$\sin(t)\,. \tag{5.9}$$

With the monomials as basis

$$\sin(t) = t - \frac{t^3}{3!} + \frac{t^5}{5!} - \ldots\,, \tag{5.10}$$

the sine function is highly non-sparse. But interpreted as a Fourier-series the sine is as sparse as possible for it is a basis vector itself. The sparsity of a function can only be judged with respect to a predefined *dictionary* $D = (d_1, d_2, \ldots)$ where each d_i is a function. Ideally, the dictionary is *complete*, meaning, it spans the whole function space of consideration. A representation or *decomposition* of a function $f : \mathbb{R} \to \mathbb{R}$

$$f(t) = a_1 d_1(t) + a_2 d_2(t) + a_3 d_3(t) + \ldots \tag{5.11}$$

is then considered to be sparse if the sequence $(a_1, a_2, \ldots)$ has only few non-vanishing elements. The sparse decomposition of functions of time is investigated under the name of *compressed sensing of analogue signals* [116] and are often called dynamic compressed sensing. However, the term *dynamic* only reflects that it is compressed sensing for something evolving in time. It should not be confused with the idea of a dynamic system, where we have knowledge about the governing laws which determine the evolution of the system.

Another notion of sparsity for dynamic input-output systems is known as *hands-off control* [120]. Here, one wants to *manually* control the time development of a system through inputs. The idea of hands-off control is to apply control inputs, e.g., only for a short interval $[t_0, t_1]$ such that the system evolves *hands-off*, i.e., without further control at a later point. One could also say, hands-off control searches for an input that is *temporally sparse*.

Coming back to the example of an electric circuit with one broken component, one will realise that the caused model error is not necessarily *hands-off*. It is rather the opposite. One component breaks at some point t_1 in time and clearly stays broken for the rest of time $[t_1, T]$, until at T we stop the system or repair the broken element.

We can neither give sense to sparsity with respect to any dictionary. For many applications of compressed sensing of analogue signals one successfully uses Fourier- or *Wavelet-decompositions* [116]. And this is fair, as one has knowledge about the physics behind it, so that one can rely on dictionaries which have also a theoretical background and have proven appropriate over the years. But in the case of *unknown inputs*, we have no understanding of functions which are natural for this problem and which could thus serve as a dictionary. Instead, in order to detect a broken component, it makes sense to search for an input vector that explains the data with only one or very few non-vanishing components. The idea of sparsity in the MELP could be called a *invariable sparsity*. That is, a single (1-sparse) perturbation that affects node i will not switch over to a different node j at some point in time. The input set S remains invariable over time but without restrictions on the functional form of the non-vanishing inputs.

5.3 Gammoid Interpretation of Structural Invertibility

The vast majority of sparse and compressed sensing literature deals with linear decompositions of vectors or functions, consider the literature given above. Despite this trend, we will see that the structure of a *matroid* is actually enough to give reason to the idea of sparsity. The 1935 seminal work on matroids by Whitney [121] was entitled *On the Abstract Properties of Independence*, for the obvious reason that the fundamentals of matroid theory are of rather abstract and theoretical nature. It is a fortunate turning, though that structural input-output systems give rise to a special class of matroids that have been investigated in the 1960s under the name of *gammoids*. With the interpretation of dynamic input-output systems as gammoids we are in the lucky situation to use the theoretical framework of matroid theory and at the same time efficient algorithms on graphs. See also [122] for a textbook on matroids and applications.

5.3.1 Matroids

Definition 20 (Independence Postulates) *Let $\mathscr{C} := \{C_1, \dots, C_N\}$ be a set with power set $\mathfrak{P}\mathscr{C}$. The* ***independent sets*** *$\mathscr{I} \subseteq \mathfrak{P}\mathscr{C}$ fulfil*

1. *$S \in \mathscr{I}$ then $S' \in \mathscr{I}$ for each $S' \subseteq S$.*
2. *Let $S = \{s_1, \dots, s_n\}$ and $T = \{t_1, \dots, t_n, t_{n+1}\}$ in $\mathscr{I}$. Then there is one element $t_i \in T$ such that $S \cup \{t_i\}$ is in $\mathscr{I}$.*

The system $\mathscr{M} = (\mathscr{C}, \mathscr{I})$ of elements $\mathscr{C}$ and independence structure $\mathscr{I}$ is called a ***matroid****.*

Definition 21 (Rank Postulates) *Let $\mathscr{C} := \{C_1, \dots, C_N\}$ be a set with power set $\mathfrak{P}\mathscr{C}$. A function* $\operatorname{rank} : \mathfrak{P}\mathscr{C} \to \mathbb{N}_0$ *that fulfils*

1. $\operatorname{rank}(\emptyset) = 0$.
2. *For $S \in \mathfrak{P}\mathscr{C}$ and $e \in \mathscr{C} \setminus S$ either*

$$\operatorname{rank}(S \cup \{e\}) = \operatorname{rank}(S) \tag{5.12}$$

or

$$\operatorname{rank}(S \cup \{e\}) = \operatorname{rank}(S) + 1\,. \tag{5.13}$$

3. *For $S \in \mathfrak{P}\mathscr{C}$ and $e_1, e_2 \in \mathscr{C} \setminus S$, if*

$$\operatorname{rank}(S \cup \{e_1\}) = \operatorname{rank}(S \cup \{e_2\}) = \operatorname{rank}(S) \tag{5.14}$$

then

$$\operatorname{rank}(S \cup \{e_1, e_2\}) = \operatorname{rank}(S) \tag{5.15}$$

will be called a ***rank*** *function. A system $\mathscr{M} = (\mathscr{C}, \operatorname{rank})$ of elements $\mathscr{C}$ and rank function is called a* ***matroid****.*

Above, there are indeed two definitions of a matroid, the first one based on the *Independence Postulates* and the second one based on the *Rank Postulates*. It is already due to the inventor Whitney that both postulate systems are equivalent [121]. Hence each independence structure is equipped with a rank function and each proper rank function induces an independence structure according to the following theorem. Henceforth, *independence structure* always refers to a system that fulfils the two Independence Postulates. There is indeed a seldom discussed generalisation of matroids also called independence structure that only fulfils the first Independence Postulate without relevance for the purposes of this thesis. The following theorem is a paraphrase of Whitney's results and given without proof.

Theorem 7 (Whitney) *Let $\mathscr{M} = (\mathscr{C}, \mathscr{I})$ be a matroid. Then*

$$\operatorname{rank} : S \mapsto \max_{T \in (\mathfrak{P}S \cap \mathscr{I})} \operatorname{card} T \tag{5.16}$$

is a rank function.

Let $\mathscr{M} = (\mathscr{C}, \operatorname{rank})$ a matroid. Then

$$S \in \mathscr{I} \overset{Def}{\Leftrightarrow} \operatorname{rank} S = \operatorname{card} S \tag{5.17}$$

defines independent sets.

As a consequence of the equivalence of the Independence Postulates and the Rank Postulates, as soon as we find a system with independence structure, we get a rank function. The complement of the rank is the **nullity** which we set as the number such that the equation

$$\operatorname{rank} S + \operatorname{null} S = \operatorname{card} S \tag{5.18}$$

holds. In linear algebra, the columns of a matrix form a matroid and the equation above is known as the the **rank-nullity theorem**. Then the nullity corresponds to the dimension of the *matrix kernel*. At a later point it will become clear that also in the non-linear d-algebraic setting, the nullity makes a statement about the dimensionality of the solution space of the MERP. At this point, however, we first want to discuss the properties of matroids before we connect these to dynamic systems. It shall raise no confusion that we speak of the equation above as rank-nullity *theorem*, though for an abstractly introduced matroid it holds actually by definition.

The terminology of matroid theory is an assembly of terms from linear algebra, graph theory, combinatorics, and other disciplines. This results in a redundant amount of definitions and terms which also shall not be the reason for confusion. As one instance, independence of a set is equivalent to the statement that its nullity is zero. As another example, a **base** of a matroid $\mathscr{M} = (\mathscr{C}, \mathscr{I})$ is a maximally large independent set $B \subseteq \mathscr{C}$. This is equivalent to

$$\operatorname{rank} B = \operatorname{rank} \mathscr{C} . \tag{5.19}$$

If the matroid is a set of matrix columns, then a matroid base is indeed a vector space base of the vector space spanned by the column vectors. Let us come to a third definition of matroids, again equivalent to the ones above.

Definition 22 (Base Postulates) *Let $\mathscr{C} := \{C_1, \ldots, C_N\}$ be a set with power set $\mathfrak{P}\mathscr{C}$. The **bases** $\mathscr{B} \subseteq \mathfrak{P}\mathscr{C}$ fulfil*

1. *If $B \in \mathscr{B}$, then $B' \notin \mathscr{B}$ for each proper subset $B' \subset B$.*
2. *Let $A, B \in \mathscr{B}$ and $a \in A$. Then there is an $b \in B$ such that $(A \setminus \{a\}) \cup \{b\} \in \mathscr{B}$.*

*The system $\mathscr{M} = (\mathscr{C}, \mathscr{B})$ of elements $\mathscr{C}$ and bases $\mathscr{B}$ is called a **matroid**.*

Let us take a look at two matroids.

Example (A Matrix as Matroid)

Consider a matrix $A = (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N)$ where each $\boldsymbol{a}_i \in \mathbb{R}^M$ is a column vector. Interpreted as a matroid, one would simply take the set of columns

$$\mathscr{C} = \{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N\} \tag{5.20}$$

as ground set. An element $S \in \mathfrak{P}\mathscr{C}$ is then likewise a subset of columns

$$S = \{\boldsymbol{a}_{s_1}, \ldots, \boldsymbol{a}_{s_n}\} \tag{5.21}$$

where $\{s_1, \ldots, s_n\}$ are some distinct indices $1 \le s_i \le N$. One will agree that it is the easiest to directly identify S with the corresponding index set

$$S \sim \{s_1, \ldots, s_n\}. \tag{5.22}$$

Clearly, such a set corresponds to a sub-matrix

$$A_S = (\boldsymbol{a}_{s_1}, \ldots, \boldsymbol{a}_{s_n}) \tag{5.23}$$

of course, only up to permutation of columns.

As matroids are motivated by the column structure of matrices, it is not a surprise that the switch from matrix to matroid is basically swapping parentheses with curly braces.

Finally, the matrix rank

$$\operatorname{rank} A_S := \text{maximal number of linearly independent columns} \tag{5.24}$$

can be interpreted as a function $\mathfrak{P}A \to \mathbb{N}_0$ that fulfils the Rank Postulates.

Example (Algebraic Matroid)

Consider a finitely generated field extension $\mathbb{L}/\mathbb{K}$. Remember that a set $\{l_1, \ldots, l_n\} \subseteq \mathbb{L}$ is said to be algebraically dependent, if there is a polynomial $p(L_1, \ldots, L_n) \in \mathbb{K}[L_1, \ldots, L_n]$ such that

$$p(l_1, \ldots, l_n) = 0. \tag{5.25}$$

In the literature, see [123] for a textbook, algebraic matroids are discussed which build on the fact that the structure introduced by algebraic dependence fulfils the Independence Postulates. Using the terminology of algebraic independence, remember the transcendence degree of a field extension $\mathbb{K}(l_1, \ldots, l_n)/\mathbb{K}$ is defined as

$$\text{tr-d}_{\mathbb{K}(l_1,\ldots,l_n)/\mathbb{K}} := \text{maximal number of algebraically independent elements}. \tag{5.26}$$

Indeed, this transcendence degree serves as the rank of $\{l_1, \ldots, l_n\}$. Finally, also a transcendence basis corresponds to a basis in the matroid sense.

Before we discuss in which sense a dynamic input-output system can be interpreted as a matroid, it shall be emphasised that in an equation like

$$A\boldsymbol{x} = \boldsymbol{y} \tag{5.27}$$

with $\boldsymbol{x} \in \mathbb{R}^N$ and $\boldsymbol{y} \in \mathbb{R}^P$, the matrix $A \in \mathbb{R}^{P \times M}$ combines two structures. On the one hand, the columns of A form an independence structure. On the other hand, A is an operator that maps between the spaces $\mathbb{R}^N$ and $\mathbb{R}^P$. In linear algebra for instance, it seems unintuitive to distinguish between these two sides as one examines the linear dependence of columns to get information about the the image and kernel of A as a map. Principally, though, these two structure need not be connected. A matroid can strand alone as well as an operator. Much emphasise is put on the distinction of these two structures for two reasons. First, to understand that the matroid structure of an input-output map Φ is a game changer for the invertibility question. Even without an explicit form of this map, we can investigate the dependence structure to get information about the invertibility of this map. Second, to understand, that *sparsity* is actually a term associated with matroids.

The Spark We introduce a last property, the *spark* of a matroid. The spark of a matrix, in some literature also known as *girth*, consider for instance [124] and references therein to see that the computation of the girth is an object of current research, has been a key in the seminal papers on compressed sensing [119]. It is therefore plausible that the results in the early papers on compressed sensing, which actually stem from the underlying matroid structure, can be generalised for the matroids arising from dynamic input-output systems.

Definition 23 *Let $\mathscr{M} = (\mathscr{C}, \mathscr{I})$ be a matroid. The spark of $\mathscr{M}$ is the largest integer such that for any set $S \in \mathfrak{P}\mathscr{C}$*

$$\operatorname{card} S < \operatorname{spark} \mathscr{M} \Rightarrow S = 0\,. \tag{5.28}$$

5.3.2 The Gammoid Structure of Dynamic Input-Output Systems

There is a problem of purely graph theoretical origin: For a given directed graph $\mathtt{g} = (\mathtt{N}, \mathtt{E})$, find the sets $\mathtt{S} \subseteq \mathtt{N}$ and $\mathtt{Z} \subseteq \mathtt{N}$ such that $\mathtt{S}$ is linked in $\mathtt{g}$ into $\mathtt{Z}$. Actually, this is exactly the problem of structural invertibility to which we came after a large d-algebra detour. Without knowing about this connection the linked paths problem was discussed in a sequence of papers mainly in the late 1960s. Actually tackling graph theoretical questions, Perfect [125] found the probably first connection between this problem and independence structures. Quickly a new name was found by combining *graph* and *matroid* to the new term *gammoid*, which seemingly appeared for the first time in a paper by Pym [126]. The field was pushed and widely understood in the following years [127, 128, 129, 130]. Also the connection to matroids became already very clear. The following definition of gammoids and the new term *weighted gammoids* below have proven appropriate for the purposes of this work.

Definition 24 *Let $\mathtt{g} = (\mathtt{N}, \mathtt{E})$ be a graph and let $\mathtt{L} \subseteq \mathtt{N}$ and $\mathtt{M} \subseteq \mathtt{N}$ be an* ***input ground set*** *and* ***output ground set****, respectively. We call*

$$\Gamma := (\mathtt{L}, \mathtt{g}, \mathtt{M}) \tag{5.29}$$

a ***gammoid****. If there is a* ***weight function*** *$F : \mathtt{E} \to \mathfrak{R}$, where $\mathfrak{R}$ is a ring, and F is a homomorphism in the sense that*

$$F(\pi \circ \rho) = F(\pi) F(\rho) \tag{5.30}$$

and for a set Π of paths

$$F(\Pi) = \sum_{\pi \in \Pi} F(\pi) \tag{5.31}$$

then we call $\mathtt{g}$ a ***weighted graph*** *and Γ a* ***weighted gammoid****.*

With input set $\mathtt{S} \subseteq \mathtt{L}$ and output set $\mathtt{Z} \subseteq \mathtt{M}$, a structural input-output system

$$(\mathtt{S}, \mathtt{g}, \mathtt{Z}) \tag{5.32}$$

can be understood as a realisation of the gammoid $(\mathtt{L}, \mathtt{g}, \mathtt{M})$ With this understanding one will see, that for instance a sensor placement algorithm like the one presented in the last chapter can also be seen as a discrete optimisation problem

$$\begin{aligned} &\text{minimise}\, \mathtt{Z} \subseteq \mathtt{M} \\ &\text{subject to}\, (\mathtt{S}\, \text{is linked into}\, \mathtt{Z}) \end{aligned} \tag{5.33}$$

Already in the early works on gammoids, a detail can sometimes make a difference. So is the initial step of the sensor placement algorithm presented in the chapter before always to check whether the input set S of is linked *into* the output ground set M. If not, there exists no sensor placement that renders the corresponding dynamic system invertible. The algorithm iteratively deletes nodes from M until we reach an output set $\mathtt{Z}^*$. This output set is minimal if and only if S is not only linked *into*, but also linked *to* $\mathtt{Z}^*$.

Input and Output Spaces Let us briefly clarify the connection of the input and output sets S and Z and the input and output spaces $\mathcal{U}$ and $\mathcal{Y}$. In the most general and unrestricted case, the differential equation of a dynamic input-output system in state space form reads

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t)) + \boldsymbol{u}(t) \tag{5.34}$$

where each vector has N components. The space of all allowed input functions $\boldsymbol{u}$ is the input space $\mathcal{U}$ which is a compound space

$$\mathcal{U} = \mathcal{U}_1 \oplus \ldots \oplus \mathcal{U}_N, \tag{5.35}$$

where $\mathcal{U}_i$ is the function space for the i^{th} component of $\boldsymbol{u}$. We have already introduced the input ground set $\mathtt{L} \subseteq \{1, \ldots, N\}$ which tells us that only those components u_i are allowed to be non-zero, for which $i \in \mathtt{L}$. For instance $\mathtt{L} = \{1, 3, 4, \ldots\}$ yields a **restricted input space**

$$\mathcal{U}_{\mathtt{L}} := \mathcal{U}_1 \oplus \{0\} \oplus \mathcal{U}_3 \oplus \mathcal{U}_4 \oplus \ldots \tag{5.36}$$

Another way to express such a restriction would be to introduce an appropriate $\mathbb{R}^{N \times L}$ matrix D so that the differential equation becomes

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t)) + D\boldsymbol{u}(t) \tag{5.37}$$

and the restricted input space takes the form

$$\mathcal{U}_{\mathtt{L}} = \bigoplus_{\mathtt{l} \in \mathtt{L}} \mathcal{U}_l. \tag{5.38}$$

The latter formulation suppresses the trivial function spaces. Both formulations are equivalent and might turn out more or less convenient for certain considerations.

With the same argumentation, we could write the output space as

$$\mathscr{Y}_{\mathtt{M}} = \bigoplus_{\mathtt{m} \in \mathtt{M}} \mathscr{Y}_{\mathtt{m}} \tag{5.39}$$

where $\mathtt{M}$ is the output ground set and $\mathscr{Y}_i$ represents the function space of the observable $y_i(t) = x_i(t)$. In practice, however, we usually work with some fixed output space $\mathscr{Y}$ of dimension $P < N$ that corresponds to the given output set $\mathtt{Z}$.

Furthermore, we usually assume that the model errors are not correlated. That means, if we restrict the input space, then only by restrictions to single components as shown above. For instance

$$-b \le u_i \le b \tag{5.40}$$

makes sense and reflects the assumption that the magnitude of the model error represented by the unknown input u_i is bounded by some value $b \in \mathbb{R}_+$. On the other hand, a restriction like

$$p(u_i, u_j) = 0 \tag{5.41}$$

for some (d-)polynomial p would represent the case of two coupled model errors. Since we have no *a priori* knowledge about the model errors, such a case cannot be excluded by a rigorous argumentation, but goes beyond the presented framework. An example for such correlated model errors was presented in [22].

Say the gammoid $\Gamma = (\mathtt{L}, \mathtt{g}, \mathtt{M})$ and an input set $\mathtt{S} \subseteq \mathtt{L}$ and output set $\mathtt{Z} \subseteq \mathtt{M}$ belong to some dynamic system. If $\mathtt{S}$ is linked into $\mathtt{Z}$ then the input-output map

$$\Phi : \mathscr{U}_{\mathtt{S}} \to \mathscr{Y} \tag{5.42}$$

is one-to-one, hence any solution of the MERP is unique.

The Independence Structure of a Dynamic Input-Output System The considerations above make clear that the structurally invertible input-output configurations of a dynamic system form a gammoid. The original argumentation that a gammoid induces a matroid is indirectly given by a combination of much more elaborated theorems, see [125] and references therein. A new and shorter proof is provided in order to circumvent this technical overload and straightforwardly shows that the linked sets in a gammoid fulfil the Independence Postulates.

Theorem 8 *Let* $\Gamma = (\mathtt{L}, \mathtt{g}, \mathtt{M})$ *be a gammoid. An input set* $\mathtt{S} \subseteq \mathtt{L}$ *is called* ***independent*** *in* Γ*, if* $\mathtt{S}$ *is linked into* $\mathtt{M}$*. Let* $\mathscr{I}$ *be the family of independent sets. Then* $\mathscr{M} = (\mathtt{L}, \mathscr{I})$ *is a matroid.*

Proof

Let $\mathtt{S} \in \mathscr{I}$. Thus, there is an output set $\mathtt{Z} \subseteq \mathtt{M}$ and a family of node-disjoint paths Π such that $\mathrm{in}(\Pi) = \mathtt{S}$ and $\mathrm{ter}(\Pi) = \mathtt{Z}$. For any subset $\mathtt{S}' \subseteq \mathtt{S}$ it is clear that there is a Π' and $\mathtt{Z}' \subseteq \mathtt{Z}$ such that $\mathrm{in}(\Pi') = \mathtt{S}'$ and $\mathrm{ter}(\Pi') = \mathtt{Z}'$. The first Independence Postulate is thus fulfilled.

Now let $\mathtt{S} = \{\mathtt{s}_1, \dots, \mathtt{s}_n\}$ and $\mathtt{T} = \{\mathtt{t}_1, \dots, \mathtt{t}_{n+1}\}$ be two independent sets. Set $\mathtt{S}_i := \mathtt{S} \cup \{\mathtt{t}_i\}$. Let us assume, each $\mathtt{S}_i$ is dependent. A **separator** of $\mathtt{S}_i$ and $\mathtt{M}$ is a set $\mathtt{A}$ such that each path that starts in $\mathtt{S}_i$ and terminates in $\mathtt{M}$ contains at least one node $\mathtt{a} \in \mathtt{A}$. The size of a **minimal separator** between $\mathtt{S}_i$ and $\mathtt{M}$ equals the maximal number of node-disjoint paths between these two sets [88].

As $\mathtt{S}$ is independent, there is a family Π of n node-disjoint paths from $\mathtt{S}$ to $\mathtt{M}$. The paths from Π contain a set $\mathtt{A} = \{\mathtt{a}_1, \dots, \mathtt{a}_n\}$ that is a minimal separator between each $\mathtt{S}_i$ and $\mathtt{M}$. To see this, let $\mathtt{A}_i$ be a minimal separator between $\mathtt{S}_i$ and $\mathtt{M}$. We first realise that $\operatorname{card}\mathtt{A}_i = n$. Since by assumption $\mathtt{S}_i$ is dependent, the size of $\mathtt{A}_i$ is smaller than $n+1$. But since $\mathtt{S} \subset \mathtt{S}_i$ has n node-disjoint paths to $\mathtt{M}$, the size of $\mathtt{A}_i$ is at least n. Hence $\operatorname{card}\mathtt{A}_i = n$.

Second, as each path from $\mathtt{S}_i$ to $\mathtt{M}$ passes $\mathtt{A}_i$, so does each path from $\mathtt{S}$. Consequently, any path family from $\mathtt{S}$ to $\mathtt{M}$ does so and if the family has n node-disjoint paths, then it includes n distinct elements of $\mathtt{A}_i$, that is, the whole set. Hence the family Π contains each $\mathtt{A}_i$.

It remains to show that there is one $\mathtt{A}$ such that we can choose $\mathtt{A}_i = \mathtt{A}$ for each i. This can be done by a separator for which we introduce the term **last separator**. The last separator between $\mathtt{S}_i$ and $\mathtt{M}$ is a minimal separator $\mathtt{A}$ such that between $\mathtt{A}$ and $\mathtt{M}$ there is no other separator $\mathtt{A}'$ except for the trivial one $\mathtt{A}' = \mathtt{A}$. If we choose $\mathtt{A}$ the last separator between $\mathtt{S}$ and $\mathtt{M}$, this is necessarily also the last separator between each $\mathtt{S}_i$ and $\mathtt{M}$.

Now that we have shown that there is a common separator $\mathtt{A}$ of size n, it is clear that each path from $\mathtt{s}_i$ or $\mathtt{t}_i$ to $\mathtt{M}$ must pass $\mathtt{A}$ for all i. Thus $\mathtt{A}$ is necessarily also a separator for $\bigcup_i \mathtt{S}_i = \mathtt{S} \cup \mathtt{T}$ and consequently for $\mathtt{T}$. But by construction $\mathtt{T}$ is independent and of size $n+1$, thus any separator between $\mathtt{T}$ and $\mathtt{M}$ must have a size of at least $n+1$. This is a contradiction, therefore the initial assumption that each $\mathtt{S}_i$ is dependent must be false and the second Independence Postulate is fulfilled.

■

A Note on the Nullity The nullity is a quantity that arises from the gammoid structure of an input-output system. In each representation some properties of the system are more obvious. Others appear to be hidden, but then a change in our point of view unveil these intrinsic qualities. Alike many other properties, the nullity is straightforwardly defined for a gammoid, but it can also viewed in from other perspectives.

Proposition 3 *Consider a dynamic input-output system with d-algebraic representation over a d-field* $\mathbb{K}$ *and structural representation* $(\mathtt{S}, \mathtt{g}, \mathtt{Z})$. *Then*

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle \boldsymbol{U}, \boldsymbol{X}\rangle_\sim}{\mathbb{K}\langle \boldsymbol{Y}\rangle} = \operatorname{null}\mathtt{S}\,. \tag{5.43}$$

Proof The validity of this theorem comes directly from the necessity and sufficiency of structural invertibility for d-algebraic invertibility.

The input set $\mathtt{S}$ has a subset $\tilde{\mathtt{S}} \subseteq \mathtt{S}$ of size $\operatorname{rank}\mathtt{S}$, such that $\tilde{\mathtt{S}}$ is independent, or, equivalently, such that $\tilde{\mathtt{S}}$ is linked in $\mathtt{g}$ into $\mathtt{Z}$. Now split

$$\boldsymbol{U} = (\tilde{\boldsymbol{U}}, \hat{\boldsymbol{U}}) \tag{5.44}$$

such that $\tilde{\boldsymbol{U}}$ corresponds to the input set $\tilde{\mathtt{S}}$. By construction $\tilde{\boldsymbol{U}}$ has a size of $\operatorname{rank}\mathtt{S}$ while the complementary set $\hat{\boldsymbol{U}}$ has a size of $\operatorname{null}\mathtt{S}$. The tower formula now yields

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle \tilde{\boldsymbol{U}}, \hat{\boldsymbol{U}}, \boldsymbol{X}\rangle_\sim}{\mathbb{K}\langle \boldsymbol{Y}\rangle} = \text{d-tr-d}\,\frac{\mathbb{K}\langle \tilde{\boldsymbol{U}}, \hat{\boldsymbol{U}}, \boldsymbol{X}\rangle_\sim}{\mathbb{K}\langle \hat{\boldsymbol{U}}, \boldsymbol{Y}\rangle} + \text{d-tr-d}\,\frac{\mathbb{K}\langle \hat{\boldsymbol{U}}, \boldsymbol{Y}\rangle}{\mathbb{K}\langle \boldsymbol{Y}\rangle}\,. \tag{5.45}$$

By lemma 3

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle\tilde{\boldsymbol{U}},\hat{\boldsymbol{U}},\boldsymbol{X}\rangle_{\sim}}{\mathbb{K}\langle\hat{\boldsymbol{U}},\boldsymbol{Y}\rangle}=\text{d-tr-d}\,\frac{\mathbb{K}\langle\tilde{\boldsymbol{U}},\boldsymbol{X}\rangle_{\sim}}{\mathbb{K}\langle\boldsymbol{Y}\rangle}=0 \tag{5.46}$$

since the structural invertibility of $(\tilde{\mathsf{S}},\mathsf{g},\mathsf{Z})$ is sufficient for a vanishing d-transcendence degree. For the second term

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle\hat{\boldsymbol{U}},\boldsymbol{Y}\rangle}{\mathbb{K}\langle\boldsymbol{Y}\rangle}=\text{d-tr-d}\,\frac{\mathbb{K}\langle\hat{\boldsymbol{U}}\rangle}{\mathbb{K}}=\operatorname{card}\hat{\boldsymbol{U}}=\operatorname{null}\mathsf{S}\,. \tag{5.47}$$

■

The proposition above connects the gammoid interpretation with the d-algebraic formulation. It shows that the (almost) equivalence of structural and d-algebraic invertibility is not restricted to the statement *invertible* or *non-invertible*. When we explore the independence structure of a gammoid, we also learn about the d-algebraic dependencies of the input-output system, its d-transcendence bases and so forth. The following proposition conveys the nullity into the state space form.

Proposition 4 *Let $\Phi:\mathscr{U}\to\mathscr{Y}$ be the input-output map of a state space system and* $(\mathsf{S},\mathsf{g},\mathsf{Z})$ *the structural representation of the system. For the input space $\mathscr{U}=\mathscr{U}_1\oplus\ldots\oplus\mathscr{U}_M$ assume*

$$\mathscr{U}_i=\mathscr{V} \tag{5.48}$$

with some appropriate function space (e.g. $\mathscr{V}=L^p([0,T])$ works for many practical problems). Then for the solution space

$$\mathscr{A}=\{\boldsymbol{u}\in\mathscr{U}\,|\,\Phi(\boldsymbol{u})=\boldsymbol{y}\} \tag{5.49}$$

for a $\boldsymbol{y}\in\mathscr{Y}$ we find

$$\mathscr{A}\cong\mathscr{V}^{\times\operatorname{null}\mathsf{S}}\,. \tag{5.50}$$

Proof This proposition comes as a corollary of the one before. From

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle\boldsymbol{U},\boldsymbol{X}\rangle_{\sim}}{\mathbb{K}\langle\boldsymbol{Y}\rangle}=\operatorname{null}\mathsf{S} \tag{5.51}$$

we deduce that there is a set $\boldsymbol{V}$ of size $\operatorname{null}\mathsf{S}=m$ such that

$$\text{d-tr-d}\,\frac{\mathbb{K}\langle\boldsymbol{U},\boldsymbol{X}\rangle_{\sim}}{\mathbb{K}\langle\boldsymbol{Y},\boldsymbol{V}\rangle}=0\,. \tag{5.52}$$

For instance $\boldsymbol{V}\subseteq\boldsymbol{U}$, i.e. $\boldsymbol{U}=(\tilde{\boldsymbol{U}},\boldsymbol{V})$. That means, we can choose m input functions $\boldsymbol{v}=(v_1,\ldots,v_m)\in\mathscr{V}^{\times m}$ as value for $\boldsymbol{V}$. Since the components of $\tilde{\boldsymbol{U}}$ are d-algebraic over $\mathbb{K}\langle\boldsymbol{Y},\boldsymbol{V}\rangle$, with respect to the d-ideal of the system, there is a generating system for each $U_i\in\tilde{\boldsymbol{U}}$. It follows that if $\boldsymbol{u}$ solves

$$\Phi(\boldsymbol{u})=\boldsymbol{y} \tag{5.53}$$

then for any choice of $\boldsymbol{v}$ we find a $\tilde{\boldsymbol{u}}$ such that

$$\Phi(\boldsymbol{u})=\Phi(\tilde{\boldsymbol{u}}+\boldsymbol{v})\,. \tag{5.54}$$

Hence, the solution $\boldsymbol{u}$ is only fixed up to

$$\mathscr{A} = \mathcal{V}^{\times m}. \tag{5.55}$$

■

5.3.3 Uniqueness of the Sparsest Solution

So far we have seen, how dynamic input-output system naturally incorporate a matroid structure. The first connection between the matroid structure and the input-output map Φ is given by the theorem below. This theorem in the matrix version stems from the original work on compressed sensing [119]. The matroid structure of a dynamic system allows to proof its validity also for non-linear input-output maps.

Definition 25 *For a vector $\boldsymbol{u} \in \mathscr{U}$ the **support** of $\boldsymbol{u}$ is the index set*

$$\operatorname{supp} \boldsymbol{u} := \{i \mid u_i \neq 0\}. \tag{5.56}$$

*The size of the support is called **0-norm** denoted*

$$\|\boldsymbol{u}\|_0 := \operatorname{card} \operatorname{supp} \boldsymbol{u}. \tag{5.57}$$

*If $\|\boldsymbol{u}\|_0 \leq k$ for a $k \in \mathbb{N}_0$, the vector $\boldsymbol{u}$ is called **k-sparse**. The set of k-sparse vectors is denoted $\Sigma_k \subseteq \mathscr{U}$. Let $\Lambda \subseteq \{1, \dots, N\}$ be an index set. The vector $\boldsymbol{u}_\Lambda$ is understood as*

$$(\boldsymbol{u}_\Lambda)_i = \begin{cases} u_i & \text{if } i \in \Lambda \\ 0 & \text{if } i \notin \Lambda \end{cases}. \tag{5.58}$$

Theorem 9 *Consider a dynamic input-output system with input-output map $\Phi : \mathscr{U} \to \mathscr{Y}$ and let Γ be the gammoid of the system. Let $\boldsymbol{y} \in \mathscr{Y}$. If an input $\boldsymbol{u} \in \mathscr{U}$ solves*

$$\Phi(\boldsymbol{u}) = \boldsymbol{y} \tag{5.59}$$

and

$$\|\boldsymbol{u}\|_0 < \frac{\operatorname{spark} \Gamma}{2}, \tag{5.60}$$

then for any other solution $\boldsymbol{v} \in \mathscr{U}$ we find

$$\|\boldsymbol{v}\|_0 > \|\boldsymbol{u}\|_0. \tag{5.61}$$

Proof We first slightly reformulate the theorem as follows: There can be at most one solution with 0-norm smaller than $\operatorname{spark}\Gamma/2$.

So assume there are two distinct solutions $\boldsymbol{u} \neq \boldsymbol{v}$ that both have 0-norm smaller than $\operatorname{spark}\Gamma/2$. If $S := \operatorname{supp} \boldsymbol{u}$ and $T := \operatorname{supp} \boldsymbol{v}$ the assumption says $\operatorname{card}(S), \operatorname{card}(T) < \operatorname{spark}\Gamma/2$. Clearly $Q := S \cup T$ has $\operatorname{card} Q < \operatorname{spark} \Gamma$ thus by definition of the spark

$$\Phi : \mathscr{U}_Q \to \mathscr{Y} \tag{5.62}$$

is one-to-one. So if there is a $\boldsymbol{w} \in \mathscr{U}_Q$ that solves

$$\Phi(\boldsymbol{w}) = \boldsymbol{y}, \tag{5.63}$$

this $\boldsymbol{w}$ is unique with respect to $\mathscr{U}_Q$.

By construction $\mathscr{U}_S \subseteq \mathscr{U}_Q$, thus the input $\boldsymbol{u}$ is a solution that lies in $\mathscr{U}_Q$. So we know that $\boldsymbol{w}$ exists and is necessarily equal to $\boldsymbol{u}$. But also $\mathscr{U}_T \subseteq \mathscr{U}_Q$ so the vector $\boldsymbol{w}$ must also equal $\boldsymbol{v}$. We found $\boldsymbol{u} = \boldsymbol{w} = \boldsymbol{v}$ which contradicts the assumption.

■

The latter theorem can be seen as the fundamental of the remainder of this and the following chapter. It connects the intrinsic independence structure of a dynamic input-output system with the uniqueness of sparse solutions. This proofs that sparsity is indeed a regularisation scheme capable of producing unique solutions in a non-linear non-invertible dynamic system.

5.4 Coherence of Input Nodes

Dynamic input-output system are naturally equipped with the independence structure of a gammoid. Say $\Gamma = (\mathtt{L}, \mathtt{g}, \mathtt{M})$ is the gammoid of such a system. We are now in the position to decide, whether, for instance, an input set $\mathrm{S} = \{\mathtt{s}, \mathtt{t}\}$ of size two is independent or not. If this set is independent, the Model Error Reconstruction Problem with the restricted input space

$$\mathscr{U}_{\mathrm{S}} = \mathscr{U}_{\mathtt{s}} \oplus \mathscr{U}_{\mathtt{t}} \tag{5.64}$$

has a unique solution.

Coherence vs. Strict Dependence Take a look at a simple toy problem in $\mathbb{R}^2$

$$\underbrace{\begin{pmatrix} 2.0 & 1.8 & 0 \\ 0 & 0 & 1.5 \end{pmatrix}}_{:=A} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \underbrace{\begin{pmatrix} 0 \\ 3.0 \end{pmatrix}}_{:=\boldsymbol{y}} . \tag{5.65}$$

Clearly, the rank of any 2 × 3 matrix is at most two, hence the system under-determined. A typical compressed sensing problem. But the case is actually worse. When we compute the spark of A, we search for the smallest dependent subset of columns. In this example, the first two columns are dependent (in the matroid sense, equivalent to linearly dependent in the matrix sense). Hence we obtain

$$\operatorname{spark} A = 2 \tag{5.66}$$

which is the worst possible value for the spark. To understand that, theorem 9 tells us that a solution $\boldsymbol{x}$ is the unique sparsest solution if

$$\|\boldsymbol{x}\|_0 < \frac{\operatorname{spark} A}{2} = 1 . \tag{5.67}$$

However, the only vector that fulfils this inequality is $\boldsymbol{x} = 0$ which makes the whole equation trivial, anyway. Despite this finding, the vector $\boldsymbol{x} = (0, 0, 2)^T$ is clearly the most intuitive and sparsest solution of the equation above. But why do we get such a frustrating result for the spark, and are still able to get a sparse solution?

Obviously, if the data vector $\boldsymbol{y}$ was non-zero in the first component, the situation would be not so obvious because the problem lies in the first two columns of A. Already in the early works on compressed sensing, the coherence

$$\mu_{ij} := \frac{\langle \boldsymbol{a}_i, \boldsymbol{a}_j \rangle}{\|\boldsymbol{a}_i\|_2 \|\boldsymbol{a}_j\|_2} \tag{5.68}$$

was introduced [119], where $\boldsymbol{a}_i$ stands for the i^{th} column of A. A coherence measure for the signal recovery problem was discussed even earlier in [62]. In Euclidean space

$$\mu_{ij} = \cos(\phi_{ij}) \tag{5.69}$$

where ϕ_{ij} is the angle between the corresponding vectors. Hence, μ_{ij} serves as a similarity measure between two columns. For the example above we find

$$\mu_{12} = 1 \quad , \quad \mu_{13} = 0 \quad , \quad \mu_{23} = 0\,. \tag{5.70}$$

Now let us add some noise to the problem,

$$\underbrace{\begin{pmatrix} 2.1 & 1.8 & -0.1 \\ -0.2 & 0.1 & 1.5 \end{pmatrix}}_{:=\tilde{A}} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \underbrace{\begin{pmatrix} 0.2 \\ 2.9 \end{pmatrix}}_{:=\tilde{\boldsymbol{y}}}\,. \tag{5.71}$$

Due to the noise, the first two columns are independent, leading to

$$\operatorname{spark} \tilde{A} = 3\,. \tag{5.72}$$

Adding noise can destroy dependence structures and a system might appear less under-determined. However, exact solutions, as for example $\boldsymbol{x} = (-673/683, 926/683, 1169/683)^T$, are all non-sparse and seem odd. In the spirit of compressed sensing we would rather consider $\boldsymbol{x} = (0,0,2)^T$ to be the most plausible *approximate* solution of the noisy problem. For the coherences we get

$$\mu_{12} \approx 0.99 \quad , \quad \mu_{13} \approx 0.16 \quad , \quad \mu_{23} \approx 0.01\,. \tag{5.73}$$

Whereas the strict definition of dependence is highly sensible to noise, the coherence of the matrix columns again indicates that the first two columns are *practically dependent*, and *practically decoupled* from the third column.

A measure for the coherence of columns can help to judge the plausibility of solutions in a compressed sensing problem. The next aim is therefore to find a coherence measure applicable to dynamic input-output systems. For the remainder of this chapter, we have to restrict the systems of consideration to *linear* dynamic input-output systems.

5.4.1 Weighted Gammoids and the Transfer Function

Henceforth, only *linear* dynamic input-output systems are considered. In a linear differential equation

$$\dot{x}_j(t) = \ldots + a_{ji} x_i(t) + \ldots \tag{5.74}$$

one always finds

$$\frac{\partial \dot{x}_j}{\partial x_i} = a_{ji} \tag{5.75}$$

to be a constant. More precisely, a_{ji} is a constant with respect to the variables $(x_1, \ldots, x_N)$. If the system of consideration is non-autonomous, i.e., it is not a system over the field $\mathbb{R}$ but the d-field $\mathbb{R}\langle t\rangle$, then $a_{ji} \in \mathbb{R}\langle t\rangle$ can be a rational function of time. In the graphical representation, the edge $(\mathtt{i} \to \mathtt{j})$ gets the weight

$$F(\mathtt{i} \to \mathtt{j}) = a_{ji} \,. \tag{5.76}$$

Clearly, for a given weighted gammoid, this identification can also be done the other way round so that gammoids with constant weights and linear dynamic input-output systems have a one-to-one correspondence. For gammoids with non-constant weight, this is not the case. To see this, take for instance

$$\dot{x}_i = f_i(\boldsymbol{x}) \tag{5.77}$$

and realise that

$$\begin{pmatrix} F(\mathtt{1} \to \mathtt{i}) \\ \vdots \\ F(\mathtt{N} \to \mathtt{i}) \end{pmatrix} = \begin{pmatrix} \partial f_i / \partial x_1 \\ \vdots \\ \partial f_i / \partial x_N \end{pmatrix} = \nabla f_i \,. \tag{5.78}$$

Thus only if the weights of the incoming edges at each node form a *conservative vector field*, it is possible to find a differential equation that produces these weights.

Linear System in Laplace Space Consider the differential equation over $\mathbb{R}$ (or any constant d-field)

$$\dot{\boldsymbol{x}}(t) = A\boldsymbol{x}(t) + D\boldsymbol{u}(t) \,. \tag{5.79}$$

The **Laplace-transformation** $\mathfrak{L}$ maps a function of time $f : [0, \infty) \to \mathbb{R}$ to a function $\mathfrak{L}(f) = \tilde{f} : \mathbb{C} \to \mathbb{C}$ in the complex plane according to

$$\mathfrak{L}(f)(s) := \int_0^\infty f(t)\, e^{-st}\, \mathrm{d}\, t \tag{5.80}$$

for $s \in \mathbb{C}$. Whether the technique of the Laplace-transform can also be applied to (non-constant) d-fields is an open question and might be part of future research about the gammoid representation of dynamic systems. Henceforth, our considerations are restricted to the field $\mathbb{R}$. We follow the convention and use a tilde to abbreviate the Laplace-transform of a function. Applying the Laplace-transform element-wise to both sides of equation 5.79 yields a **linear dynamic input-output system in Laplace-space**

$$s\tilde{\boldsymbol{x}}(s) = A\tilde{\boldsymbol{x}}(s) + D\tilde{\boldsymbol{u}}(s) \quad \forall s \in \mathbb{C} \,. \tag{5.81}$$

This shift to Laplace-space is a common technique in the literature about the engineering and control of linear systems, see [27] for a textbook. Rewritten as

$$\tilde{\boldsymbol{x}}(s) = (s - A)^{-1} D \tilde{\boldsymbol{u}}(s) \tag{5.82}$$

and together with a linear observation function

$$\tilde{\boldsymbol{y}}(s) = C\tilde{\boldsymbol{x}}(s) \tag{5.83}$$

one obtains the equation

$$\tilde{\boldsymbol{y}}(s) = C(s - A)^{-1} D \tilde{\boldsymbol{u}}(s) . \tag{5.84}$$

The **transfer function**

$$T(s) := C(s - A)^{-1} D \tag{5.85}$$

maps the input $\tilde{\boldsymbol{u}}$ to the output $\tilde{\boldsymbol{y}}$ and is therefore the Laplace-space representation of the input-output map Φ. The proposition below allows to compute the transfer function without the detour over state-space and Laplace-transform directly from the weighted gammoid.

Lemma 4 *Let $A \in \mathbb{R}^{N\times N}$ be a matrix and $g = (\mathtt{N}, \mathtt{E})$ the graph with nodes $\mathtt{N} = \{1, \ldots, N\}$ and edges $(\mathtt{i} \to \mathtt{j}) \in \mathtt{E}$ whenever $A_{ji} \neq 0$. For each edge we define the weight as $F(i \to j) := A_{ji}$. Then*

$$A^k_{ba} = F(\mathscr{P}_k(\mathsf{a}, \mathsf{b})) . \tag{5.86}$$

Proof Let $\mathtt{l}_0, \mathtt{l}_1, \ldots, \mathtt{l}_k \in \mathscr{N}$ be a list of nodes. If the path $\pi := (\mathtt{l}_0 \to \ldots \to \mathtt{l}_k)$ exists, then $\pi \in \mathscr{P}_k(\mathtt{l}_0, \mathtt{l}_k)$ and we can use the homomorphic property of F to get

$$F(\pi) = F(\mathtt{l}_0 \to \mathtt{l}_1) \ldots F(\mathtt{l}_{k-1} \to \mathtt{l}_k) = A_{l_k l_{k-1}} \ldots A_{l_1 l_0} . \tag{5.87}$$

On the other hand, if π does not exist, at least one of the terms $A_{l_i l_{i-1}}$ equals zero and

$$A_{l_k l_{k-1}} \ldots A_{l_1 l_0} = 0 . \tag{5.88}$$

When we compute the powers of A we find

$$A^k_{ba} = \sum_{l_1, \ldots, l_{k-1} = 1}^{N} A_{b l_{k-1}} \ldots A_{l_1 a} \tag{5.89}$$

which sums up all node lists $\mathtt{l}_1, \ldots, \mathtt{l}_{k-1} \in \mathtt{N}$ with $\mathtt{l}_0 = \mathsf{a}$ and $\mathtt{l}_k = \mathsf{b}$ fixed. It is clear that the terms in the sum do not vanish if and only if the path $(\mathsf{a} \to \mathtt{l}_1 \to \ldots \to \mathtt{l}_{k-1} \to \mathsf{b})$ exists. Thus we can replace the sum by

$$A^k_{ba} = \sum_{\pi \in \mathscr{P}_k(\mathsf{a}, \mathsf{b})}^{N} A_{b l_{k-1}} \ldots A_{l_1 a} \tag{5.90}$$

and we have already seen that for an existing path we can replace the right hand side by

$$A^k_{ba} = \sum_{\pi \in \mathscr{P}_k(\mathsf{a}, \mathsf{b})} F(\pi) = F(\mathscr{P}_k(\mathsf{a}, \mathsf{b})) \tag{5.91}$$

where the second equality comes simply from the definition of the weight function for sets of paths.

■

Proposition 5 *Let T be the transfer function of a linear dynamic system with input set* $\mathsf{S} = (\mathsf{s}_1, \dots, \mathsf{s}_M)$ *and and output set* $\mathsf{Z} = (\mathsf{z}_1, \dots, \mathsf{z}_P)$*. Then*

$$T_{ji}(s) = \frac{1}{s} \sum_{\pi \in \mathscr{P}(\mathsf{s}_i, \mathsf{z}_j)} \frac{F(\pi)}{s^{len \pi}}. \tag{5.92}$$

Proof There are two ways of writing the transfer function. For an input set $\mathsf{S} = \{\mathsf{s}_1, \dots, \mathsf{s}_M\}$ and output set $\mathsf{Z} = \{\mathsf{z}_1, \dots, \mathsf{z}_P\}$ we start with the *full* transfer function $(s-A)^{-1}$ and eliminate the columns with indices not in S and rows with indices not in Z. By this we obtain a complex $P \times M$ matrix. Consequently, since T maps $\tilde{\boldsymbol{w}}$ to $\tilde{\boldsymbol{y}}$, we have to eliminate the components of $\tilde{\boldsymbol{u}}$ with indices not S. Alternatively, one refrains from deleting columns of $(s-A)^{-1}$ such that the transfer function is $P \times N$ and one sets the components of $\boldsymbol{u}$ to zero whenever the index is not in S. So far we have worked with the latter form, however it turns out that the first one will be more convenient in this part. We use the Neumann-series to write

$$T_{ji}(s) = \left[(s-A)^{-1}\right]_{\mathsf{z}_j \mathsf{s}_i} = \frac{1}{s}\left[\left(1 - \frac{A}{s}\right)^{-1}\right]_{\mathsf{z}_j \mathsf{s}_i} = \frac{1}{s} \sum_{k=0}^{\infty} \frac{\left[A^k\right]_{\mathsf{z}_j \mathsf{s}_i}}{s^k}. \tag{5.93}$$

where the brackets just indicate that we first take the matrix power and then take the ji-element. With lemma 4 we get

$$T_{ji}(s) = \frac{1}{s} \sum_{k=0}^{\infty} \sum_{\pi \in \mathscr{P}_k(\mathsf{s}_i, \mathsf{z}_j)} \frac{F(\pi)}{s^k}. \tag{5.94}$$

We now see that $k = \operatorname{len} \pi$. Furthermore we can combine the two sums to get

$$T_{ji}(s) = \frac{1}{s} \sum_{\pi \in \mathscr{P}(s_i, z_j)} \frac{F(\pi)}{s^k}. \tag{5.95}$$

■

5.4.2 Concatenation of Gammoids

In the last section, a proposition was proven that allows to compute the transfer function by finding paths in a weighted gammoid. Before we want to apply this theorem, we first want to consider a helpful construction, the *concatenation* of gammoids. The construction or decomposition of electronic systems in Laplace-space are commonly used in engineering problems, see again the textbook [27]. The concatenation of gammoids transfers this idea to make it applicable in a more general context.

Transposed Gammoid First, the idea of a *transpose* shall be introduced for gammoids. For a linear dynamic system $\mathscr{S}$

$$\begin{aligned}\dot{\boldsymbol{x}}(t) &= A\boldsymbol{x}(t) + B\boldsymbol{w}(t)\\ \boldsymbol{x}(0) &= \boldsymbol{x}_0 \\ \boldsymbol{y}(t) &= C\boldsymbol{x}(t)\end{aligned} \tag{5.96}$$

one finds the **dual system** [27] $\mathscr{S}'$ to be

$$\begin{aligned}\dot{\boldsymbol{x}}(t) &= A^T\boldsymbol{x}(t) + C^T\boldsymbol{y}(t)\\ \boldsymbol{x}(0) &= \boldsymbol{x}_0 \\ \boldsymbol{w}(t) &= B^T\boldsymbol{x}(t)\,.\end{aligned} \tag{5.97}$$

To avoid confusion, note that the term *dual* is used with different meanings in the literature. The duality principle for systems must not be confused with the duality principles for optimal control, see [131] for a textbook, or the duality in the sense of matroids [121]. The *duality principle for systems* is a convenient tool as many properties inherit this duality. For instance, it is known that *observability* and *controllability* are dual, meaning, $\mathscr{S}$ is observable/controllable if and only if $\mathscr{S}'$ is controllable/observable and vice versa [27].

The gammoid of the dual system is considered as the *transposed gammoid* Γ'. The terminology *transposed* is preferred to avoid confusion as the the term *dual gammoid* is already occupied and refers to the matroid duality principle investigated in [129]. The transposed gammoid can also be directly introduced as an *involution* $\Gamma \mapsto \Gamma'$ for arbitrary gammoids. Let $\Gamma = (\mathtt{L}, \mathtt{g}, \mathtt{M})$ be a weighted gammoid. The transposed graph $\mathtt{g}' = (\mathtt{N}', \mathtt{E}')$ can be constructed from $\mathtt{g} = (\mathtt{N}, \mathtt{E})$ in the following way. For each node $\mathtt{n} \in \mathtt{N}$ there is a $\mathtt{n}' \in \mathtt{N}'$. For each edge $(\mathtt{a} \to \mathtt{b}) \in \mathtt{E}$ there is an edge $(\mathtt{b}' \to \mathtt{a}') \in \mathtt{E}'$. The **transposed gammoid** is defined as

$$\Gamma' := (\mathtt{M}', \mathtt{g}', \mathtt{L}')\,. \tag{5.98}$$

Due to the very symmetric relation between Γ and its transpose Γ' it is easy to see that any path

$$\pi = (\mathtt{a}_0 \to \mathtt{a}_1 \ldots \to \mathtt{a}_{k-1} \to \mathtt{a}_k) \tag{5.99}$$

through Γ has a one-to one correspondence to a path

$$\pi' = (\mathtt{a}'_k \to \mathtt{a}'_{k-1} \to \ldots \to \mathtt{a}'_1 \to \mathtt{a}'_0) \tag{5.100}$$

through Γ' of the same length and weight.

Concatenation of Gammoids A cascade of gammoids can be constructed by feeding the output of $\Gamma_i = (\mathtt{L}_i, \mathtt{g}_i, \mathtt{L}_i)$ into the input of $\Gamma_{i+1} = (\mathtt{L}_{i+1}, \mathtt{g}_{i+1}, \mathtt{L}_{i+1})$. Clearly, two things must be noticed:

1. The output ground set $\mathtt{M}_i$ and the input ground set $\mathtt{L}_{i+1}$ must be of the same size.

2. It does matter which output $\mathtt{m} \in \mathtt{M}_i$ is fed into which input $\mathtt{l} \in \mathtt{L}_{i+1}$.

These issues can be handled in different ways. One can consider an injective map $\varphi : \mathtt{M}_i \to \mathtt{L}_{i+1}$. The second point makes clear that there are many, more precisely there is a number of $P!$, of such maps $\varphi_1, \dots, \varphi_{P!}$, where P is the size of the ground sets. Each φ_k represents one possible concatenation of Γ_i and Γ_{i+1}. In a graph theoretical formulation, this is can also be seen as a *matching problem* in the bipartite node set $\mathtt{M}_i \dot\cup \mathtt{L}_{i+1}$. The easiest way to handle this is, however, to simply assume that the sets are given with a certain order, $\mathtt{M}_i = \{\mathtt{m}_1, \dots, \mathtt{m}_P\}$ and $\mathtt{L}_{i+1} = \{\mathtt{l}_1, \dots, \mathtt{l}_P\}$ and one would identify

$$m_k \cong l_k \tag{5.101}$$

for $k = 1, \dots, P$. The resulting gammoid is understood as the **concatenation**

$$\Gamma_i \circ \Gamma_{i+1} . \tag{5.102}$$

Proposition 6 *Let* $\Gamma_1 = (\mathtt{L}_1, \mathtt{g}_1, \mathtt{M}_1)$ *and* $\Gamma_2 = (\mathtt{L}_2, \mathtt{g}_2, \mathtt{M}_2)$ *be two gammoids and* $\Gamma_1 \circ \Gamma_2 = (\mathtt{L}, \mathtt{g}, \mathtt{M})$ *the result of the concatenation. For any* $\mathtt{S} \subseteq \mathtt{L}$ *and* $\mathtt{K} \subseteq \mathtt{M}$ *we find:* $\mathtt{S}$ *is linked in* $\mathtt{g}$ *to* $\mathtt{K}$ *if and only if there is a*

$$\mathtt{Z} \subseteq \mathtt{M}_1 \cong \mathtt{L}_2 \tag{5.103}$$

such that $\mathtt{S}$ *is linked in* $\mathtt{g}_1$ *to* $\mathtt{Z}$ *and* $\mathtt{Z}$ *is linked in* $\mathtt{g}_2$ *to* $\mathtt{K}$.

Proof First, note that in the act of concatenation $\mathtt{M}_1$ is identified with $\mathtt{L}_2$. Assume $\mathtt{S} \subseteq \mathtt{L}$ is linked in g to $\mathtt{K} \subseteq \mathtt{M}$. Hence, there is a linking

$$\Pi = \{\pi_1, \dots, \pi_R\} \tag{5.104}$$

from $\mathtt{S}$ to $\mathtt{M}$. Here R is the size of $\mathtt{S}$ and $\mathtt{K}$. For any edge $\mathtt{a} \to \mathtt{b}$ with $\mathtt{a} \in \mathtt{N}_1$ and $\mathtt{b} \in \mathtt{N}_2$ we necessarily find that either

$$\mathtt{a} \in \mathtt{M}_1 \tag{5.105}$$

or

$$\mathtt{b} \in \mathtt{M}_1 . \tag{5.106}$$

Since $\mathtt{S} \subseteq \mathtt{L}_1 \subseteq \mathtt{N}_1$ and $\mathtt{K} \subseteq \mathtt{M}_2 \subseteq \mathtt{N}_2$, each path π_i goes through a one node $\mathtt{z}_i \in \mathtt{M}_1$ and these nodes are pairwise distinct. We find that $\mathtt{Z} := \{\mathtt{z}_1, \dots, \mathtt{z}_R\} \subseteq \mathtt{M}_1$ acts as a separator.

We can decompose the path

$$\pi_i = \pi_i^1 \circ \pi_i^2 \tag{5.107}$$

at $\mathtt{z}_i$ such that π_i^1 starts in $\mathtt{S}$ and terminates at $\mathtt{z}_i$ and π_i^2 starts at $\mathtt{z}_i$ and terminates in $\mathtt{K}$. We find that

$$\Pi^1 = \{\pi_1^1, \dots, \pi_R^1\} \tag{5.108}$$

is a linking such that $\mathtt{S}$ is linked in $\mathtt{g}_1$ to $\mathtt{Z}$. Analogously, through Π^2 we see that $\mathtt{Z}$ is linked in $\mathtt{g}_2$ to $\mathtt{K}$.

Now assume $\mathtt{S}$ is linked in $\mathtt{g}_1$ to $\mathtt{Z}$ and $\mathtt{Z}$ is linked in $\mathtt{g}_2$ to $\mathtt{K}$. Let $\mathtt{Z} = (\mathtt{z}_1, \dots, \mathtt{z}_R)$ where again R is the cardinality of $\mathtt{S}$ and $\mathtt{K}$. That means we find a linking

$$\Pi^1 = \{\pi_1^1, \dots, \pi_R^1\} \tag{5.109}$$

such that π_i^1 starts in $\mathtt{S}$ and terminates at $\mathtt{z}_i$. We also find a linking

$$\Pi^2 = \{\pi_1^2, \dots, \pi_R^2\} \tag{5.110}$$

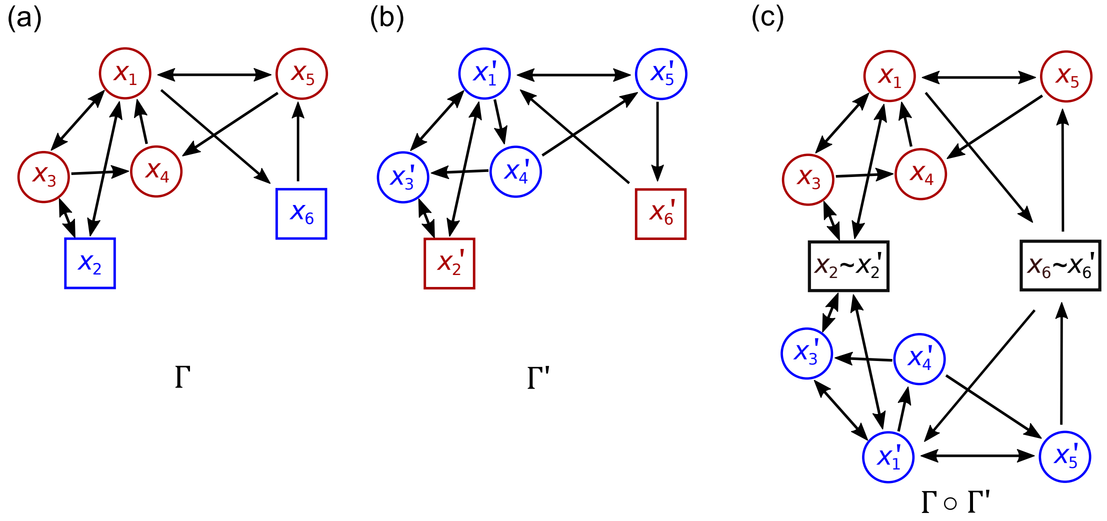

Figure 5.2: Construction of the Gramian gammoid. (a) An exemplary gammoid Γ. The red round nodes are the input set, the blue squared nodes represent the outputs. (b) The transposed gammoid Γ'. The round blue nodes are the output set and the red squares represent the input nodes. (c) The outputs of Γ are identified with the inputs of Γ'. We obtain the Gramian gammoid $\Gamma \circ \Gamma'$ with inputs indicated in red and outputs indicated in blue.

such that π_i^2 starts at $\mathtt{z}_i$ and terminates in $\mathtt{K}$. We can concatenate the paths

$$\pi_i := \pi_i^1 \circ \pi_i^2 . \tag{5.111}$$

Since π_i^1 only contains nodes from $\mathtt{N}_1$ and π_i^2 from $\mathtt{N}_2$, it is clear that the paths from

$$\Pi := \{\pi_1, \dots, \pi_R\} \tag{5.112}$$

are again node-disjoint. Hence $\mathtt{S}$ is linked in $\mathtt{g}$ to $\mathtt{K}$.

■

The proposition above emphasises that, though the input ground set $\mathtt{L}_1$ is the same for Γ_1 and Γ, the independence structure might be different. For instance, an input set $\mathtt{S} \subseteq \mathtt{L}_1$ can be independent in Γ_1 but dependent in Γ; indicating that it is the structure of Γ_2 that renders the system structurally non-invertible.

A special case is the concatenation of a gammoid with its own transpose,

$$\Gamma \circ \Gamma' \tag{5.113}$$

We will call this the **Gramian gammoid** of Γ. In figure 5.2, please find a visualisation of the construction of a Gramian gammoid. The aforementioned issue disappears for a Gramian gammoid: An input set is independent in $\Gamma \circ \Gamma'$ if and only if it is independent in Γ.

5.4.3 Gramian

In the work on compressed sensing with a matrix A, the *Gramian* matrix

$$G := A^T A \tag{5.114}$$

yields a convenient way to compute the coherences of the matrix columns, see for instance the overview [117]. The Gramian introduced below shall not be confused with the *input Gramian* or *output Gramian* from control theory [27]. The latter two are also useful for the controllability and observability of dynamic systems and related but not identical with the Gramian presented here.

Definition 26 *Let $T : \mathbb{C} \to \mathbb{C}^{P\times M}$ be the transfer function of a linear dynamic input-output system with P output nodes and M input nodes. The **Gramian** of the system is defined as*

$$G(s) := T^*(s)T(s) \quad s \in \mathbb{C} \tag{5.115}$$

where the asterisk denotes hermitian conjugate.

Making use of proposition 5 we can compute the Gramian via

$$G(s)_{ba} = \sum_{i=1}^{P} \frac{1}{|s|^2} \sum_{\rho\in\mathscr{P}(\mathsf{s}_a,\mathsf{z}_i)} \frac{F(\rho)}{s^{\operatorname{len}\rho}} \sum_{\pi\in\mathscr{P}(\mathsf{s}_b,\mathsf{z}_i)} \frac{F(\pi)}{\bar{s}^{\operatorname{len}\pi}} . \tag{5.116}$$

The Gramian of the transfer function is closely related to the Gramian gammoid of the system. Indeed, in the equation above one can interpret

$$\psi := \rho \circ \pi' \tag{5.117}$$

as a path in $\Gamma \circ \Gamma'$ of weight

$$F(\psi) = F(\pi)F(\rho) \tag{5.118}$$

and length

$$\operatorname{len}\psi = \operatorname{len}(\rho) + \operatorname{len}(\pi) \tag{5.119}$$

Because any path ψ in $\Gamma\circ\Gamma'$ has always a unique decomposition in such a ρ and π' we can introduce the multi-index notation

$$s^{\psi} := s^{\operatorname{len}\rho}\bar{s}^{\operatorname{len}\pi} . \tag{5.120}$$

The necessity for this notation arises from the fact that the transposed gammoid Γ' represents the transposed transfer function T^T but not the conjugate T^* which actually appears in the Gramian.

An alternative way to circumvent the multi-index can be the following. In the transfer function element T_{ji}, a path

$$\pi = (\mathsf{s}_i \to \mathsf{a}_1 \to \ldots \to \mathsf{a}_{L-1} \to \mathsf{z}_j) \tag{5.121}$$

produces an additive term

$$T_{ji}(s) = \ldots + \frac{F(\pi)}{s^{L+1}} + \ldots \tag{5.122}$$

Each edge $(\mathsf{a}_k \to \mathsf{a}_{k+1})$ contributes a weight $F(\mathsf{a}_k \to \mathsf{a}_{k+1})$ and a factor $1/s$. The additional $1/s$ stems from the imaginary edge from the source of the input to the input node. In this thesis, the input node $\mathtt{s}_i$ represents a state variable that is directly targeted by an input. In the literature about dynamic systems with inputs, it is also common to have a node $\mathtt{u}_i$ that represents the source of the input, see for instance [53]. To build the bridge to the notation used in this thesis, one simply adds an edge

$$\mathtt{u}_i \to \mathtt{s}_i \tag{5.123}$$

of weight one. This elongates π to a length of $L+1$. It is possible to shift the $1/s$ into the weight function. The new **complex weight** Ψ of an edge $(\mathtt{a} \to \mathtt{b})$ is then

$$\Psi(\mathtt{a} \to \mathtt{b}) := \frac{F(\mathtt{a} \to \mathtt{b})}{s}\,. \tag{5.124}$$

Analogous to the transposed gammoid $\Gamma' = (\mathtt{N}', \mathtt{E}')$ we can get a **conjugate** gammoid $\Gamma^* = (\mathtt{N}^*, \mathtt{E}^*)$ if we additionally complex conjugate the weights,

$$\Psi(\mathtt{a} \to \mathtt{b}) = \overline{\Psi\left(\mathtt{b}^* \to \mathtt{a}^*\right)}\,. \tag{5.125}$$

Anyway, we finally achieve a simple expression for the Gramian, either with multi-index

$$G(s)_{ba} = \frac{1}{|s|^2} \sum_{\psi \in \mathscr{P}(\mathtt{s}_a, \mathtt{s}'_b)} \frac{F(\psi)}{s^{\psi}} \tag{5.126}$$

or with the complex weight function

$$G(s)_{ba} = \Psi\left(\mathscr{P}(\mathtt{s}_a, \mathtt{s}^*_b)\right)\,. \tag{5.127}$$

In both notations, the computation of the Gramian $G(s)$ becomes mainly a search for paths, which is the same for $\Gamma \circ \Gamma'$ and $\Gamma \circ \Gamma^*$.

A Lemma and a Proposition Before we turn our interest to the meaning of the Gramian for the invertibility of a system, tools are provided in the form of the following lemma and proposition.

Lemma 5 *Let* $\Gamma = (\mathtt{L}, \mathtt{g}, \mathtt{Z})$ *be a gammoid and* $\mathtt{s}_a, \mathtt{s}_b \in \mathscr{L}$ *input nodes. In* $\Gamma \circ \Gamma'$ *let* $\gamma_{aa'}$ *denote the shortest path from* $\mathtt{s}_a$ *to* $\mathtt{s}'_a$, $\eta_{bb'}$ *denote the shortest path from* $\mathtt{s}_b$ *to* $\mathtt{s}'_b$ *and* $\eta_{ab'}$ *shall denote the shortest path from* $\mathtt{s}_a$ *to* $\mathtt{s}'_b$*. Then*

$$\frac{\operatorname{len}\eta_{aa'} + \operatorname{len}\eta_{bb'}}{2} \leq \operatorname{len}\eta_{ab'} \tag{5.128}$$

Proof By construction there is a $\mathtt{z} \in \mathtt{Z}$ and a decomposition $\eta_{ab'} = \alpha \circ \beta'$ such that α goes from $\mathtt{s}_a$ to node $\mathtt{z}$ and β goes from $\mathtt{s}_b$ to $\mathtt{z}$. We now find that $\alpha \circ \alpha'$ goes from $\mathtt{s}_a$ to $\mathtt{s}'_a$. By assumption of the lemma, $\alpha \circ \alpha'$ is not shorter than $\eta_{aa'}$, thus

$$\operatorname{len}\eta_{aa'} \leq \operatorname{len}(\alpha \circ \alpha') = 2 \operatorname{len}\alpha\,. \tag{5.129}$$

Analogously

$$\operatorname{len}\eta_{bb'} \leq 2 \operatorname{len}\beta\,. \tag{5.130}$$

We sum up these two inequalities to get

$$\operatorname{len}\eta_{aa'} + \operatorname{len}\eta_{bb'} \leq 2(\operatorname{len}\alpha + \operatorname{len}\beta) \tag{5.131}$$

and since $\alpha \circ \beta'$ is the decomposition of $\eta_{ab'}$

$$\operatorname{len}\eta_{aa'} + \operatorname{len}\eta_{bb'} \leq 2\operatorname{len}\eta_{ab'}\,. \tag{5.132}$$

■

In a more provocative formulation the inequality above could also be written as

$$\operatorname{len}\eta_{aa'} + \operatorname{len}\eta_{bb'} \leq \operatorname{len}\eta_{ab'} + \operatorname{len}\eta_{ba'}\,. \tag{5.133}$$

It says that for two nodes a, b and their mirror partners a', b', the cross distances, i.e., from a to b' and from b to a', are never shorter than the straight distances from a to a' and from b to b'. As we have already discovered that the path lengths correspond to the powers of $1/s$ in the Gramian $G(s)$, this lemma can be used in the following proposition about the convergence in the $s \to \infty$ limit.

Proposition 7 *Let G be the Gramian of a linear dynamic input-output system and*

$$\mu_{ab}(s) := \frac{|G_{ab}(s)|}{\sqrt{G_{aa}(s)G_{bb}(s)}}\,. \tag{5.134}$$

Let $\eta_{aa'}, \eta_{bb'}$ and $\eta_{ab'}$ be the shortest paths as defined in lemma 5.

If lemma 5 holds with equality, then

$$\lim_{|s|\to\infty} \mu_{ab}(s) = \frac{|F(\eta_{ab'})|}{\sqrt{F(\eta_{aa'})F(\eta_{bb'})}}\,. \tag{5.135}$$

If the lemma holds with strict inequality, then

$$\lim_{|s|\to\infty} \mu_{ab}(s) = 0\,. \tag{5.136}$$

Proof As a first remark, note that $\mu_{ab}(s)$ is well defined. The transfer matrix has the form

$$T(s) = C(\mathbb{1}s - A)^{-1}D \tag{5.137}$$

where the matrices A, C and D define the system. The transfer function and the Gramian $G = T^*T$ have singularities, more precisely, poles whenever $s \in \mathbb{C}$ is in the spectrum of A. But in μ_{ab} these poles disappear.

Using equation (5.127) we can write

$$\mu_{ab}(s) = \frac{|G_{ab}(s)|}{\sqrt{G_{aa}(s)G_{bb}(s)}} = \frac{\left|\sum_{\psi\in\mathscr{P}(\mathsf{s}_a,\mathsf{s}'_b)} F(\psi)s^{-\psi}\right|}{\sqrt{\sum_{\pi\in\mathscr{P}(\mathsf{s}_a,\mathsf{s}'_a)} F(\pi)s^{-\pi}\sum_{\theta\in\mathscr{P}(\mathsf{s}_b,\mathsf{s}'_b)} F(\theta)s^{-\theta}}} \tag{5.138}$$

First, let us see that the terms under the square-root are non-negative. For that, let $\pi = \alpha \circ \beta'$ be a path from s_a to s'_a. Then we know that also $\beta\circ\alpha'$, $\alpha\circ\alpha'$ and $\beta\circ\beta'$ exist and all go from s_a to s'_a. Thus, in the component G_{aa} we always find the four terms

$$R := \frac{F(\alpha\circ\beta')}{s^{\operatorname{len}\alpha}\bar{s}^{\operatorname{len}\beta}} + \frac{F(\beta\circ\alpha')}{s^{\operatorname{len}\beta}\bar{s}^{\operatorname{len}\alpha}} + \frac{F(\alpha\circ\alpha')}{s^{\operatorname{len}\alpha}\bar{s}^{\operatorname{len}\alpha}} + \frac{F(\beta\circ\beta')}{s^{\operatorname{len}\beta}\bar{s}^{\operatorname{len}\beta}} \tag{5.139}$$

together. So it is sufficient to show that R is non-negative. With $A := F(\alpha)/s^{\operatorname{len}\alpha}$ and $B := F(\beta)/s^{\operatorname{len}\beta}$ we can rewrite R as

$$R = A\bar{B} + B\bar{A} + A\bar{A} + B\bar{B}. \tag{5.140}$$

With $A = x + \mathrm{i}y$ and $B = u + \mathrm{i}v$ we find

$$R = (x+u)^2(y+v)^2 \geq 0\,. \tag{5.141}$$

We now proceed with (5.138). As we want to take the limit $|s| \to \infty$, the smallest powers of s will be dominant. The smallest powers of s correspond to the shortest paths. We neglect higher orders and get the asymptotic behaviour

$$\mu_{ab}(s) \simeq \frac{\left|F(\eta_{ab'})s^{-\eta_{ab'}}\right|}{\sqrt{F(\eta_{aa'})F(\eta_{bb'})s^{-(\eta_{aa'}+\eta_{bb'})}}} \tag{5.142}$$

where we use the sign "$\simeq$" to denote the asymptotic behaviour for $|s| \to \infty$. Since the path $\eta_{aa'}$ is always symmetric in the sense $(\eta_{aa'})' = \eta_{aa'}$ we find

$$s^{-\eta_{aa'}} = |s|^{-\operatorname{len}\eta_{aa'}}\,. \tag{5.143}$$

The same holds for $\eta_{bb'}$. It follows that

$$\mu_{ab}(s) \simeq \frac{|F(\eta_{ab'})|}{\sqrt{F(\eta_{aa'})F(\eta_{bb'})}}|s|^{\frac{1}{2}(\operatorname{len}\eta_{aa'}+\operatorname{len}\eta_{bb'})-\operatorname{len}\eta_{ab'}}\,. \tag{5.144}$$

Lemma 5 tells us that the exponent of $|s|$ is always non-positive, thus the limit always exists. If the lemma holds with equality, then the exponent exactly vanishes and we get

$$\mu_{ab}(s) \simeq \frac{|F(\eta_{ab'})|}{\sqrt{F(\eta_{aa'})F(\eta_{bb'})}}\,. \tag{5.145}$$

If inequality holds, then the exponent of $|s|$ is negative and we find

$$\lim_{|s|\to\infty} \mu_{ab}(s) = 0\,. \tag{5.146}$$

■

We always speak of, e.g., $\eta_{aa'}$ as *the* shortest path. In general, there might be several paths of minimal length. Due to the homomorphic property of F, the calculations above stay valid if $\eta_{aa'}$ is a set of paths.

5.5 Spark Estimation via the Mutual Coherence

In the original works on compressed sensing, the coherence was not just introduces as a measure that relaxes the strict dependence structure of a matrix, but it's value lies in the fact that we can compute an estimate for the spark, see [119] for early ideas and [116] for a broader treatment. As the calculation of the spark of a matrix (and of any matroid) is a combinatorial problem, it becomes quickly infeasible for realistic problems.

5.5.1 Eigenvalues of the Gramian

The *Gershgorin Circle Theorem* has proven a convenient tool in the theory of (static) compressed sensing [116]. Fortunately, it stays valid for the more general case of compound (possibly infinite dimensional) Banach spaces [26] which already indicates that it an also be used for dynamic compressed sensing in function spaces. In contrast to the static case where one has a real-valued Gramian matrix $A^T A$, for linear dynamic systems we get a complex-valued and non-constant Gramian $G : \mathbb{C} \to \mathbb{C}^{M \times M}$, where M is the number of input nodes.

Rank and Normal Rank of the Transfer Function Let $T : \mathbb{C} \to \mathbb{C}^{P \times M}$ be the transfer function of a linear dynamic system and assume for an $s_0 \in \mathbb{C}$ we find the rank

$$\operatorname{rank} T(s_0) = r < M. \tag{5.147}$$

At s_0, the transfer function is not one-to-one. If we visualise the transfer function as a set of column vectors,

$$T(s) = (\boldsymbol{t}_1(s), \dots, \boldsymbol{t}_M(s)), \tag{5.148}$$

the set $\{\boldsymbol{t}_1(s_0), \dots, \boldsymbol{t}_M(s_0)\}$ must be linearly dependent.

This dependence can have two reasons. Either, $\{\boldsymbol{t}_1, \dots, \boldsymbol{t}_M\}$ *as function* are linearly dependent, say of rank r. Then the rank of $T(s)$ will never exceed r. Or, $\{\boldsymbol{t}_1, \dots, \boldsymbol{t}_M\}$ is a set of linearly independent functions, and the linear dependence at s_0 is just an unfortunate coincidence. For instance

$$T(s) = \begin{pmatrix} 1/s & 1/s^2 \\ 1/s^2 & 1/s \end{pmatrix} \tag{5.149}$$

at $s_0 = 1$ has $\operatorname{rank} T(1) = 1$. In such a case, one just has to make a step somewhere into the vicinity of s_0 to see that rank of $T(s)$ is two almost everywhere. For this reason, one defines the **normal rank** of T as

$$\operatorname{Rank} T := \max_{s \in \mathscr{C}} \operatorname{rank} T(s) \tag{5.150}$$

and we know that $\operatorname{Rank} T = \operatorname{rank} T(s)$ almost everywhere. For the invertibility of a linear dynamic input-output system with M inputs, $\operatorname{Rank} T = M$ is necessary and sufficient for analytic invertibility.

From this point on we follow the proof idea of [116] but give an slightly enhanced version so that the results are valid for a dynamic system in Laplace-space. If and only if $\operatorname{rank} T(s) < M$, then $\lambda = 0$ is an eigenvalue of $G(s)$. For most of the calculation, it is convenient to assume s to be a fixed complex number. Due to the argumentation above, it is sufficient to find one $s \in \mathbb{C}$ that renders the transfer function $T(s)$ *locally one-to-one*, in order to make the whole system invertible.

The Gershgorin Circle Theorem [132] has originally been formulated for matrices and has found various applications, compressed sensing as one example. Several extensions, for instance to infinite matrices or to matrices of operators have been developed. The lemma below is actually a corollary of the original Gershorin Circle Theorem, formulated such that it is directly applicable in our setting.

Lemma 6 (Corollary of the Gershgorin Theorem) *Let $G : \mathbb{C} \to \in \mathbb{C}^{M\times M}$ be the Gramian of a linear dynamic input-output system and*

$$\mu_{ab}(s) := \frac{|G_{ab}(s)|}{\sqrt{G_{aa}(s)G_{bb}(s)}}\,. \tag{5.151}$$

We call G strictly diagonally dominant in s if for all $a = 1,\dots,M$ we find

$$G_{aa}(s) > \sum_{b\neq a} |G_{ab}(s)|\,. \tag{5.152}$$

If G is strictly diagonally dominant in s, then zero is not an eigenvalue of $G(s)$.

Proof As a preliminary, note that the diagonal elements of a Gramian are always real and non-negative,

$$G_{ab}(s) = \sum_{b=1}^{P} T^{*}_{ab}(s)T_{ba}(s) = \sum_{b=1}^{P} |T_{ba}(s)|^2\,, \tag{5.153}$$

where P is the number of output nodes, or, equivalently, the number of rows of the transfer function T. A diagonal element G_{aa} is the zero function if and only if each T_{ba} for $b = 1,\dots,P$ is the zero function. That is, if and only if the a-th column of T is zero for all $s \in \mathbb{C}$. Recall the dynamic equation of the system in Laplace-space

$$T(s)\tilde{\boldsymbol{u}}(s) = \tilde{\boldsymbol{y}}(s)\,. \tag{5.154}$$

If the a-th column of T is zero for all $s \in \mathrm{C}$, then the input $\tilde{u}_a$ which targets the input node s_a has no influence on the output at all. It is therefore trivially impossible to gain any information about $\tilde{w}_a$. This trivial case shall be neglected, henceforth the diagonal elements of G are assumed to be non-zero.

For the sake of completeness the full proof including the Gershgorin Circle Theorem is provided in a consistent notation. Let $s \in \mathbb{C}$ such that $G(s)$ achieves the normal rank and set $A := G(s)$. We already know that this works almost everywhere in $\mathbb{C}$. The Gershgorin Circle Theorem states, that all eigenvalues of A lie within circles of a certain radius r_i around the diagonal elements A_{ii}. We want to show that if A strictly diagonally dominant, then r_i is small enough so that none of these circles include zero.

Assume $\lambda \in \mathbb{C}$ is an eigenvalue of A with eigenvector $\boldsymbol{v} \in \mathbb{C}^M$. Let v_i be a component of $\boldsymbol{v}$ of maximal magnitude, i.e.,

$$|v_i| \geq |v_j| \tag{5.155}$$

for all $j = 1,\dots,M$. Without loss of generality we assume $v_i = 1$. The eigenvalue equation in components reads

$$\sum_{j=1}^{M} A_{ij}v_j = \lambda v_i\,. \tag{5.156}$$

We extract $j = i$ from the sum to get

$$\sum_{j \neq i} A_{ij} v_j = \lambda - A_{ii}\,. \tag{5.157}$$

Taking the absolute value and applying the triangle inequality yields

$$|\lambda - A_{ii}| \leq \sum_{j \neq i} |A_{ij}||v_j|\,. \tag{5.158}$$

By construction $|v_j| \leq |v_i| = 1$, thus we get the final result

$$|\lambda - A_{ii}| \leq \sum_{j \neq i} |A_{ij}|\,. \tag{5.159}$$

With $r_i = \sum_{j \neq i} |A_{ij}|$ we can give a bound for the spectrum $\sigma(A)$ via a union of circles in the complex plane,

$$\sigma(A) \subseteq \bigcup_{i=1}^{M} \{\lambda \in \mathbb{C} \,|\, |\lambda - A_{ii}| \leq r_i\}\,. \tag{5.160}$$

As explained above, if we want to exclude the trivial case we can assume $A_{ii} \neq 0$ for each $i = 1, \ldots, M$.

If A is strictly diagonally dominant, then $A_{ii} > r_i$ for each $i = 1, \ldots, M$. Therefore

$$0 \notin \{\lambda \in \mathbb{C} \,|\, |\lambda - A_{ii}| \leq r_i\} \tag{5.161}$$

and consequently

$$0 \notin \sigma(A)\,. \tag{5.162}$$

■

5.5.2 Strict Diagonal Dominance for Invertibility

In an introductory example, we have seen that an appropriate coherence measure can help to circumvent the issue arising when the categorical structure of a matroid encounters random noise, e.g., noisy data. The following definition gives a coherence function for the input nodes of a linear dynamic system in Laplace-space. The linear equation system that appears in the classical compressed sensing problem can be interpreted as a dynamic system where the dynamic part turns out to be trivial. In this sense the definition indeed reproduces the older definition.

Definition 27 *Let $G : \mathbb{C} \to \mathbb{C}^{M \times M}$ be the Gramian of a linear dynamic input-output system with input set $\mathsf{S} = \{\mathsf{s}_1, \ldots, \mathsf{s}_M\}$. The **coherence** of the input nodes $\mathsf{s}_a \in \mathsf{S}$ and $\mathsf{s}_b \in \mathsf{S}$ is*

$$\mu_{ab}(s) := \frac{|G_{ab}(s)|}{\sqrt{G_{aa}(s) G_{bb}(s)}}\,. \tag{5.163}$$

Example (2-In-2-Out System)

Consider a linear system with transfer function

$$T(s) = \begin{pmatrix} 1/s & 1/s^2 \\ 1/s^2 & 1/s \end{pmatrix}. \tag{5.164}$$

The number of columns indicates that there are two input nodes, hence there exists only one coherence which turns out to be

$$\mu_{12}(s) = \frac{2\Re(s)}{1+|s|^2}. \tag{5.165}$$

In polar coordinates $s = re^{i\varphi}$ one also finds a nice formulation

$$\mu_{12}(r,\varphi) = 2\cos(\varphi)\frac{r}{1+r^2}, \tag{5.166}$$

where one can see that the poles of the transfer function disappear in the coherence function. We apply a numerical inverse Laplace-transform, see figure 5.3 for the result. In Laplace-space the coherence $\mu_{12}(r,\varphi)$ approaches the worst case

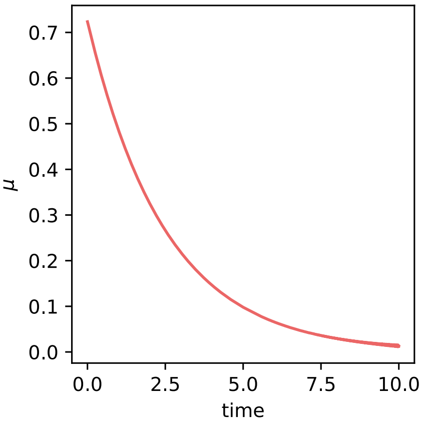

Figure 5.3: The result of the numerical inverse Laplace-transform of the coherence of the exemplary 2-in-2-out system.

$$\mu_{12}(1,0) = 1 \tag{5.167}$$

but tends to the best case

$$\mu_{12}(r,0) \xrightarrow{r\to\infty} 0 \tag{5.168}$$

as the distance r to the origin gets very small or very large. This result indicates that there are some regimes in the domain $\mathbb{C}$, for which the two inputs are very coherent hence hardly distinguishable, but there are also regimes, for which the coherence nearly vanishes.

The example above presents a property of the coherence in a dynamic system, which is absent in the static case: The coherence in a dynamic system is a quantity that evolves over time. Note that the coherence $\mu_{ab} : \mathbb{C} \to [0,1]$ is defined in Laplace-space. As shown in the example, the complex variable $s \in \mathbb{C}$ and the time t in state space are connect via the Laplace-transform so that one could principally apply an inverse Laplace-transform to see how the coherence develops over time.

This development can clearly go into both directions: If two input nodes are coherent in a certain domain, they can still be distinguishable in another, but also vice versa. It is therefore fair to distinguish between the **local coherence** $\mu_{ab}(s)$ which represents the distinguishability of two input nodes s_a and $\mathsf{s}_b \in \mathsf{S}$ in a vicinity of $s \in \mathbb{C}$, and the **global coherence**

$$\mu_{ab} := \inf_{s\in\mathbb{C}} \mu_{ab}(s), \tag{5.169}$$

which tells, whether s_a and s_b are distinguishable *somewhere*.

Mutual Coherence To carry on the thoughts above: What happens in a system with three input nodes s_1, s_2 and s_3? We get three pair-wise dynamic coherences $\mu_{12}(s)$, $\mu_{13}(s)$ and $\mu_{23}(s)$. As before, these coherences can be close to one in some regions and close to zero in others. Say for a region $U \subseteq \mathbb{C}$ we find

$$\mu_{12}(s) \approx 0 \quad \text{for } s \in U \tag{5.170}$$

and for another region $V \subseteq \mathbb{C}$

$$\mu_{13}(s) \approx 0 \quad \text{for } s \in V\,. \tag{5.171}$$

Now what happens, if $U \cap V = \emptyset$? Then for $s \in U$ we can distinguish s_1 and s_2, but we cannot distinguish s_1 from s_3. Symbolically, one could write

$$(\mathsf{s}_1, \mathsf{s}_2, \mathsf{s}_3) = (\mathsf{s}_1, \mathsf{s}_3) \oplus \mathsf{s}_2\,, \tag{5.172}$$

To *distinguish* two nodes s_1 and s_2 means the ability to decide from the output of the system, which input node was truly affected by an unknown input. The true unknown input could attack only s_1, or s_3, or even both.

But what we *can* decide is whether the model error lies somewhere in $\{\mathsf{s}_1, \mathsf{s}_3\}$ or on the node s_2. In the case $s \in V$, we get a different result,

$$(\mathsf{s}_1, \mathsf{s}_2, \mathsf{s}_3) = (\mathsf{s}_1, \mathsf{s}_2) \oplus \mathsf{s}_3\,. \tag{5.173}$$

In this region, we can decide whether an input attacks $(\mathsf{s}_1, \mathsf{s}_2)$ or s_3.

Full invertibility is only achieved, when for some subspace of the complex plane we get the separation

$$(\mathsf{s}_1, \mathsf{s}_2, \mathsf{s}_3) = \mathsf{s}_1 \oplus \mathsf{s}_2 \oplus \mathsf{s}_3\,, \tag{5.174}$$

meaning, each input can be reconstructed individually. Hence, for observability it is necessary that all coherences μ_{ab} tend to zero in the *same* region of the complex plane. It makes sense to introduce the **mutual coherence** at s,

$$\mu(s) := \max_{a \neq b} \mu_{ab}(s)\,. \tag{5.175}$$

While the pair-wise coherence $\mu_{ab}(s)$ yields information if a pair of input nodes is distinguishable, the mutual coherence is the relevant quantity to make a statement whether the whole system is invertible or not.

Global Mutual Coherence From the argumentation above, one will understand that the mutual coherence $\mu(s)$ carries the information about the overall distinguishability of input nodes at a certain point $s \in \mathbb{C}$. For instance the pairwise coherence $\mu_{ab}(s)$ might shrink to zero at some point s_0,

$$\mu_{ab}(s) \overset{s \to s_0}{\longrightarrow} 0\,. \tag{5.176}$$

This indicates the distinguishability of the input nodes a and s_b but it does not indicate invertibility, as the coherence with other possible input nodes $\mathsf{s}_c, \mathsf{s}_d, \ldots$ can make the accurate reconstruction impossible.

However, we can still exploit the fact that invertibility at one $s_0 \in \mathbb{C}$ is sufficient to show that the system is invertible almost everywhere. Hence it is fair to introduce the **global mutual coherence**

$$\mu := \inf_{s \in \mathbb{C}} \mu(s) . \tag{5.177}$$

An note of caution is stated with the inequality

$$\max_{a \neq b} \inf_{s \in \mathbb{C}} \mu_{ab}(s) \leq \inf_{s \in \mathbb{C}} \max_{a \neq b} \mu_{ab}(s) , \tag{5.178}$$

or in terms of the global coherence μ_{ab} and global mutual coherence μ

$$\max_{a \neq b} \mu_{ab} \leq \mu . \tag{5.179}$$

Shortest and Longest Path Coherence The varieties of the coherence measure discussed above yield information about the distinguishability of pairs or sets of input nodes. Unfortunately, they all suffer from the issue, that one has to compute the transfer matrix and search for the infimum of a function in the complex plain. The **shortest path coherence**

$$\mu_{ab}^{\text{short}} := \lim_{|s| \to \infty} \mu_{ab}(s) \tag{5.180}$$

provides relief.

In the discussion of the Gramian we have already found proposition 7 which we can now write as

$$\mu_{ab}^{\text{short}} \leq \frac{|F(\eta_{ab'})|}{\sqrt{F(aa')F(\eta_{bb'})}} \tag{5.181}$$

where $\eta_{ab'}$ denotes the shortest path (or set of shortest paths) through the gramian gammoid $(\Gamma \circ \Gamma')$ from s_a to s'_b.

Proposition 8 *Let $\mu_{ab}(s)$ be the coherence of two input nodes s_a and s_b. Then*

$$\lim_{|s| \to \infty} \max_{a \neq b} \mu_{ab}(s) = \max_{a \neq b} \lim_{|s| \to \infty} \mu_{ab}(s) \tag{5.182}$$

Proof First note that $\mu_{ab}(s)$ is continuous, $0 \leq \mu_{ab}(s) \leq 1$, and that any singularity of $\mu_{ab}(s)$ is removable. Furthermore we have already seen that each $\mu_{ab}(s)$ convergent with limit μ_{ab}^{short} as $|s| \to \infty$. Due to this behaviour it is sufficient to consider only real s.

To simplify the notation let $f_a : \mathbb{R}_{\geq 0} \to \mathbb{R}$ a family of continuous and bounded functions with $a \in I$ where I is a finite index set. We want to show that

$$\lim_{x \to \infty} \max_{a \in I} f_a(x) = \max_{a \in I} \lim_{x \to \infty} f_a(x) . \tag{5.183}$$

To see that, let $M_a := \lim_{x \to \infty} f_a(x)$ and let $a^* \in I$ such that $M_{a^*} = \max_{a \in I} M_a$. By definition of M_a, for any $\epsilon > 0$ there is an x_a such that for all $x > x_a$ we have

$$|f_a(x) - M_a| < \epsilon . \tag{5.184}$$

Set $x_0 := \max_{a\in I} x_a$ such that we can use the same epsilon and x_0 for all indices a. Let us divide $I = I_0 \dot\cup I_1$ such that I_0 contains all indices with $M_a < M_{a^*}$ and I_1 contains those indices with $M_a = M_{a^*}$

Let us first consider all I_0 and let us choose ϵ such that

$$\epsilon < \frac{1}{2}|M_{a^*} - M_a| \tag{5.185}$$

for all $a \in I_0$. We can rewrite this as

$$M_a + \epsilon < M_{a^*} - \epsilon\,. \tag{5.186}$$

For this choice of ϵ there is an x_0 such that $|f_a(x) - M_a| < \epsilon$ for $x > x_0$, i.e.,

$$f_a(x) \in (M_a - \epsilon, M_a + \epsilon) \tag{5.187}$$

and analogously

$$f_{a^*}(x) \in (M_{a^*} - \epsilon, M_{a^*} + \epsilon)\,. \tag{5.188}$$

Due to our choice of epsilon these two intervals are disjoint and one can see that $f_a(x) < f_{a^*}(x)$ for all $x > x_0$. Thus for x large enough we can always neglect I_0,

$$\max_{a\in I} f_a(x) = \max_{a\in I_1} f_a(x)\,. \tag{5.189}$$

Let us now focus on I_1. Consider

$$r_a(x) := |f_a(x) - M_{a^*}| \tag{5.190}$$

for $a \in I_1$ and let $a' \in I_1$ such that $f_{a'}(x) = \max_{a\in I_1} f_a(x)$. It is now clear that

$$r_{a'}(x) \le \max_{a\in I_1} r_a(x) \tag{5.191}$$

for all x. Insertion of the definitions yields

$$\left|f_{a'}(x) - M_{a^*}\right| \le \max_{a\in I_1} |f_a(x) - M_{a^*}| \tag{5.192}$$

and for $x > x_0$ we deduce

$$\left|f_{a'}(x) - M_{a^*}\right| < \epsilon\,. \tag{5.193}$$

Since we can do that for arbitrarily small $\epsilon > 0$ we find the convergence

$$\lim_{x\to\infty} f_{a'}(x) = M_{a^*}\,. \tag{5.194}$$

We can now insert the definitions

$$f_{a'}(x) = \max_{a\in I_1} f_a(x) = \max_{a\in I} f_a(x) \tag{5.195}$$

and

$$M_{a^*} = \max_{a\in I} M_a = \max_{a\in I} \lim_{x\to\infty} f_a(x) \tag{5.196}$$

to get the desired result

$$\lim_{x\to\infty}\max_{a\in I} f_a(x) = \max_{a\in I}\lim_{x\to\infty} f_a(x)\,. \tag{5.197}$$

■

In the discussion of the global mutual coherence we have seen that the maximum and infimum operations do not commute. As a consequence, if we want to compute the global mutual coherence, we have to handle the possibly piece-wise defined mutual coherence

$$\mu(s) = \max_{a\neq b}\mu_{ab}(s) \tag{5.198}$$

and search for the infimum. But the proposition above shows that the maximum and the limit $s\to\infty$ do commute. It is therefore allowed to compute the limit

$$\mu_{ab}^{\text{short}} = \lim_{s\to\infty}\mu_{ab}(s) \tag{5.199}$$

before taking the maximum of the produced real-matrix $(\mu_{ab})_{1\leq a,b\leq M}$. This reduces the estimation of the global mutual coherence to

$$\mu \leq \max_{a\neq b}\mu_{ab}^{\text{short}} \tag{5.200}$$

where each μ_{ab}^{short} can individually be computed via a simple *shortest path problem* for which there exists a broad range of theories and algorithms in the literature, see for instance the textbook [87].

We may add a brief additional finding for the sake of consistency. In analogy to the shortest path coherence can also consider the **longest path coherence**

$$\mu_{ab}^{\text{long}} \leq \frac{|F(\psi_{ab'})|}{\sqrt{F(\psi_{aa'})F(\psi_{bb'})}} \tag{5.201}$$

where $\psi_{ab'}$ denotes the longest path (or set of longest paths) through $\Gamma\circ\Gamma'$ from s_a to s'_b. Unfortunately, many real world dynamic systems have not a tree-like structure but incorporate loops and often self-loops. As a consequence, μ_{ab}^{long} is not well-defined in such scenarios. Given that there is a maximal path length, we find in analogy to proposition 7 that

$$\mu_{ab}^{\text{long}} \leq \lim_{s\to 0}\mu_{ab}(s)\,. \tag{5.202}$$

Diagonal Dominance In [119] the following proposition is presented for real, constant matrices, which represents only a special case of the more general transfer function T. We can now show its validity for the non-constant and complex transfer function of a linear dynamic system.

Proposition 9 *Let $s\in\mathbb{C}$ be fixed, $G:\mathbb{C}\to\mathbb{C}^{M\times M}$ a Gramian and $\mu(s)$ the mutual coherence at s. If the inequality*

$$G_{ii}(s) > \mu(s) \tag{5.203}$$

holds for $i = 1,\ldots,M$, then G is strictly diagonally dominant at s.

Proof For convenience, rescale the system by

$$G(s) \rightsquigarrow \frac{G(s)}{\operatorname{tr} G(s)} \tag{5.204}$$

which gives the property

$$\sum_{i=1}^{M} G_{ii}(s) = 1\,. \tag{5.205}$$

As in the proof of proposition 7, $G_{ii}(s) > 0$ and from the equation above $G_{ii} < 1$. The special case $M = 1$ is trivial. By definition of the mutual coherence at s we find for all $i, j = 1, \ldots, M$ with $i \neq j$

$$|G_{ij}(s)| \leq \mu(s) \sqrt{G_{ii}(s) G_{jj}(s)}\,. \tag{5.206}$$

By assumption of the proposition $\mu(s) \leq G_{ii}(s)$ for all $i = 1, \ldots, M$, and since all quantities are non-negative

$$\mu < \sqrt{G_{ii}(s) G_{jj}(s)} \tag{5.207}$$

for all $i \neq j$. Combine the latter two inequalities and sum over all $j = 1, \ldots, i-1, i+1, \ldots, M$ to get

$$\sum_{j \neq i} |G_{ij}(s)| < G_{ii}(s) \sum_{j \neq i} G_{jj}(s) \tag{5.208}$$

for all $i, \ldots, M$. Due to equation (5.205), the sum on the right hand side is smaller than one, thus for all $i = 1, \ldots, M$ we find

$$\sum_{j \neq i} |G_{ij}(s)| < G_{ii}(s) \tag{5.209}$$

which is exactly the definition of strict diagonal dominance at s.

■

5.5.3 Estimation of the Spark

So far we have always assumed there is a connection between the coherence and the possibility to distinguish the input nodes. To make this a trustworthy statement, we show that the mutual coherence indeed yields a lower bound for the spark of a gammoid. The same bound is already known for the spark and mutual coherence of a real-valued matrix. As we have seen that the spark and mutual coherence are quantities that can be generalised to weighted gammoids, it is not surprising, that with the generalised notions of spark and mutual coherence, we can do an analogous calculation to get a generalised theorem.

By definition of the mutual coherence at s we get

$$|G_{ij}(s)| \leq \mu(s) \sqrt{G_{ii}(s) G_{jj}(s)}\,. \tag{5.210}$$

Now let $G_{kk}(s)$ be the maximum of all $G_{ii}(s)$ for $i = 1, \ldots, M$. Replace the square root and sum over all $j = 1, \ldots, i-1, i+1, \ldots, M$ to get

$$\sum_{j \neq i} |G_{ij}(s)| \leq (M-1)\mu(s) G_{kk}(s)(s)\,, \tag{5.211}$$

where we used that the right hand side is independent of j. One finds the Gramian to be strictly diagonally dominant if for all $i = 1, \dots, M$

$$(M-1)\mu(s)G_{kk}(s)(s) < G_{ii}(s)\,. \tag{5.212}$$

The latter inequality is therefore sufficient for strict diagonal dominance at s. Use $G_{ii}(s)/G_{kk}(s) < 1$ to get

$$M < \frac{1}{\mu(s)} + 1\,. \tag{5.213}$$

Note that the gammoid G depends on the choice of the input set $\mathtt{S} \subseteq \mathtt{L}$. However, the sufficient condition for strict diagonal dominance only takes $M = \operatorname{card}\mathtt{S}$ into account. It becomes clear that for all $\tilde{\mathtt{S}} \subseteq \mathtt{L}$ with $\operatorname{card}\tilde{\mathtt{S}} \leq M$ we get strict diagonal dominance. Since strict diagonal dominance leads to structural invertibility, it also tells us that $\tilde{\mathtt{S}}$ is independent Γ. By definition, $\operatorname{spark}\Gamma$ is the largest integer such that $\tilde{\mathtt{S}}$ is independent in Γ whenever $\operatorname{card}\tilde{\mathtt{S}} \leq \operatorname{spark}\Gamma$. Since any M that fulfils (5.213) leads to independence, the spark is not smaller than the right hand side of this inequality. Therefore

$$\operatorname{spark}\Gamma \geq \frac{1}{\mu(s)} + 1\,. \tag{5.214}$$

The best estimation of the spark is attain for the global mutual coherence $\mu = \inf_{s\in\mathbb{C}} \mu(s)$.

5.5.4 Feed-Forward Graphs

Let us finally discuss a special class of dynamic systems.

Definition 28 *Let* $\mathtt{g} = (\mathtt{N}, \mathtt{E})$ *be the influence graph of a dynamic system. If one can split*

$$\mathtt{N} = \mathtt{N}_0 \dot{\cup} \dots \dot{\cup} \mathtt{N}_L \tag{5.215}$$

such that for all edges $(\mathtt{a} \to \mathtt{b}) \in \mathtt{E}$ *we find* $\mathtt{a} \in \mathtt{N}_{i-1}$ *and* $\mathtt{b} \in \mathtt{N}_i$ *for some* $i \in \{1, \dots, L\}$*, then we call the system* ***feed-forward****. We call* $\mathtt{N}_i$ *the* i^{th} ***layer****. The gammoid*

$$\Gamma = (\mathtt{N}_0, \mathtt{g}, \mathtt{N}_L) \tag{5.216}$$

is called a ***cascade*** *gammoid. We call* $\mathtt{N}_0$ *the* ***input layer*** *and* $\mathtt{N}_L$ *the* ***output layer****.*

For $L = 1$*, the graph is called* ***bipartite*** *and the gammoid is called a* ***deltoid****.*

In a cascade gammoid, all paths from the input layer to the output layer have the same length L. We find the transfer function to be

$$T_{ji}(s) = \frac{1}{s^L} \sum_{\pi \in \mathscr{P}(\mathtt{s}_i, \mathtt{z}_j)} F(\pi) \tag{5.217}$$

where $s_i \in \mathtt{N}_0$ and $z_j \in \mathtt{N}_L$. Set

$$a_{ji} := \sum_{\pi \in \mathscr{P}(\mathtt{s}_i, \mathtt{z}_j)} F(\pi) \tag{5.218}$$

to get the simple form

$$T(s) = \frac{1}{s^L} A \tag{5.219}$$

with a real $P \times M$ matrix $A = (a_{ij})_{1 \le i \le P, 1 \le j \le M}$. Calling $\boldsymbol{a}_i$, the i^{th} column of A, we find the coherence $\mu_{ij}(s)$ to be a constant

$$\mu_{ij}(s) = \frac{\langle a_i, a_j \rangle}{\| a_i \| \| a_j \|}. \tag{5.220}$$

The latter equation looks exactly like the coherence one defines for static compressed sensing problems, see the textbook [116]. In this particular case we find

$$\mu = \mu^{\text{short}} \tag{5.221}$$

that means, in feed-forward systems, the shortest path coherence is exact.

5.6 Overview: Gammoids for Sparse Sensing of Model Errors

After the last two sections equipped us with rather technical results for weighted gammoids, let us briefly condense the results of this chapter.

Starting with a successful model error reconstruction in a non-invertible system, we came to the hypothesis that the *sparsity* of the model error is the reason why it was so easy to pick the input $\boldsymbol{w}^*$ from the infinite dimensional input space $\mathscr{U}$, which describes the model error correctly. We then discussed the *matroid* as the minimal structure to give reason to the term sparsity and showed that dynamic input-output systems fulfil the postulates of matroid theory and for the special class of *gammoids*. Due to our results on structural invertibility, we were able to connect the *input ground set* L of a gammoid to the restricted input space $\mathscr{U}_{\mathrm{L}}$. Though

$$\Phi : \mathscr{U} \to \mathscr{Y} \tag{5.222}$$

might not be invertible, but

$$\Phi : \mathscr{U}_{\mathrm{L}} \to \mathscr{Y} \tag{5.223}$$

can still be invertible.

The system with input set $\mathrm{S} \subseteq \mathrm{L}$ is invertible if and only if S is an *independent set* of the gammoid. This motivated the formulation of the *Model Error Localisation Problem*: If the system is not invertible, first find the independent input sets of the system. If the model error lies within one independent set S, then we can still reconstruct the model error, even if the full system might be non-invertible. With this result, the invertibility problem is completely formulated in the language of gammoids. We have shown that the *spark* of a gammoid yields a theorem that ensures the *uniqueness of the sparsest solution* for linear and non-linear systems. This proves that we can find the correct model error $\boldsymbol{w}^*$, given the model error is sparse, even if the system is non-invertible.

For the special case of linear systems, we discussed *measures of coherence.* These coherences make a statement about the distinguishability of input nodes and by this about the distinguishability of model errors. We presented the *shortest path coherence* which reduces the computation of the coherence to a shortest path problem on a graph. The coherence measures will play an important role in the *cluster approach* presented in chapter 7.

CHAPTER 6

Dynamic Compressed Sensing

Chapter 5 has shown that the assumption of sparsity provides uniqueness for the MERP. For this result we have utilised the intrinsic gammoid structure of dynamic input-output systems. Unfortunately, as discussed below, in real problems it is likely to happen that stochastic noise corrupts the gammoid structure.

Say, a single model error perturbs state variable x_1, so that the true input vector takes the form

$$\boldsymbol{w}^* = (w_1, 0, 0, \ldots, 0)^T, \tag{6.1}$$

so that

$$\boldsymbol{y}^{\text{noiseless}} = \Phi(\boldsymbol{w}^*) \tag{6.2}$$

but

$$\boldsymbol{y}^{\text{data}} = \Phi(\boldsymbol{w}^*) + \xi \tag{6.3}$$

where ξ represents some noise. Due to the noise, the data vector $\boldsymbol{y}^{\text{data}}$ cannot be fitted by a sparse input any more, but by some input

$$\boldsymbol{u} = (w^* + v_1, v_2, \ldots, v_N)^T \tag{6.4}$$

where $\boldsymbol{v}$ is some input of small magnitude, only there to fit the noise.

Clearly, fitting the noise should not be our goal. Instead, we search for a way to suppress the terms v_i which actually do not stem from the structural model error.

After an introductory example, this chapter will introduce a norm on the input and output spaces of a dynamic input-output system that is consistent with our notion of sparsity. We will then discuss the sparse sensing problem P0 and the relaxed convex problem P1. With an appropriately formulated RIP2k we will show how we can use the relaxed problem to solve the compressed sensing problem for model errors in *linear* dynamic systems.

From Sparse Sensing to Compressed Sensing Let us take a closer look on a classical problem of compressed sensing [119]. Let $\boldsymbol{y} \in \mathbb{R}^P$ and $A \in \mathbb{R}^{P \times N}$ be given with $P << N$: Solve

$$A\boldsymbol{x} = \boldsymbol{y} \tag{6.5}$$

for $\boldsymbol{x}$. From linear algebra it is clear that this problem is highly under-determined and the solution space is at least $N-P$ dimensional. In the spirit of sparse sensing, we search for a vector $\boldsymbol{x} \in \mathbb{R}^N$ that solves the equation above and at the same time has a minimal number of non-zero components. This task can be formulated as the optimisation problem

$$\begin{aligned}&\text{minimise } \|\boldsymbol{x}\|_0 \\ &\text{subject to } A\boldsymbol{x} = \boldsymbol{y}\end{aligned} \tag{6.6}$$

where $\|\cdot\|_0$ counts the number of non-zero components. In reality, a vector of measured quantities like $\boldsymbol{y}$ will always incorporate measurement noise. So the equation to solve would rather read

$$A\boldsymbol{x} + \xi = \boldsymbol{y}^{\text{noisy}} \tag{6.7}$$

where the vector ξ is a realisation of, e.g., a Gaussian noise. Usually one has some knowledge or good assumptions about the noise vector. For instance, in the case of white noise we have (in the expectation value)

$$\sum_{i=1}^{P} \xi_i = 0 \tag{6.8}$$

or in the case of bounded noise

$$\sum_{i=1}^{P} |\xi_i|^2 < \epsilon \tag{6.9}$$

where ϵ is a known boundary for the noise. The existence of such an additional, unknown vector causes an issue for the idea of sparse sensing.

Example (Directions)

The matrix

$$A = \begin{pmatrix} 1 & 1/\sqrt{2} & 0 & 0 & 1/\sqrt{3} & 1/\sqrt{2} & 0 \\ 0 & 1/\sqrt{2} & 1 & 1/\sqrt{2} & 1/\sqrt{3} & 0 & 0 \\ 0 & 0 & 0 & 1/\sqrt{2} & 1/\sqrt{3} & 1/\sqrt{2} & 1 \end{pmatrix} \tag{6.10}$$

comprises some directions in the first octant of $\mathbb{R}^3$. Say we aim for a sparse decomposition of

$$\boldsymbol{y} = \begin{pmatrix} 2 \\ 3 \\ 1 \end{pmatrix} \tag{6.11}$$

with respect to the dictionary of directions defined by A. One will find that the sparsest solution of

$$A\boldsymbol{x} = \boldsymbol{y} \tag{6.12}$$

is the vector

$$\boldsymbol{x} = (0, \sqrt{8}, 0, \sqrt{2}, 0, 0, 0)^T . \tag{6.13}$$

However, as soon as a noise vector, e.g.,

$$\xi = \begin{pmatrix} 0.01 \\ -0.03 \\ 0.01 \end{pmatrix} \tag{6.14}$$

is added, a sparse decomposition of

$$\boldsymbol{y}^{\text{noisy}} = \begin{pmatrix} 2.01 \\ 2.97 \\ 1.01 \end{pmatrix} \tag{6.15}$$

needs at least three non-zero components. Since A serves as a complete but redundant dictionary it is obvious that there exists a solution of

$$A\boldsymbol{x} = \boldsymbol{y}^{\text{noisy}}, \tag{6.16}$$

but this solution is neither sparse nor unique.

Apart from the loss of the matroid structure due to the noisy measurements, another practical aspect causes trouble. The optimisation problem 6.6 is numerically hard to solve, due to the $\|\cdot\|_0$ term, see [133] for a good overview over sparse optimisation problems and the references therein. Optimisation problems of this and similar kinds are subject of current research and the literature mentioned can only give a small excerpt of this topic, which lies beyond the scope and possibilities of this thesis. The idea of the LASSO-regularisation, stemming from [134] and reinvented by Tibshirani [135], shall however be presented, as it is a central idea to handle both issues at once.

LASSO-Regularisation for Compressed Solutions The noise vector ξ has no sparse representation, hence to searching for a sparse solution of

$$A\boldsymbol{x} = \boldsymbol{y}^{\text{noisy}} \tag{6.17}$$

would not make sense. But exploiting the standard $\|\cdot\|_2$-norm on $\mathbb{R}^N$

$$\|A\boldsymbol{x} - \boldsymbol{y}^{\text{noisy}}\|_2 = \|\xi\|_2 < \epsilon\,. \tag{6.18}$$

Hence, one can try to suppress the effects of noise by minimising the term

$$\|A\boldsymbol{x} - \boldsymbol{y}^{\text{noisy}}\|_2 \tag{6.19}$$

instead of finding an exact solution. At the same time, it has been shown that, under certain conditions, minimisation of the 1-norm

$$\|\boldsymbol{x}\|_1 := |x_1| + \ldots |x_N| \tag{6.20}$$

also leads to sparsity. The idea of LASSO-regularisation is to solve the slightly softened problem

$$\text{minimise}\|A\boldsymbol{x} - \boldsymbol{y}^{\text{noisy}}\|_2^2 + \|\boldsymbol{x}\|_1\,. \tag{6.21}$$

Problem Formulation From the considerations above, we can reformulate the Model Error Reconstruction Problem, incorporating sparsity ($\|\cdot\|_0$-norm) and noise ($\epsilon > 0$) as the problem P0.

Problem (P0)

$$\text{minimise}\,\|\boldsymbol{u}\|_0\,\text{s.t.}\,\|\Phi(\boldsymbol{u}) - \boldsymbol{y}^{\text{data}}\|_2 < \epsilon. \tag{6.22}$$

As explained above, the P0 problem is hard to solve in practice and it was soon replaced by the LASSO problem P1.

Problem (P1)

$$\text{minimise}\,\|\boldsymbol{u}\|_1\,\text{s.t.}\,\|\Phi(\boldsymbol{u}) - \boldsymbol{y}^{\text{data}}\|_2 < \epsilon. \tag{6.23}$$

The problems P0 and P1 have been written in the typical forms of compressed sensing problems, compare for instance the problems in [116], but we must confess that the *norms* on the input and output space are not yet defined. With the exception of the 0-norm which is defined via the support and not a proper norm.

The first step in this chapter must therefore be the definition of a proper and appropriate norm for input and output vectors, so that the analogous formulations of P0 and P1 make sense for dynamic systems. Then, it will be shown that P1 is a convex optimisation problem. An adjusted form of the *Restricted-Isometry-Property* [61] is given and it will be proven that under this condition, the solution of P1 is the best possible approximation of P0. The final result will be the proof of the following theorem which one will fully understand at the end of this chapter.

Theorem 10 *Assume Φ is linear and the RIP of order $2k$ holds. Let $\boldsymbol{w}^*$ be the solution of P0 and $\hat{\boldsymbol{w}}$ the solution of P1. Then*

$$\|\hat{\boldsymbol{w}} - \boldsymbol{w}^*\|_2 \le C_0 \frac{\sigma_k(\boldsymbol{w}^*)_1}{\sqrt{k}} + C_2\epsilon \tag{6.24}$$

with non-negative constants C_0 and C_2.

6.1 Spaces and Norms

The first section is dedicated to the norms on the input and output spaces of the input-output system in state space formulation. An input vector $\boldsymbol{u} \in \mathcal{U}$

$$\boldsymbol{u} = \begin{pmatrix} u_1 \\ \vdots \\ u_M \end{pmatrix} \tag{6.25}$$

is a vector of functions. The *Dynamic Elastic-Net* from [21] or the *CLOT* from [120] have suggested norms which are motivated by the scalar product

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = \sum_{i=1}^{M} \int_0^T u_i(t)\, v_i(t)\, \mathrm{d}\, t \tag{6.26}$$

which can be defined on $\mathscr{U}$.

In [120] it was shown that the regularisation with the induced norm can produce a *hands-off* control, meaning an input with a small *temporal support*. For the concept of sparsity which is appropriate for the MERP, we do not want to restrict the time interval where the model errors takes places, but their number.

6.1.1 Input Space

Henceforth we will assume the input component $u_i \in \mathscr{U}_i$ to lie in $L^p([0,T])$ for a fixed p and be piecewise $C^\infty[0,T]$.

Underline We first introduce the underline notation

$$\underline{u_i} := \|u_i\|_{L^p} \tag{6.27}$$

and for the vector $\boldsymbol{u} \in \mathscr{U}$ we write

$$\underline{\boldsymbol{u}} := \begin{pmatrix} \underline{u_1} \\ \vdots \\ \underline{u_L} \end{pmatrix}, \tag{6.28}$$

so the underline operator maps $\boldsymbol{u}$ to a vector $\underline{\boldsymbol{u}} \in \mathbb{R}^N$. In the more general setting, $\mathscr{U}_i$ is only assumed to be a Banach space with some norm $\|\cdot\|_{\mathscr{U}_i}$ and the underline maps according to this norm. The underline notation will help clarify our understanding of sparsity and will be useful for the formulation of theorems and proofs. As the underline is basically a norm, it inherits the properties of a norm. The following lemma merely represent calculation rules for underlined vectors.

Lemma 7 *1. For $u_i \in \mathscr{U}_i$ we find $\underline{u_i} = 0 \Leftrightarrow u_i(t) = 0$ a.e. in $[0,T]$.*

2. For $u_i, v_i \in \mathscr{U}_i$ we find $\underline{u_i + v_i} \leq \underline{u_i} + \underline{v_i}$.

3. For $u_i \in \mathscr{U}_i$ and $a \in \mathbb{R}$ we find $\underline{au_i} = |a|\underline{u_i}$.

Proof

1. From the definition of the L^p spaces we get

$$\|u_i\|_{L^p} = 0 \Leftrightarrow u_i = 0 \text{ a.e.} \tag{6.29}$$

2. Again from the definition we find

$$\underline{u_i + v_i} = \|u_i + v_i\|_{L^p} \leq \|u_i\|_{L^p} + \|v_i\|_{L^p} = \underline{u_i} + \underline{v_i}. \tag{6.30}$$

3. Since the L^p-norm is homogeneous we have

$$\underline{au_i} = \|au_i\|_{L^p} = |a|\|u_i\|_{L^p} = |a|\underline{u_i}. \tag{6.31}$$

■

Proper Norm Utilising the underline notation, we are ready to define the q-norm on $\mathscr{U}$ as

$$\|\boldsymbol{u}\|_q := \|\underline{\boldsymbol{u}}\|_q \tag{6.32}$$

where on the right hand side the standard q-norm on $\mathbb{R}^L$ is understood.

Proposition 10 *The q-norm on $\mathscr{U}$ is a proper norm.*

Proof The L^p norm and the q-norm on $\mathbb{R}^N$ both are proper norms that fulfil the three properties *positive definiteness*, the *triangle property*, and *homogeneity*. Since the q-norm on $\mathscr{U}$ is a combination of these two norms, it again fulfils these properties.

1. Positive definiteness. We find
$$\|\boldsymbol{u}\|_q = \|\underline{\boldsymbol{u}}\|_q \geq 0 \tag{6.33}$$
where the inequality comes from the fact that we have a proper norm on $\mathbb{R}^L$. Equality holds if and only if $\underline{\boldsymbol{u}} = 0$. Due to lemma 7 this is the case if and only if $\boldsymbol{u} = 0$.

2. Triangle inequality. Let $\boldsymbol{u}, \boldsymbol{v} \in \mathscr{U}$.
$$\|\boldsymbol{u}+\boldsymbol{v}\|_q = \|\underline{\boldsymbol{u}+\boldsymbol{v}}\|_q = \left(\sum_{i=1}^{N} \underline{(u_i + v_i)^q}\right)^{1/q} \tag{6.34}$$
From lemma 7
$$\underline{(u_i + v_i)^q} \leq (\underline{u_i} + \underline{v_i})^q \tag{6.35}$$
so we get
$$\|\boldsymbol{u}+\boldsymbol{v}\|_q \leq \left(\sum_{i=1}^{N} (\underline{u_i} + \underline{v_i})^q\right)^{1/q} = \|\underline{\boldsymbol{u}} + \underline{\boldsymbol{v}}\|_q \tag{6.36}$$
and since the latter is a proper q-norm on $\mathbb{R}^L$
$$\|\boldsymbol{u}+\boldsymbol{v}\|_q \leq \|\underline{\boldsymbol{u}}\|_q + \|\underline{\boldsymbol{v}}\|_q = \|\boldsymbol{u}\|_q + \|\boldsymbol{v}\|_q. \tag{6.37}$$

3. Finally we proof homogeneity. Let $\boldsymbol{u} \in \mathscr{U}$ and $a \in \mathbb{R}$.
$$\|a\boldsymbol{u}\|_q = \|\underline{a\boldsymbol{u}}\|_q = \left(\sum_{i=1}^{L} \underline{(au_i)}^q\right)^{1/q} = \left(\sum_{i=1}^{L} (|a|\underline{u_i})^q\right)^{1/q} = |a|\left(\sum_{i=1}^{L} (\underline{u_i})^q\right)^{1/q} = |a|\|\boldsymbol{u}\|_q \tag{6.38}$$

■

Disjoint Supports Remember the definition of the *support* of input vectors. One will see that the space of k-sparse inputs, Σ_k, is the union of all $\mathscr{U}_\Lambda$ with $\operatorname{card}\Lambda \leq k$, where $\Lambda \subseteq \{1,\ldots,N\}$ are index sets.

The three following facts can be proven directly: Let Λ_0 and Λ_1 be two index sets with $\operatorname{card}\Lambda_0 = \operatorname{card}\Lambda_1 = k$ and let $\boldsymbol{u} \in \mathscr{U}$. We find

$$\boldsymbol{u}_{\Lambda_0}, \boldsymbol{u}_{\Lambda_1} \in \Sigma_k \tag{6.39}$$

and

$$\boldsymbol{u}_{\Lambda_0} + \boldsymbol{u}_{\Lambda_1} \in \Sigma_{2k}\,. \tag{6.40}$$

If Λ_0 and Λ_1 are disjoint we also get

$$\boldsymbol{u}_{\Lambda_0} + \boldsymbol{u}_{\Lambda_1} = \boldsymbol{u}_{\Lambda_0 \cup \Lambda_1}\,. \tag{6.41}$$

Lemma 8 *Let $\boldsymbol{u}, \boldsymbol{v} \in \mathcal{U}$ with disjoint support, then*

$$\underline{\boldsymbol{u} \pm \boldsymbol{v}} = \underline{\boldsymbol{u}} + \underline{\boldsymbol{v}} \tag{6.42}$$

and for the q-norm

$$\|\underline{\boldsymbol{u}} - \underline{\boldsymbol{v}}\|_q = \|\boldsymbol{u} \pm \boldsymbol{v}\|_q = \|\underline{\boldsymbol{u}} + \underline{\boldsymbol{v}}\|_q\,. \tag{6.43}$$

Proof If i is in the support of $\boldsymbol{u}$, then $v_i = 0$ and equation (6.42) reduces to

$$\underline{u_i} = \underline{u_i} \tag{6.44}$$

which is true. If i is in the support of $\boldsymbol{v}$, then $u_i = 0$. Equation (6.42) together with lemma 7 then becomes

$$\underline{\pm v_i} = |\pm 1|\underline{v_i} = \underline{v_i} \tag{6.45}$$

which also holds true. If i is in neither support, the equation becomes trivial. For equation (6.45) note, that

$$\|\boldsymbol{u} \pm \boldsymbol{v}\|_q = \|\underline{\boldsymbol{u} \pm \boldsymbol{v}}\|_q = \|\underline{\boldsymbol{u}} + \underline{\boldsymbol{v}}\|_q \tag{6.46}$$

where the first equality is clear by definition and the second equality was just proven.

It remains to show that for any two vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^N$ with disjoint support one gets

$$\|\boldsymbol{x} - \boldsymbol{y}\|_q = \|\boldsymbol{x} + \boldsymbol{y}\|_q\,. \tag{6.47}$$

Due to the disjoint support $|x_i + y_i|^q = |x_i|^q + |y_i|^q = |x_i - y_i|^q$, since at least one of the two terms is zero. One finds

$$\|\boldsymbol{x} - \boldsymbol{y}\|_q^q = \sum_{i=1}^{N} |x_i - y_1|^q = \sum_{i=1}^{N} |x_i + y_1|^q = \|\boldsymbol{x} + \boldsymbol{y}\|_q^q\,. \tag{6.48}$$

■

The following proposition for the case, where $\boldsymbol{u}, \boldsymbol{v}$ are $\mathbb{R}^N$ vectors, stems from [136] and has already been used for the static problem in [137]. We proof its validity for $\boldsymbol{u}, \boldsymbol{v}$ being input vectors of a dynamic system.

Proposition 11 *Let $\boldsymbol{u}, \boldsymbol{v} \in \mathcal{U}$ with disjoint support, then*

$$\|\boldsymbol{u} + \boldsymbol{v}\|_q^q = \|\boldsymbol{u}\|_q^q + \|\boldsymbol{v}\|_q^q\,. \tag{6.49}$$

Proof With lemma 8 we find

$$\|\boldsymbol{u}+\boldsymbol{v}\|_q^q = \|\underline{\boldsymbol{u}+\boldsymbol{v}}\|_q^q = \|\underline{\boldsymbol{u}}+\underline{\boldsymbol{v}}\|_q^q = \sum_{i=1}^{N} (\underline{u_i}+\underline{v_i})^q . \tag{6.50}$$

Again due to the disjoint support we can write

$$(\underline{u_i}+\underline{v_i})^q = (\underline{u_i})^q + (\underline{v_i})^q \tag{6.51}$$

to get

$$\|\boldsymbol{u}+\boldsymbol{v}\|_q^q = \sum_{i=1}^{N} (\underline{u_i})^q + \sum_{i=1}^{N} (\underline{v_i})^q = \|\boldsymbol{u}\|_q^q + \|\boldsymbol{v}\|_q^q . \tag{6.52}$$

■

We close the investigation of the input space with three lemmas on norm-inequalities. For $\boldsymbol{u}$ being a vector in $\mathbb{R}^N$, these lemmas are proven in [116]. For our purposes, it is necessary to prove the validity for the q-norms on composite Banach spaces.

Lemma 9 *For $\boldsymbol{u} \in \Sigma_k$ we find*

$$\frac{1}{\sqrt{k}}\|\boldsymbol{u}\|_1 \leq \|\boldsymbol{u}\|_2 \leq \sqrt{k}\|\boldsymbol{u}\|_\infty . \tag{6.53}$$

Proof Let $\langle\cdot,\cdot\rangle$ denote the standard scalar product on $\mathbb{R}^N$. We can write

$$\|\boldsymbol{u}\|_1 = \|\underline{\boldsymbol{u}}\|_1 = \langle \underline{u}, \operatorname{sgn}\underline{\boldsymbol{u}}\rangle \leq \|\underline{\boldsymbol{u}}\|_2 \|\operatorname{sgn}\underline{\boldsymbol{u}}\|_2 , \tag{6.54}$$

where the latter inequality comes from Cauchy-Schwartz and the signum function is understood component-wise. By assumption $\boldsymbol{u}$ is k-sparse, hence $\|\operatorname{sgn}\underline{\boldsymbol{u}}\|_2^2$ is a sum of at most k ones. We obtain

$$\|\boldsymbol{u}\|_1 \leq \sqrt{k}\|\underline{\boldsymbol{u}}\|_2 = \sqrt{k}\|\boldsymbol{u}\|_2 . \tag{6.55}$$

One will see, that

$$\underline{u_i} \leq \|\underline{\boldsymbol{u}}\|_\infty = \|\boldsymbol{u}\|_\infty , \tag{6.56}$$

where both sides of the inequality are non-negative. Let $\Lambda = \operatorname{supp}\boldsymbol{u}$, then

$$\|\boldsymbol{u}\|_2^2 = \|\underline{\boldsymbol{u}}\|_2^2 = \sum_{i\in\Lambda} (\underline{u_i})^2 \leq \|\boldsymbol{u}\|_\infty^2 \sum_{i\in\Lambda} 1 = k\|\boldsymbol{u}\|_\infty^2 . \tag{6.57}$$

Taking the square-root leads to the desired inequality.

■

Lemma 10 *For $\boldsymbol{u}, \boldsymbol{v} \in \mathcal{U}$ with disjoint support we find*

$$\|\boldsymbol{u}\|_2 + \|\boldsymbol{v}\|_2 \leq \sqrt{2}\|\boldsymbol{u}+\boldsymbol{v}\|_2 . \tag{6.58}$$

Proof Consider the $\mathbb{R}^2$ vector

$$\boldsymbol{x} := \begin{pmatrix} \|\boldsymbol{u}\|_2 \\ \|\boldsymbol{v}\|_2 \end{pmatrix} . \tag{6.59}$$

Lemma 9 holds also for constant vectors and by construction $\boldsymbol{x}$ is $k = 2$ sparse, thus

$$\|\boldsymbol{x}\|_1 \leq \sqrt{2}\|\boldsymbol{x}\|_2 \,. \tag{6.60}$$

We can now replace

$$\|\boldsymbol{x}\|_1 = \|\boldsymbol{u}\|_2 + \|\boldsymbol{v}\|_2 \,. \tag{6.61}$$

For the right hand side we find

$$\|\boldsymbol{x}\|_2^2 = \|\boldsymbol{u}\|_2^2 + \|\boldsymbol{v}\|_2^2 = \|\boldsymbol{u} + \boldsymbol{v}\|_2^2 \tag{6.62}$$

where the second equality comes from proposition 11. Combining the latter equations leads to the desired inequality.

■

Lemma 11 *Let $\boldsymbol{u} \in \mathscr{U}$ an Λ_0 an index set of cardinality k. For better readability we write $\boldsymbol{x} := \underline{\boldsymbol{u}_{\Lambda_0^c}}$. Note, that $\boldsymbol{x}$ is a $\mathbb{R}^N$ vector with non-negative components. Let $L = (l_1, \dots, l_N)$ a list of indices with $l_i \neq l_j$ for $i \neq j$ such that for the components of $\boldsymbol{x}$ we find*

$$x_{l_1} \geq x_{l_2} \geq \ldots \geq x_{l_N} \,. \tag{6.63}$$

Define index sets $\Lambda_1 := \{l_1, \dots, l_k\}$, $\Lambda_2 := \{l_{k+1}, \dots, l_{2k}\}$ and so forth until the whole list L is covered. If k is not a divisor of N the last index set would have less than k elements. This can be remedied by appending zero elements.

We find

$$\sum_{j \geq 2} \|\boldsymbol{u}_{\Lambda_j}\|_2 \leq \frac{\|\boldsymbol{u}_{\Lambda_0^c}\|_1}{\sqrt{k}} \,. \tag{6.64}$$

Proof First note that by construction all Λ_i for $i = 0, 1, \ldots$ are pair-wise disjoint and that

$$\Lambda_0^c = \Lambda_1 \dot{\cup} \Lambda_2 \dot{\cup} \ldots \tag{6.65}$$

For any $m \in \Lambda_{j-1}$ we get by construction

$$\underline{u_m} \geq \|\boldsymbol{u}_{\Lambda_j}\|_\infty \,. \tag{6.66}$$

We now take the sum over all $m \in \Lambda_{j-1}$

$$\|\boldsymbol{u}_{\Lambda_{j-1}}\|_1 \geq k\|\boldsymbol{u}_{\Lambda_j}\|_\infty \,. \tag{6.67}$$

From lemma 9 we have for each j

$$\|\boldsymbol{u}_{\Lambda_j}\|_2 \leq \sqrt{k}\|\boldsymbol{u}_{\Lambda_j}\|_\infty \tag{6.68}$$

and in combination with the latter inequality

$$\|\boldsymbol{u}_{\Lambda_j}\|_2 \leq \frac{1}{\sqrt{k}}\|\boldsymbol{u}_{\Lambda_{j-1}}\|_1 \,. \tag{6.69}$$

We take the sum over $j \geq 2$

$$\sum_{j\geq 2} \|\boldsymbol{u}_{\Lambda_j}\|_2 \leq \sum_{j\geq 2} \frac{1}{\sqrt{k}} \|\boldsymbol{u}_{\Lambda_{j-1}}\|_1 . \tag{6.70}$$

In the right hand side we first perform an index shift and then use proposition 11 to see that

$$\sum_{j\geq 2} \|\boldsymbol{u}_{\Lambda_{j-1}}\|_1 = \sum_{j\geq 1} \|\boldsymbol{u}_{\Lambda_j}\|_1 = \|\boldsymbol{u}_{\Lambda_1 \cup \Lambda_2 \cup \ldots}\|_1 = \|\boldsymbol{u}_{\Lambda_0^c}\|_1 . \tag{6.71}$$

■

6.1.2 Output Space

In analogy to the input space $\mathscr{U}$, the underline notation can be used for the output space $\mathscr{Y} = \mathscr{Y}_1 \oplus \ldots \oplus \mathscr{Y}_P$,

$$\underline{y_i} := \|y_i\|_{p'} . \tag{6.72}$$

Here, we assume that each $y_i \in \mathscr{Y}_i$ is in $L^{p'}[0,T]$ an piecewise $C^\infty[0,T]$. Principally, p' does not have to match the parameter p from the input spaces. It will be suppressed from out notation as the result are valid for any fixed value of p'. Again it would also be sufficient that $\mathscr{Y}_i$ is a Banach space. The q-norm on $\mathscr{Y}$ is defined as

$$\|\boldsymbol{y}\|_q := \|\underline{\boldsymbol{y}}\|_q . \tag{6.73}$$

Clearly, all rules we have derived for underlined input vectors hold true for underlined output vectors. The following lemma yields a last inequality for underlined vectors.

Lemma 12 *Let $\boldsymbol{y}, \boldsymbol{z} \in \mathscr{Y}$, then*

$$\|\underline{\boldsymbol{y}} - \underline{\boldsymbol{z}}\|_2^2 \leq \|\boldsymbol{y} + \boldsymbol{z}\|_2^2 . \tag{6.74}$$

Proof The inequality can be written as

$$\sum_{i=1}^{P} \left(\underline{y_i} - \underline{z_i}\right)^2 \leq \sum_{i=1}^{P} \left(\underline{y_i + z_i}\right)^2 . \tag{6.75}$$

To prove the validity of the latter inequality it suffices to show that

$$|\underline{y_i} - \underline{z_i}| \leq |\underline{y_i + z_i}| \tag{6.76}$$

for each i in order to complete the proof. First, consider the case $\underline{y_i} \geq \underline{z_i}$. From lemma 7 we get

$$\underline{y_i} = \underline{(y_i + z_i) - z_i} \leq \underline{y_i + z_i} + \underline{z_i} \tag{6.77}$$

and subtracting $\underline{z_i}$ on both sides yields

$$\underline{y_i} - \underline{z_i} \leq \underline{y_i + z_i} . \tag{6.78}$$

Both sides are positive so (6.76) holds. Second, consider the case $\underline{z_i} \geq \underline{y_i}$ and perform the same steps with z_i and y_i swapped to get

$$\underline{z_i} - \underline{y_i} \leq \underline{z_i + y_i}\,. \tag{6.79}$$

For the right hand side it is clear that

$$\underline{z_i + y_i} = \underline{y_i + z_i} \tag{6.80}$$

and for the left hand side

$$\underline{z_i} - \underline{y_i} = |\underline{y_i} - \underline{z_i}| \tag{6.81}$$

thus equation (6.76) holds also in this case.

■

6.2 Convex Optimisation

Proposition 12 *If Φ is linear, then P1 is a convex optimisation problem.*

Proof We first show that the constraint set

$$\mathscr{A} := \{\boldsymbol{u} \in \mathscr{U} \mid \|\Phi(\boldsymbol{u}) - \boldsymbol{y}\|_2 \leq \epsilon\} \tag{6.82}$$

with $\boldsymbol{y} \in \mathscr{Y}$ and $\epsilon > 0$ is convex. Let $\alpha, \beta > 0$ such that $\alpha + \beta = 1$. We have to show that

$$\boldsymbol{u}, \boldsymbol{v} \in \mathscr{A} \Rightarrow (\alpha\boldsymbol{u} + \beta\boldsymbol{v}) \in \mathscr{A}\,. \tag{6.83}$$

Given that Φ is linear we can estimate

$$\|\Phi(\alpha\boldsymbol{u} + \beta\boldsymbol{u}) - \boldsymbol{y}\|_2 = \|\alpha\Phi(\boldsymbol{u}) - \alpha\boldsymbol{y} + \beta\Phi(\boldsymbol{v}) - \beta\boldsymbol{y}\|_2 \tag{6.84}$$

and with the triangle inequality

$$\|\Phi(\alpha\boldsymbol{u} + \beta\boldsymbol{u}) - \boldsymbol{y}\|_2 \leq \alpha\|\Phi(\boldsymbol{u}) - \boldsymbol{y}\|_2 + \beta\|\Phi(\boldsymbol{v}) - \boldsymbol{y}\|_2 \leq \alpha\epsilon + \beta\epsilon = \epsilon\,. \tag{6.85}$$

Since $\|\cdot\|_q$ is a proper norm on $\mathscr{U}$ we again apply the triangle inequality

$$\|\alpha\boldsymbol{u} + \beta\boldsymbol{v}\|_1 \leq \alpha\|\boldsymbol{u}\|_1 + \beta\|\boldsymbol{v}\|_1 \tag{6.86}$$

to see that the function we want to minimize is convex.

■

6.3 Restricted Isometry Property

The *Restricted Isometry Property* first appeared in [61] and was also considered under the name ϵ-isometry in [137]. In the following we will consider *linear* dynamic system.

Definition 29 *The Restricted Isometry Property of order $2k$ (RIP$2k$) is fulfilled if there is a constant*

δ_{2k} such that for $\boldsymbol{u},\boldsymbol{v}\in\Sigma_k$ the inequalities

$$(1-\delta_{2k})\|\underline{\boldsymbol{u}}-\underline{\boldsymbol{v}}\|_2^2 \le \|\underline{\Phi(\boldsymbol{u})}-\underline{\Phi(\boldsymbol{v})}\|_2^2 \tag{6.87}$$

and

$$\|\underline{\Phi(\boldsymbol{u})}+\underline{\Phi(\boldsymbol{v})}\|_2^2 \le (1+\delta_{2k})\|\underline{\boldsymbol{u}}+\underline{\boldsymbol{v}}\|_2^2 \tag{6.88}$$

hold.

The following lemma was proven for matrices in [116] and can now be generalised to linear dynamic systems.

Lemma 13 *Assume the RIP$2k$ holds and let $\boldsymbol{u},\boldsymbol{v}$ with disjoint support. Then*

$$\langle\underline{\Phi(\boldsymbol{u})},\underline{\Phi(\boldsymbol{v})}\rangle \le \delta_{2k}\|\boldsymbol{u}\|_2\|\boldsymbol{v}\|_2\,. \tag{6.89}$$

Proof First we divide (6.89) by $\|\boldsymbol{u}\|_2\|\boldsymbol{v}\|_2$ to get a normalised version of (6.89)

$$\frac{\langle\underline{\Phi(\boldsymbol{u})},\underline{\Phi(\boldsymbol{v})}\rangle}{\|\boldsymbol{u}\|_2\|\boldsymbol{v}\|_2} = \left\langle \underline{\Phi\left(\frac{\boldsymbol{u}}{\|\boldsymbol{u}\|_2}\right)}, \underline{\Phi\left(\frac{\boldsymbol{v}}{\|\boldsymbol{v}\|_2}\right)}\right\rangle \le \delta_{2k} \tag{6.90}$$

where we used the homogeneity of $\underline{\cdot}$ and the linearity of Φ. Underlined vectors are simply vectors in $\mathbb{R}^P$, hence the scalar product is the standard scalar product. For simplicity of notation we henceforth assume $\|\boldsymbol{u}\|_2=\|\boldsymbol{v}\|_2=1$. We apply the parallelogram identity to the left hand side of (6.89) to get

$$\langle\underline{\Phi(\boldsymbol{u})},\underline{\Phi(\boldsymbol{v})}\rangle = \frac{1}{4}\left(\|\underline{\Phi(\boldsymbol{u})}+\underline{\Phi(\boldsymbol{v})}\|_2^2 - \|\underline{\Phi(\boldsymbol{u})}-\underline{\Phi(\boldsymbol{v})}\|_2^2\right) \tag{6.91}$$

and since RIP$2k$ holds we get

$$\langle\underline{\Phi(\boldsymbol{u})},\underline{\Phi(\boldsymbol{v})}\rangle = \frac{1}{4}\left((1+\delta_{2k})\|\underline{\boldsymbol{u}}+\underline{\boldsymbol{v}}\|_2^2 - (1-\delta_{2k})\|\underline{\boldsymbol{u}}-\underline{\boldsymbol{v}}\|_2^2\right). \tag{6.92}$$

We apply lemma 8 to see that due to the disjoint support we get

$$\|\underline{\boldsymbol{u}}+\underline{\boldsymbol{v}}\|_2^2 = \|\underline{\boldsymbol{u}}-\underline{\boldsymbol{v}}\|_2^2 = \|\boldsymbol{u}+\boldsymbol{v}\|_2^2 \tag{6.93}$$

and proposition 11 yields

$$\langle\underline{\Phi(\boldsymbol{u})},\underline{\Phi(\boldsymbol{v})}\rangle \le 2\delta_{2k}\left(\|\boldsymbol{u}\|_2^2+\|\boldsymbol{v}\|_2^2\right) = \delta_{2k}\,. \tag{6.94}$$

■

To see that our framework is in agreement with the results for the static problem consider the following: Let $A\in\mathbb{R}^{P\times N}$ and $\boldsymbol{y}\in\mathbb{R}^P$ be given and $\boldsymbol{w}\in\mathbb{R}^N$. Solve

$$A\boldsymbol{w}=\boldsymbol{y} \tag{6.95}$$

for $\boldsymbol{w}$. This problem can be seen as a dynamic system with trivial time-development and input-output map A. Consequently an input set $\mathsf{S}=\{\mathsf{s}_1,\mathsf{s}_2,\dots\}$ is independent in the gammoid if and only if the columns $\{\boldsymbol{a}_{s_1}, \boldsymbol{a}_{s_2}, \ldots\}$ of A are linearly independent. In this special case we can drop the underline notation and we can rewrite the RIP2k condition as follows. Let $\kappa := 2k$. For each $\boldsymbol{u}, \boldsymbol{v} \in \Sigma_k$

$$(1 - \delta_{2k}) \|\boldsymbol{u} - \boldsymbol{v}\|_2^2 \leq \|A\boldsymbol{u} - A\boldsymbol{v}\|_2^2 \tag{6.96}$$

is equivalent to saying that for each $\boldsymbol{w} := \boldsymbol{u} - \boldsymbol{v}$ in $\Sigma_{2k} = \Sigma_\kappa$ we have

$$(1 - \delta_\kappa) \|\boldsymbol{w}\|_2^2 \leq \|A\boldsymbol{w}\|_2^2 . \tag{6.97}$$

The same argumentation holds for the second inequality, so that we can say, the system has the RIP of order κ if and only if for all $\boldsymbol{w} \in \Sigma_\kappa$

$$(1 - \delta_\kappa) \|\boldsymbol{w}\|_2^2 \leq \|A\boldsymbol{w}\|_2^2 \leq (1 + \delta_\kappa) \|\boldsymbol{w}\|_2^2 . \tag{6.98}$$

The latter is exactly the RIP condition that one usually formulates for static compressed sensing.

We now turn our interest to one important proposition about systems that fulfil the RIP2k. This one corresponds to Lemma 1.3 from [116]. We follow the idea of the proof given there, however, some additional steps are necessary in order to get a result valid for dynamic systems.

Proposition 13 *Assume the RIP2k holds and let $\boldsymbol{u} \in \mathscr{U}$ and $k \in \mathbb{N}$. Let Λ_0 be an index set of size* $\operatorname{card} \Lambda_0 \leq k$. *Let Λ_1 correspond to the k largest entries of $\underline{\boldsymbol{u}_{\Lambda_0^c}}$ (see lemma 11), Λ_2 to the second largest and so forth. We set $\Lambda := \Lambda_0 \cup \Lambda_1$ and*

$$\alpha := \frac{\sqrt{2}\delta_{2k}}{1 - \delta_{2k}} \quad , \quad \beta := \frac{1}{1 - \delta_{2k}} . \tag{6.99}$$

Then

$$\|\boldsymbol{u}_\Lambda\|_2 \leq \alpha \frac{\|\boldsymbol{u}_{\Lambda_0^c}\|_1}{\sqrt{k}} + \beta \frac{\left\langle \underline{\Phi(\boldsymbol{u}_\Lambda)}, \underline{\Phi(\boldsymbol{u})} \right\rangle}{\|\boldsymbol{u}_\Lambda\|_2} . \tag{6.100}$$

Proof For two complementary sets Λ and Λ^c we have $\Lambda \cap \Lambda^c = \emptyset$ so that we can use equation (6.41). Furthermore, $\Lambda \cup \Lambda^c = \{1, \ldots, N\}$ shows that $\boldsymbol{u} = \boldsymbol{u}_\Lambda + \boldsymbol{u}_{\Lambda^c}$. Thus due to linearity of Φ we get

$$\Phi(\boldsymbol{u}_\Lambda) = \Phi(\boldsymbol{u}) - \Phi(\boldsymbol{u}_{\Lambda^c}) . \tag{6.101}$$

By construction $\Lambda^c = \Lambda_2 \dot{\cup} \Lambda_3 \dot{\cup} \ldots$ so we can substitute the latter term by a sum

$$\Phi(\boldsymbol{u}_\Lambda) = \Phi(\boldsymbol{u}) - \sum_{j \geq 2} \Phi(\boldsymbol{u}_{\Lambda_j}) . \tag{6.102}$$

By construction we also know that $\boldsymbol{u}_{\Lambda_0}, \boldsymbol{u}_{\Lambda_1} \in \Sigma_k$, so from the RIP2$k$ we get

$$(1 - \delta_{2k}) \|\underline{\boldsymbol{u}_{\Lambda_0}} - \underline{\boldsymbol{u}_{\Lambda_1}}\|_2^2 \leq \|\underline{\Phi(\boldsymbol{u}_{\Lambda_0})} - \underline{\Phi(\boldsymbol{u}_{\Lambda_1})}\|_2^2 \tag{6.103}$$

On the left hand side we notice that Λ_0 and Λ_1 are disjoint so we use lemma 8 to see that

$$\|\underline{\boldsymbol{u}_{\Lambda_0}} - \underline{\boldsymbol{u}_{\Lambda_1}}\|_2^2 = \|\boldsymbol{u}_{\Lambda_0} + \boldsymbol{u}_{\Lambda_1}\|_2^2 = \|\boldsymbol{u}_\Lambda\|_2^2 . \tag{6.104}$$

On the right hand side we use lemma 12 and the linearity of Φ to get

$$\|\underline{\Phi(\boldsymbol{u}_{\Lambda_0})} - \underline{\Phi(\boldsymbol{u}_{\Lambda_1})}\|_2^2 \leq \|\Phi(\boldsymbol{u}_{\Lambda_0}) + \Phi(\boldsymbol{u}_{\Lambda_1})\|_2^2 = \|\Phi(\boldsymbol{u}_{\Lambda})\|_2^2. \tag{6.105}$$

We can estimate $\|\Phi(\boldsymbol{u}_{\Lambda})\|_2^2$ by

$$\|\Phi(\boldsymbol{u}_{\Lambda})\|_2^2 \leq \left\langle \underline{\Phi(\boldsymbol{u}_{\Lambda})}, \underline{\Phi(\boldsymbol{u})} \right\rangle + \left\langle \underline{\Phi(\boldsymbol{u}_{\Lambda})}, \sum_{j\geq 2} \underline{\Phi(\boldsymbol{u}_{\Lambda_j})} \right\rangle. \tag{6.106}$$

To see that, we first write the norm as a scalar product

$$\|\Phi(\boldsymbol{u}_{\Lambda})\|_2^2 = \left\langle \underline{\Phi(\boldsymbol{u}_{\Lambda})}, \underline{\Phi(\boldsymbol{u}) - \sum_{j\geq 2} \Phi(\boldsymbol{u}_{\Lambda_j})} \right\rangle \tag{6.107}$$

where the second vector comes from equation (6.102). The triangle inequality from lemma 7 yields

$$\underline{\Phi_i(\boldsymbol{u}) - \sum_{j\geq 2} \Phi_i(\boldsymbol{u}_{\Lambda_j})} \leq \underline{\Phi_i(\boldsymbol{u})} + \underline{\sum_{j\geq 2} \Phi_i(\boldsymbol{u}_{\Lambda_j})} \tag{6.108}$$

as well as

$$\underline{\sum_{j\geq 2} \Phi_i(\boldsymbol{u}_{\Lambda_j})} \leq \sum_{j\geq 2} \underline{\Phi_i(\boldsymbol{u}_{\Lambda_j})} \tag{6.109}$$

for each component $i = 1, \ldots, P$. We proceed with the second scalar product in equation (6.106)

$$\left\langle \underline{\Phi(\boldsymbol{u}_{\Lambda})}, \sum_{j\geq 2} \underline{\Phi(\boldsymbol{u}_{\Lambda_j})} \right\rangle = \left\langle \underline{\Phi(\boldsymbol{u}_{\Lambda_0})}, \sum_{j\geq 2} \underline{\Phi(\boldsymbol{u}_{\Lambda_j})} \right\rangle + \left\langle \underline{\Phi(\boldsymbol{u}_{\Lambda_1})}, \sum_{j\geq 2} \underline{\Phi(\boldsymbol{u}_{\Lambda_j})} \right\rangle \tag{6.110}$$

where we again used the linearity and triangle inequality in each component,

$$\underline{\Phi_i(\boldsymbol{u}_{\Lambda})} = \underline{\Phi_i(\boldsymbol{u}_{\Lambda_0}) + \Phi_i(\boldsymbol{u}_{\Lambda_1})} \leq \underline{\Phi_i(\boldsymbol{u}_{\Lambda_0})} + \underline{\Phi_i(\boldsymbol{u}_{\Lambda_1})}. \tag{6.111}$$

For $m = 0, 1$ we first write the sum outside of the scalar product and apply lemma 13

$$\left\langle \underline{\Phi(\boldsymbol{u}_{\Lambda_m})}, \sum_{j\geq 2} \underline{\Phi(\boldsymbol{u}_{\Lambda_j})} \right\rangle = \sum_{j\geq 2} \left\langle \underline{\Phi(\boldsymbol{u}_{\Lambda_m})}, \underline{\Phi(\boldsymbol{u}_{\Lambda_j})} \right\rangle \leq \sum_{j\geq 2} \delta_{2k} \|\boldsymbol{u}_{\Lambda_m}\|_2 \|\boldsymbol{u}_{\Lambda_j}\|_2 \tag{6.112}$$

and then lemma 11 to get

$$\left\langle \underline{\Phi(\boldsymbol{u}_{\Lambda_m})}, \sum_{j\geq 2} \underline{\Phi(\boldsymbol{u}_{\Lambda_j})} \right\rangle \leq \delta_{2k} \|\boldsymbol{u}_{\Lambda_m}\|_2 \frac{\|\boldsymbol{u}_{\Lambda_0^c}\|_1}{\sqrt{k}}. \tag{6.113}$$

We add the two inequalities for $m = 0$ and $m = 1$ to get

$$\left\langle \Phi(\boldsymbol{u}_{\Lambda}), \sum_{j\geq 2} \Phi(\boldsymbol{u}_{\Lambda_j}) \right\rangle \leq \delta_{2k}(\|\boldsymbol{u}_{\Lambda_0}\|_2 + \|\boldsymbol{u}_{\Lambda_1}\|_2) \frac{\|\boldsymbol{u}_{\Lambda_0^c}\|_1}{\sqrt{k}}. \tag{6.114}$$

From lemma 10 we know that

$$\|\boldsymbol{u}_{\Lambda_0}\|_2 + \|\boldsymbol{u}_{\Lambda_1}\|_2 \leq \sqrt{2}\,\|\boldsymbol{u}_{\Lambda_0} + \boldsymbol{u}_{\Lambda_1}\|_2 = \sqrt{2}\|\boldsymbol{u}_{\Lambda}\|_2 . \tag{6.115}$$

We combine all these results to get

$$(1-\delta_{2k})\|\boldsymbol{u}_{\Lambda}\|_2^2 \leq \langle \underline{\Phi(\boldsymbol{u}_{\Lambda})}, \underline{\Phi(\boldsymbol{u})}\rangle + \delta_{2k}\,\sqrt{2}\|\boldsymbol{u}_{\Lambda}\|_2 \frac{\|\boldsymbol{u}_{\Lambda_0^c}\|_1}{\sqrt{k}} . \tag{6.116}$$

For $\delta_{2k} < 1$ and with α and β as defined above we can divide the inequality by $(1-\delta)\|\boldsymbol{u}_{\Lambda}\|_2$ to get the desired inequality

$$\|\boldsymbol{u}_{\Lambda}\|_2 \leq \alpha \frac{\|\boldsymbol{u}_{\Lambda_0^c}\|_1}{\sqrt{k}} + \beta \frac{\langle \underline{\Phi(\boldsymbol{u}_{\Lambda})}, \underline{\Phi(\boldsymbol{u})}\rangle}{\|\boldsymbol{u}_{\Lambda}\|_2} . \tag{6.117}$$

■

Best Sparse Approximation We want to make use of the latter proposition in the following way: Say $\boldsymbol{v} \in \mathscr{U}$ is a fixed vector, e.g., the true model error that we want to reconstruct, and $\boldsymbol{u}$ is our estimate for the model error, e.g., obtained by some optimisation procedure. To measure how good the optimisation procedure performs, we compute the difference $\boldsymbol{w} := \boldsymbol{u} - \boldsymbol{v}$ and utilise the proposition to get an upper bound for $\|\boldsymbol{w}\|_2$

Now assume the RIP2k holds with a constant $\delta_{2k} < \sqrt{2} - 1$. For the proposition we can choose an arbitrary index set Λ_0 of size k. We choose Λ_0 such that it corresponds to the k components of $\underline{\boldsymbol{v}}$ with highest magnitude. As said in the proposition, Λ_1 will now correspond to the k largest components in $\underline{\boldsymbol{w}_{\Lambda_0^c}}$, Λ_2 to the second largest an so forth and we set $\Lambda = \Lambda_0 \cup \Lambda_1$.

Remember that $\sigma_k(\boldsymbol{v})_q$ is the distance between $\boldsymbol{v}$ and the the best k-sparse approximation [118]

$$\sigma_k(\boldsymbol{v})_q := \min_{\tilde{\boldsymbol{v}} \in \Sigma_k} \|\tilde{\boldsymbol{v}} - \boldsymbol{v}\|_q \tag{6.118}$$

and note that

$$\sigma_k(\boldsymbol{v})_1 = \|\boldsymbol{v}_{\Lambda_0} - \boldsymbol{v}\|_1 . \tag{6.119}$$

To see the latter equation, let $\tilde{\boldsymbol{v}} \in \Sigma_k$ such that $\|\tilde{\boldsymbol{v}} - \boldsymbol{v}\|_1$ is minimal. Since $\tilde{\boldsymbol{v}}$ is k-sparse, there is a $\tilde{\Lambda}$ such that $\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}_{\tilde{\Lambda}}$. Written as a sum

$$\|\tilde{\boldsymbol{v}} - \boldsymbol{v}\|_1 = \sum_{i \in \tilde{\Lambda}} \underline{\tilde{v}_i - v_i} + \sum_{i \in \tilde{\Lambda}^c} \underline{v_i} . \tag{6.120}$$

The non-zero components must be chosen $\tilde{\boldsymbol{v}}_i = \boldsymbol{v}_i$ since this makes the first sum vanish. In order to minimise the second sum the index set $\tilde{\Lambda}^c$ must correspond to the smallest $\underline{v_i}$, in other words, $\tilde{\Lambda}$ corresponds to the largest $\underline{v_i}$, thus $\tilde{\boldsymbol{v}} = \boldsymbol{v}_{\Lambda_0}$. In the remainder of this section we follow the argumentation line of [116] where the static problem in $\mathbb{R}^N$ was considered. Therein, also results are utilised which were first presented in [138]. We show, how a line of reasoning can be made for the general case of composite Banach spaces.

Lemma 14 *Consider the setting explained above and assume* $\|\boldsymbol{u}\|_1 \le \|\boldsymbol{v}\|_1$. *Then*

$$\|\boldsymbol{w}\|_2 \le 2\|\boldsymbol{w}_\Lambda\|_2 + 2\frac{\sigma_k(\boldsymbol{v})_1}{\sqrt{k}}. \tag{6.121}$$

Proof We begin with splitting $\boldsymbol{w}$ and applying the triangle inequality

$$\|\boldsymbol{w}\|_2 = \|\boldsymbol{w}_\Lambda + \boldsymbol{w}_{\Lambda^c}\|_2 \le \|\boldsymbol{w}_\Lambda\|_2 + \|\boldsymbol{w}_{\Lambda^c}\|_2. \tag{6.122}$$

Since $\Lambda^c = \Lambda_2 \dot\cup \Lambda_3 \dot\cup \ldots$ we can apply the triangle inequality and lemma 11

$$\|\boldsymbol{w}_{\Lambda^c}\|_2 = \left\| \sum_{j\ge 2} \boldsymbol{w}_{\Lambda_j} \right\|_2 \le \sum_{j\ge 2} \|\boldsymbol{w}_{\Lambda_j}\|_2 \le \frac{\|\boldsymbol{w}_{\Lambda_0^c}\|_1}{\sqrt{k}}. \tag{6.123}$$

By construction $\boldsymbol{u} = \boldsymbol{v} + \boldsymbol{w}$ so it is clear that

$$\|\boldsymbol{v} + \boldsymbol{w}\|_1 \le \|\boldsymbol{v}\|_1. \tag{6.124}$$

For the complementary sets Λ_0 and Λ_0^c we can apply proposition 11 with $q = 1$ to get

$$\|\boldsymbol{v} + \boldsymbol{w}\|_1 = \|\boldsymbol{v}_{\Lambda_0} + \boldsymbol{w}_{\Lambda_0}\|_1 + \|\boldsymbol{w}_{\Lambda_0^c} + \boldsymbol{v}_{\Lambda_0^c}\|_1. \tag{6.125}$$

In the proof of lemma 12 we know for each component that

$$||\underline{v_i}| - |\underline{w_i}|| \le |\underline{v_i + w_i}|. \tag{6.126}$$

So we estimate the norm

$$\begin{aligned}\|\boldsymbol{v}_{\Lambda_0} + \boldsymbol{w}_{\Lambda_0}\|_1 &= \sum_{i\in\Lambda_0} |\underline{v_i + w_i}| \ge \sum_{i\in\Lambda_0} ||\underline{v_i}| - |\underline{w_i}|| \\ &\ge \left| \sum_{i\in\Lambda_0} (|\underline{v_i}| - |\underline{w_i}|) \right| = |(\|\boldsymbol{v}_{\Lambda_0}\|_1 - \|\boldsymbol{w}_{\Lambda_0}\|_1)| \ge \|\boldsymbol{v}_{\Lambda_0}\|_1 - \|\boldsymbol{w}_{\Lambda_0}\|_1\end{aligned} \tag{6.127}$$

and the same holds for Λ_0^c. With this result equation (6.125) becomes

$$\|\boldsymbol{v}_{\Lambda_0}\|_1 - \|\boldsymbol{w}_{\Lambda_0}\|_1 + \|\boldsymbol{w}_{\Lambda_0^c}\|_1 - \|\boldsymbol{v}_{\Lambda_0^c}\|_1 \le \|\boldsymbol{v}\|_1 \tag{6.128}$$

which yields a lower bound for $\|\boldsymbol{w}_{\Lambda_0^c}\|_1$

$$\|\boldsymbol{w}_{\Lambda_0^c}\|_1 \le \|\boldsymbol{v}\|_1 - \|\boldsymbol{v}_{\Lambda_0}\|_1 + \|\boldsymbol{w}_{\Lambda_0}\|_1 + \|\boldsymbol{v}_{\Lambda_0^c}\|_1. \tag{6.129}$$

A calculation as in equation (6.127) together with equation (6.119) shows that

$$\|\boldsymbol{v}\|_1 - \|\boldsymbol{v}_{\Lambda_0}\|_1 \le \|\boldsymbol{v} - \boldsymbol{v}_{\Lambda_0}\|_1 = \sigma_k(\boldsymbol{v})_1 \tag{6.130}$$

and since Λ_0^c is complementary to Λ_0, $\boldsymbol{v} = \boldsymbol{v}_{\Lambda_0} + \boldsymbol{v}_{\Lambda_0^c}$, thus

$$\|\boldsymbol{v}_{\Lambda_0^c}\|_1 = \|\boldsymbol{v} - \boldsymbol{v}_{\Lambda_0}\|_1 = \sigma_k(\boldsymbol{v})_1. \tag{6.131}$$

Combining the latter three results we get

$$\|\boldsymbol{w}_{\Lambda_0^c}\|_1 \leq \|\boldsymbol{w}_{\Lambda_0}\|_1 + 2\sigma_k(\boldsymbol{v})_1 . \tag{6.132}$$

Insertion into equation (6.123) yields

$$\|\boldsymbol{w}_{\Lambda^c}\|_2 \leq \frac{\|\boldsymbol{w}_{\Lambda_0}\|_1 + 2\sigma_k(\boldsymbol{v})_1}{\sqrt{k}} . \tag{6.133}$$

From lemma 9 we get

$$\|\boldsymbol{w}_{\Lambda^c}\|_2 \leq \|\boldsymbol{w}_{\Lambda_0}\|_2 + 2\frac{\sigma_k(\boldsymbol{v})_1}{\sqrt{k}} . \tag{6.134}$$

We now use the triangle inequality

$$\|\boldsymbol{w}\|_2 \leq \|\boldsymbol{w}_{\Lambda}\|_2 + \|\boldsymbol{w}_{\Lambda^c}\|_2 \tag{6.135}$$

and the fact, that $\Lambda_0 \subseteq \Lambda$, thus

$$\|\boldsymbol{w}_{\Lambda_0}\|_2 \leq \|\boldsymbol{w}_{\Lambda}\|_2 \tag{6.136}$$

to get the desired inequality

$$\|\boldsymbol{w}\| \leq 2\|\boldsymbol{w}_{\Lambda}\|_2 + 2\frac{\sigma_k(\boldsymbol{v})_1}{\sqrt{k}} . \tag{6.137}$$

■

The following theorem follows from a combination of proposition 13 and lemma 14. It is the key result for the optimisation problem we present for linear dynamic input-output systems.

Theorem 11 *Let $\boldsymbol{u}, \boldsymbol{v} \in \mathcal{U}$ with $\|\boldsymbol{u}\|_1 \leq \|\boldsymbol{v}\|_1$ and assume for Φ we have the RIP$2k$ with $\delta_{2k} < \sqrt{2} - 1$. Let Λ_0 correspond to the k largest components of $\underline{v}$. We set $\boldsymbol{w} := \boldsymbol{u} - \boldsymbol{v}$ and let Λ_1 correspond to the k largest components of $\underline{\boldsymbol{w}_{\Lambda_0^c}}$, Λ_2 to the second largest and so forth. Let $\Lambda := \Lambda_0 \cup \Lambda_1$. There are two constants C_0 and C_1 such that*

$$\|\boldsymbol{w}\|_2 \leq C_0\frac{\sigma_k(\boldsymbol{v})_1}{\sqrt{k}} + C_1\frac{\langle\underline{\Phi(\boldsymbol{w}_{\Lambda})}, \underline{\Phi(\boldsymbol{w})}\rangle}{\|\boldsymbol{w}_{\Lambda}\|_2} . \tag{6.138}$$

Proof We start with proposition 13 applied to $\boldsymbol{w}$

$$\|\boldsymbol{w}_{\Lambda}\|_2 \leq \alpha\frac{\|\boldsymbol{w}_{\Lambda_0^c}\|_1}{\sqrt{k}} + \beta\frac{\langle\underline{\Phi(\boldsymbol{w}_{\Lambda})}, \underline{\Phi(\boldsymbol{w})}\rangle}{\|\boldsymbol{w}_{\Lambda}\|_2} . \tag{6.139}$$

In lemma 14 equation 6.134 we found an estimate of $\|\boldsymbol{w}_{\Lambda_0^c}\|_1$, inserted into the latter inequality

$$\|\boldsymbol{w}_{\Lambda}\|_2 \leq \alpha\frac{\|\boldsymbol{w}_{\Lambda_0}\|_1}{\sqrt{k}} + 2\alpha\frac{\sigma_k(\boldsymbol{v})_1}{\sqrt{k}} + \beta\frac{\langle\underline{\Phi(\boldsymbol{w}_{\Lambda})}, \underline{\Phi(\boldsymbol{w})}\rangle}{\|\boldsymbol{w}_{\Lambda}\|_2} . \tag{6.140}$$

The first term can be treated with lemma 9 and then we use the fact that $\Lambda_0 \subseteq \Lambda$, thus $\|\boldsymbol{w}_{\Lambda_0}\|_2 \leq$

$\|\boldsymbol{w}_\Lambda\|_2$, to get

$$\|\boldsymbol{w}_\Lambda\|_2 \leq \alpha\|\boldsymbol{w}_\Lambda\|_2 + 2\alpha\frac{\sigma_k(\boldsymbol{v})_1}{\sqrt{k}} + \beta\frac{\langle\Phi(\boldsymbol{w}_\Lambda),\Phi(\boldsymbol{w})\rangle}{\|\boldsymbol{w}_\Lambda\|_2}. \tag{6.141}$$

Due to the assumption $\delta_{2k} < \sqrt{2}-1$ we also get $\alpha < \sqrt{2}-1 < 1$ hence $(1-\alpha)$ is positive so we rewrite the latter equation as

$$\|\boldsymbol{w}_\Lambda\|_2 \leq \frac{2\alpha}{1-\alpha}\frac{\sigma_k(\boldsymbol{v})_1}{\sqrt{k}} + \frac{\beta}{1-\alpha}\frac{\langle\Phi(\boldsymbol{w}_\Lambda),\Phi(\boldsymbol{w})\rangle}{\|\boldsymbol{w}_\Lambda\|_2}. \tag{6.142}$$

To close the proof we use lemma 14 to get

$$\frac{1}{2}\left(\|\boldsymbol{w}\| - 2\frac{\sigma_k(\boldsymbol{v})_1}{\sqrt{k}}\right) \leq \frac{2\alpha}{1-\alpha}\frac{\sigma_k(\boldsymbol{v})_1}{\sqrt{k}} + \frac{\beta}{1-\alpha}\frac{\langle\Phi(\boldsymbol{w}_\Lambda),\Phi(\boldsymbol{w})\rangle}{\|\boldsymbol{w}_\Lambda\|_2} \tag{6.143}$$

and with

$$C_0 := \left(\frac{4\alpha}{1-\alpha} + 2\right) \tag{6.144}$$

and

$$C_1 := \frac{2\beta}{1-\alpha} \tag{6.145}$$

we finally obtain

$$\|\boldsymbol{w}\|_2 \leq C_0\frac{\sigma_k(\boldsymbol{v})_1}{\sqrt{k}} + C_1\frac{\langle\Phi(\boldsymbol{w}_\Lambda),\Phi(\boldsymbol{w})\rangle}{\|\boldsymbol{w}_\Lambda\|_2}. \tag{6.146}$$

■

Application to the MERP We can now apply theorem 11 to the solutions of P0 and P1. Assume $\boldsymbol{y}^{\text{data}} \in \mathscr{Y}$ is given data which is produced by a sparse *true* input $\boldsymbol{w}^* \in \mathscr{U}$, i.e.,

$$\Phi(\boldsymbol{w}^*) = \boldsymbol{y}^{\text{data}}. \tag{6.147}$$

We want to infer $\boldsymbol{w}^*$ from $\boldsymbol{y}^{\text{data}}$. Sparsity of the true input $\boldsymbol{w}^*$ means that for all $\boldsymbol{u}$ in

$$\mathscr{A} := \left\{\boldsymbol{u} \in \mathscr{U} \mid \|\Phi(\boldsymbol{u}) - \boldsymbol{y}\|_2 = 0\right\} \tag{6.148}$$

we find

$$\|\boldsymbol{w}^*\|_0 \leq \|\boldsymbol{u}\|_0. \tag{6.149}$$

Now let $\hat{\boldsymbol{w}}$ be a solution of the convex P1 optimisation problem, that is, for all $\boldsymbol{u} \in \mathscr{A}$ we find

$$\|\hat{\boldsymbol{w}}\|_1 \leq \|\boldsymbol{u}\|_1. \tag{6.150}$$

We can now apply theorem 11 to $\boldsymbol{w} = \hat{\boldsymbol{w}} - \boldsymbol{w}^*$. Note, that

$$\Phi(\hat{\boldsymbol{w}}) = \Phi(\boldsymbol{w}^*) \Rightarrow \Phi(\boldsymbol{w}) = 0. \tag{6.151}$$

We find

$$\|\hat{\boldsymbol{w}} - \boldsymbol{w}^*\|_2 \leq C_0 \frac{\sigma_k(\boldsymbol{w}^*)_1}{\sqrt{k}}. \tag{6.152}$$

Note that by definition $\sigma_k(\boldsymbol{w}^*)_1$ is the best k-sparse approximation to $\boldsymbol{w}^*$ in 1-norm. Thus we have shown that the convex $\|\cdot\|_1$ optimisation yields an approximation of the unknown, true input.

Theorem 11 might be adjusted for various scenarios where we can make further assumptions about the model error $\boldsymbol{w}^*$ or about stochastic or measurement errors. We want to derive a last inequality for the case of bounded noise. Let ξ represent the noise, and $\Phi_{\text{noiseless}}$ the solution operator we have discussed so far. We now consider a new solution operator

$$\Phi(\boldsymbol{u})(t) := \Phi_{\text{noiseless}}(\boldsymbol{u})(t) + \xi(t) \tag{6.153}$$

that incorporates the noise ξ. Since $\xi(t) \in \mathbb{R}^P$ we can interpret $\xi \in \mathscr{Y}$ and use the norm on $\mathscr{Y}$. We assume that ξ is a bounded noise with $\epsilon > 0$,

$$\|\xi\|_2 \leq \epsilon. \tag{6.154}$$

We adjust the solution set

$$\mathscr{A} := \left\{\boldsymbol{u} \in \mathscr{U} \mid \|\Phi(\boldsymbol{u}) - \boldsymbol{y}\|_2 \leq \epsilon\right\} \tag{6.155}$$

and define $\boldsymbol{w}^*, \hat{\boldsymbol{w}} \in \mathscr{A}$ as before with minimal $\|\cdot\|_0$ and $\|\cdot\|_1$ norm, respectively. From the theorem we get

$$\|\boldsymbol{w}\|_2 \leq C_0 \frac{\sigma_k(\boldsymbol{w}^*)_1}{\sqrt{k}} + C_1 \frac{\langle \underline{\Phi(\boldsymbol{w}_\Lambda)}, \underline{\Phi(\boldsymbol{w})} \rangle}{\|\boldsymbol{w}_\Lambda\|_2}. \tag{6.156}$$

We want to estimate the scalar product by the Cauchy-Schwartz inequality

$$\langle \underline{\Phi(\boldsymbol{w}_\Lambda)}, \underline{\Phi(\boldsymbol{w})} \rangle \leq \|\Phi(\boldsymbol{w}_\Lambda)\|_2 \|\Phi(\boldsymbol{w})\|_2. \tag{6.157}$$

By construction $\Lambda = \Lambda_0 \cup \Lambda_1$ has at most $2k$ elements. Thus, it is possible to write $\boldsymbol{w} = \boldsymbol{u} + \boldsymbol{v}$ where $\boldsymbol{u}, \boldsymbol{v} \in \Sigma_k$ have disjoint support. With lemma 7 we get

$$\|\Phi(\boldsymbol{w}_\Lambda)\|_2 \ = \|\underline{\Phi(\boldsymbol{u}) + \Phi(\boldsymbol{v})}\|_2 \leq \|\underline{\Phi(\boldsymbol{u})} + \underline{\Phi(\boldsymbol{v})}\|_2. \tag{6.158}$$

We can now use the RIP2k to get

$$\|\Phi(\boldsymbol{w}_\Lambda)\|_2 \leq \sqrt{1 + \delta_{2k}} \|\underline{\boldsymbol{u}} + \underline{\boldsymbol{v}}\|_2 \tag{6.159}$$

and with lemma 8

$$\|\Phi(\boldsymbol{w}_\Lambda)\|_2 \leq \sqrt{1 + \delta_{2k}} \|\boldsymbol{u} + \boldsymbol{v}\|_2 = \sqrt{1 + \delta_{2k}} \|\boldsymbol{w}_\Lambda\|_2. \tag{6.160}$$

On the other hand we can write $\boldsymbol{w} = \hat{\boldsymbol{w}} - \boldsymbol{w}^*$ and get

$$\|\Phi(\boldsymbol{w})\| = \left\| \left(\Phi(\hat{\boldsymbol{w}}) - \boldsymbol{y}\right) - \left(\Phi(\boldsymbol{w}^*) - \boldsymbol{y}\right) \right\|_2 \leq \left\|\Phi(\hat{\boldsymbol{w}}) - \boldsymbol{y}\right\|_2 + \left\|\Phi(\boldsymbol{w}^*) - \boldsymbol{y}\right\|_2 \leq 2\epsilon. \tag{6.161}$$

With this, equation (6.157) becomes

$$\langle \underline{\Phi(\boldsymbol{w}_\Lambda)}, \underline{\Phi(\boldsymbol{w})} \rangle \leq 2\epsilon \sqrt{1 + \delta_{2k}} \|\boldsymbol{w}_\Lambda\|_2 \tag{6.162}$$

and inserting this into equation (6.156) leads to

$$\|\boldsymbol{w}\|_2 \leq C_0 \frac{\sigma_k(\boldsymbol{w}^*)_1}{\sqrt{k}} + C_1 2\epsilon \sqrt{1+\delta_{2k}}. \tag{6.163}$$

We can now adjust the constant $C_2 := \sqrt{1+\delta_{2k}} C_1$ to get the result

$$\|\hat{\boldsymbol{w}} - \boldsymbol{w}^*\|_2 \leq C_0 \frac{\sigma_k(\boldsymbol{w}^*)_1}{\sqrt{k}} + C_2\epsilon \tag{6.164}$$

which is the equation from theorem 10.

6.4 Overview: Dynamic Compressed Sensing of Model Errors

For the sake of a clearer understanding of this very technical chapter, let us interpret the results to round up the scene.

First, we have seen that measurement noise spoils the independence structure which was presented in chapter 5. To remedy this issue, the idea of a *compressed solution* was discussed: Instead of finding a sparse solution

$$\Phi(\boldsymbol{u}) = \boldsymbol{y}^{\text{data}} \tag{6.165}$$

we search for a sparse solution of

$$\|\Phi(\boldsymbol{u}) - \boldsymbol{y}^{\text{data}}\|_2 \leq \epsilon \,. \tag{6.166}$$

We allow a small deviation of the augmented model $\Phi(\boldsymbol{u})$ from the data. In return we can *compress* the input vector $\boldsymbol{u}$ and set the small but non-vanishing components to zero. By this we achieve a sparser solution that still matches the data reasonably good.

We presented the P0 and P1 problems, with the aim to minimise $\|\boldsymbol{u}\|_0$ or $\|\boldsymbol{u}\|_1$, respectively, with respect to (6.166). Since $\|\boldsymbol{u}\|_0$ counts the number of non-zero components, P0 is exactly the problem we want to solve, i.e., finding a sparse input vector that approximately reproduces the data. As this is a non-convex optimisation problem, $P1$ was discussed as a convex alternative. For the formulation of P0 and P1 we need a norm on the input and output spaces $\mathscr{U}$ and $\mathscr{Y}$. We presented a compound p-q-norm

$$\|\boldsymbol{u}\|_q = \|\underline{\boldsymbol{u}}\|_q \quad , \quad \underline{u_i} = \|u_i\|_{L^p} \,. \tag{6.167}$$

In contrast to other varieties of dynamic compressed sensing, the p-1-norm helps to achieve the desired *invariable sparsity*, whereas for example the 1-1-norm and the 2-2-norm are taken for hands-off control or in the DEN. The dependence on the parameter p can be suppressed, as our results stay valid for any choice of p. So we usually simply speak of the q-norm.

A version of the Restricted-Isometry-Property that is consistent with this new q-norm was presented and we were finally able to show that if the system fulfils this property, then P1 indeed leads to a sparse solution of the dynamic compressed sensing of model errors.

CHAPTER 7

Applications

The theoretical framework for the localisation and reconstruction of structural model errors presented in this thesis comprises three central elements: The gammoid interpretation of input-output systems, a convex optimisation problem, and measures of coherence for error sources as well as for sensors. It is an urge to finally provide possible applications of the developed theory. We will discuss applications to the European power grid, to a biological neural network, to the famous chaotic Lorenz system, and finally we demonstrate an iterative error localisation strategy for a highly under-determined synthetic system.

Fault Detection *Fault diagnosis* is an important and widely discussed topic in the engineering literature, see for instance [39, 50] for textbooks or [41] for recent developments in unknown input observers for autonomous spacecrafts. Though fault diagnosis might be seen as a synonym for error detection, i.e., to construct an *unknown input observer* must be distinguished from sparse error reconstruction. In a linear (or linearised) system

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= A\boldsymbol{x}(t) + B\boldsymbol{u}(t) \\ \boldsymbol{y}(t) &= C\boldsymbol{x}(t) \end{aligned} \tag{7.1}$$

one necessary requirement for the existence of an unknown input observer is the condition [139]

$$\operatorname{rank} CB = \operatorname{rank} B\,. \tag{7.2}$$

This condition can be translated into gammoid language as

$$\mathsf{S} \subseteq \mathsf{Z}\,, \tag{7.3}$$

i.e., it is required to cover each potential input node with a sensor.

In engineering problems, where the systems go through a design process and are build to work in a specific way, this assumption might be justified. Each component of the system is well known and sensors are designed to detect a malfunction of the monitored element. The fault detection approach stemming from the gammoid interpretation makes no prior assumptions and is not designed for a certain system. Instead, it works on the very topological structure and utilises the spark to ensure uniqueness of an invariable sparse error localisation.

Model Correction In 1963 Lorenz simplified the description of the *cellular convection* and developed the *convection equations* [140]

$$\begin{aligned} \dot{x}(t) &= \sigma y(t) - \sigma x(t) \\ \dot{y}(t) &= -x(t)z(t) + \rho x(t) - y(t) \\ \dot{z}(t) &= x(t)y(t) - \beta z(t)\,. \end{aligned} \tag{7.4}$$

The parameters

$$\rho = 28 \quad , \quad \sigma = 10 \quad , \text{and} \quad \beta = 8/3 \tag{7.5}$$

are a standard choice and be traced back to earlier works of Saltzmann [141]. The Lorenz system, though comprising only three state variables as apparently simple differential equations, offers a chaotic behaviour and is therefore well suited to investigate the efficiency of data driven methods under numerically stiff conditions. Data-driven methods like *SINDy* aim to deduce the 7 terms of the Lorenz system solely from the measurement data [33]. Such purely data-driven approaches try to infer the governing differential equations *from scratch* and become computationally infeasible as the systems get larger and more complex.

Our approach would rather be the following: Say, we have a first guess, a linear nominal model

$$\begin{aligned} \dot{x}(t) &= \sigma y(t) - \sigma x(t) \\ \dot{y}(t) &= \rho x(t) - y(t) \\ \dot{z}(t) &= -\beta z(t)\,. \end{aligned} \tag{7.6}$$

One will see that the nominal model has a model error

$$\boldsymbol{w}^*(t) = \big(0, -x(t)z(t), x(t)y(t)\big)^T\,. \tag{7.7}$$

Thus, we make use of our nominal guess and compute a quantitative estimate $\hat{\boldsymbol{w}}$. One could then for instance proceed with a model identification method applied to the remaining 2 non-vanishing components of the model error, to finally deduce a differential equation that describes the model error.

So, the second application focuses on the non-linearities of the Lorenz system. In order to compute quantitative estimates we tackle the convex P1 problem. Let henceforth $\Delta : \boldsymbol{y}^{\text{data}} \mapsto \hat{\boldsymbol{w}}$ denote a recovery algorithm that maps the data to an estimate for the model error. In practice, a minimisation of the cost functional

$$J[\boldsymbol{u}] = \frac{\alpha}{2}\|\boldsymbol{y}^{\text{data}} - \Phi(\boldsymbol{u})\|_2^2 + \beta\|\boldsymbol{u}\|_1 \tag{7.8}$$

is equivalent to P1, see for instance [118]. This again is a standard form for dynamic optimisation. We used the `python` implementation of `CasADi` to solves this problem computationally. In the remainder, Δ will refer to a minimisation of (7.8) via `CasADi`. Another recovery algorithm would be the `R` implementation of the *Dynamic Elastic Net* [7].

Clustering Approach for Input Localisation and Sensor Placement Finally we consider a case that will occur frequently in real problems: The system is non-invertible and from the experimental side it is impractically to observe all the state variables that would be necessary to obtain invertibility. It is therefore a relevant question if it is possible to relax the exact localisation of model errors and instead seek for the best localisation given the insufficient data.

We can make use of the coherence in a twofold way: First, we can search for clusters of indistinguishable input nodes. As one example, we will present the input clusters of the neural network of *Caenorhabditis elegans (C.elegans)*. *C.elegans* is an approximately 1 mm long worm which attracted attention for it was the first individual whose neural structure was mapped and catalogued completely. As another example, we consider a non-invertible version of the $N = 30$ nodes example of chapter 5 again and search for input clusters. Though we cannot localise the model error within one cluster, we will at least be able to distinguish the *active* clusters which contain one or more model errors, from the *inactive* clusters for which we can surely say that they do not contain any unknown inputs. Second, we can search for clusters of indistinguishable output nodes. Sensors that lie within the same cluster provide the same information about the model errors. One can use this information to replace redundant sensors and by this increase the information content of the data.

7.1 Fault Detection

A single fault in a system can take different forms, so that an invariable sparse fault does not necessarily mean invariable 1-sparse. Consider the following two trains of though: A malfunction of a single element, for instance one component x_i of an electric circuit, can be remedied by a single additional input

$$\dot{x}(t) = \tilde{f}_i(\boldsymbol{x}(t)) = f_i(\boldsymbol{x}(t)) + w_i^*(t)\,, \tag{7.9}$$

where $\tilde{f}_i$ is a unknown function that represents the malfunctioning component, whereas f_i models the known, normal behaviour of this component. The model error $\boldsymbol{w}^* = (0,\ldots,0,w_i^*,0,\ldots,0)^T$ representing such a malfunction is characterised by

$$\|\boldsymbol{w}^*\|_0 = 1\,. \tag{7.10}$$

Remember

$$\|\hat{\boldsymbol{w}}\|_0 < \frac{\operatorname{spark}\Gamma}{2} \Rightarrow \hat{\boldsymbol{w}} \text{ is the unique sparsest solution}\,. \tag{7.11}$$

We come to the conclusion that $\operatorname{spark}\Gamma \geq 3$ is necessary to guarantee the uniqueness of an 1-sparse error estimate. Let $\Gamma = (\mathtt{L},\mathtt{g},\mathtt{M})$ be a gammoid. A spark of 3 is achieved, if every input set $\mathtt{S} \subseteq \mathtt{L}$ of size two is linked into $\mathtt{M}$. As a simple corollary, note that a minimum of two sensors is necessary to localise a single broken component.

But a single fault can also lie in the interaction of two components, as for instance a misspecified transition or transport, say from x_1 to x_2. In such a case, the time course of both reactants are affected and two inputs are needed to correct the model,

$$\begin{aligned} \dot{x}_1(t) &= \tilde{f}_1(\boldsymbol{x}) = f_1(\boldsymbol{x}) + w_1^*(t) \\ \dot{x}_2(t) &= \tilde{f}_2(\boldsymbol{x}) = f_2(\boldsymbol{x}) + w_2^*(t)\,. \end{aligned} \tag{7.12}$$

For a unique localisation of the fault, i.e., of w_1^* and w_2^*, a spark of 5 and at least 4 sensors are required.

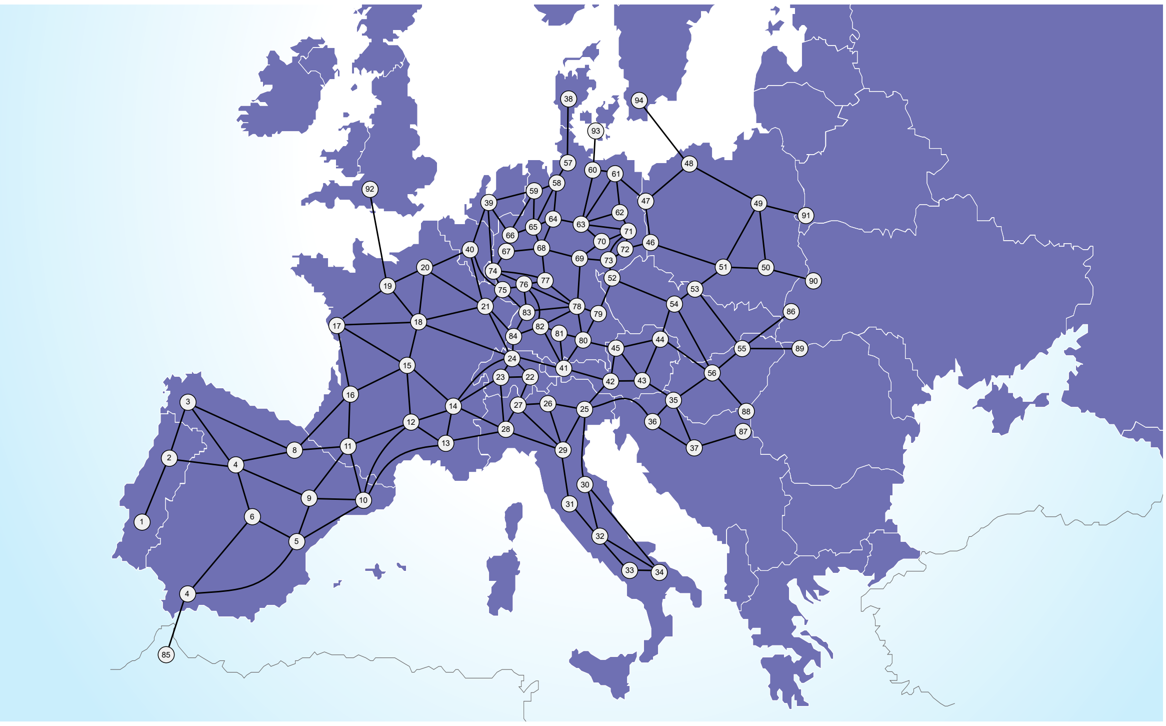

Figure 7.1: A map of the European power grid UCTE, status as of summer 2004. The nodes represent large power and transformer stations as presented in [142]. Each edge represents a direct connection in both directions, for a better visualisation represented by a single undirected edge. 🅭🅯 *Map of Europe* by albertoalvarez.es, source [143], image section and colour adjusted. Graph illustrating the UCTE as presented in [142].

European Power Grid

As one application, let us consider the European power grid *UCTE* presented in figure 7.1, where each node represents a large power or transformer station and each edge a power line in both directions. In [142], the fault robustness and synchronisation of the European power grid was discussed focusing on two malfunctions which occurred on April 2 2003 at *Paluel* (node 19) and on June 5 2003 at *Schwarze Pumpe* (node 72). Later, network controllability principles were utilised to find optimal actuator placements for the UCTE [144]. These two treatments, however, build on the assumption that each node of the power grid can be observed.

In the following, let $\mathtt{g}$ be the graph with nodes $\mathtt{N} = \{1, \dots, 94\}$ and edges shown in figure 7.1, where each link represents two directed edges (always understood in both directions). We do not want to make prior assumptions about malfunctions, so the input ground set is $\mathtt{L} = \mathtt{N}$. We search for a minimal sensor placement, i.e., for an output set $\mathtt{Z} \subseteq \mathtt{N}$ that leads to a sufficiently large spark of the gammoid $\Gamma = (\mathtt{L}, \mathtt{g}, \mathtt{Z})$ and allows for a detection of malfunctions in the UCTE.

1-Sparse Fault The malfunction of a single power plant falls in the category of 1-sparse faults. From the discussion above we have learned that the unique detection of a 1-sparse input requires at least two sensors and a spark of three. However, we soon encounter a difficulty when we, for instance, consider the nodes 1 and 2 in figure 7.1: Each path starting or terminating at node 1 will unavoidably pass node 2. Thus, the set $\mathrm{S} = \{1,2\}$ is dependent, unless $1 \in \mathrm{Z}$. The same argumentation holds for each node which is connected to only one other node of the graph. Consequently, the smallest output set capable of rendering each size two input set independent is

$$\mathrm{Z}_1 = \{1,38,85,86,87,88,89,90,91,92,93,94\}. \tag{7.13}$$

Indeed, we find that

$$\operatorname{spark}(\mathrm{L},\mathrm{g},\mathrm{Z}_1) = 3. \tag{7.14}$$

2-Sparse Fault A damaged power cable affects the electricity transmission between two nodes. It leads to a 2-sparse model and requires a spark of five to be detectable in the sense of gammoid theory. The output set Z_1, though its size is clearly over the minimum of four sensors, is not sufficient.

Here we see by example that already the search for a sensor set that yields a spark of five becomes costly: Given a candidate Z_2 for the output set, a spark of five requires all 4-node input sets to be independent. For the UTCE with $N = 94$ nodes this makes

$$N(N-1)(N-2)(N-3) = 73.188.024 \tag{7.15}$$

input sets which must be considered. The approximation via theorem 9 does neither lead to useful results, as in most cases the shortest path mutual coherence equals one and thus leads to the trivial bound

$$\operatorname{spark}\Gamma \geq 2. \tag{7.16}$$

For practical purposes, both approaches are hard to handle and motivate the clustering approach in section three.

Summary This section considered the problem of fault diagnosis in networks. A fault in a network can appear in different forms: For instance, the malfunction of a single component introduces a 1-sparse model error, whereas a broken connection between two components will introduce a 2-sparse model error. In order to detect the fault that causes of a k-sparse model error, at least $2k$ sensors are needed, and a spark of at least $2k+1$.

For the example of the European power grid UCTE we have, however, seen that a number of $P = 12$ sensors is needed to ensure the detectability of a 1-sparse malfunction. Though this number is clearly above the lower bound of 2, one must keep in mind that the UCTE comprises $N = 94$ power plants each of which could be the source of a model error. In a network like the UCTE, a sensor placement that respects the gammoid structure can be capable of ensuring a unique fault detection, even if the system size N exceeds the possible number of sensors P. It must be mentioned that our analysis takes only the gammoid structure of the UCTE into account. In reality, economical and political factors will probably play the dominant role when it comes to the organisation of an international power grid.

It shall also be emphasised that the gammoid approach takes only the network topology into account. This comes with the advantage that the sensor placement does not rely on a concrete model about the interactions and stays untouched even if the numerical properties of the model chance (for instance, consider the exchange of a power cable to reduce electrical losses). However, the disadvantage emerging from the gammoid approach is the blindness to numerical issues. So, though the solution of the model error reconstruction might be unique in theory, a numerically ill-conditioned problem or strong measurement noise might make a trustworthy error estimation practically impossible.

7.2 The Lorenz System

The equations of the Lorenz system (7.4) provide highly nonlinear, chaotic dynamics. It shall therefore be exploited to learn more about the capabilities of the convex optimisation P1. So assume we have measured data $(x^{\text{data}}, z^{\text{data}})$ for the Lorenz system. Here, we purposefully omit data for the state variable y for the reason that our approach is especially suited to handle systems where we cannot measure each state variable.
Often, nominal models describe a first attempt to characterise a system. It is usually justified to start with a linear model. Thus, we want to consider the system (7.6) as a nominal guess for the Lorenz system. We now want to compare the flow $\varphi_t(\boldsymbol{x}_0)$ of the nominal system to the given data in order to compute a quantitative estimate of the model error (7.7). As initial value we choose $\boldsymbol{x}_0 = (1,1,1)$.

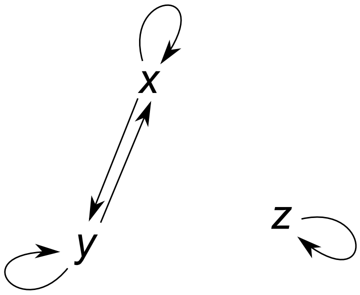

Figure 7.2: The influence graph of the linearised Lorenz system.

Input-Output Map The augmentation of the nominal model leads to the input-output system

$$\begin{aligned}\dot{x}(t) &= \sigma y(t) - \sigma x(t) + u_1(t)\\ \dot{y}(t) &= \rho x(t) - y(t) + u_2(t)\\ \dot{z}(t) &= -\beta z(t) + u_3(t)\,.\end{aligned} \tag{7.17}$$

One will find the flow of this system to be

$$\varphi_t^{\boldsymbol{u}}(\boldsymbol{x}_0) = \mathrm{e}^{At}\boldsymbol{x}_0 + \int_0^t \mathrm{e}^{A(t-\tau)}\boldsymbol{u}(\tau)\,\mathrm{d}\tau \tag{7.18}$$

with a matrix

$$A := \begin{pmatrix} -\sigma & \sigma & 0 \\ \rho & -1 & 0 \\ 0 & 0 & -\beta \end{pmatrix}. \tag{7.19}$$

See figure 7.2 for the influence graph of the nominal model. There, one will realise that z is isolated, whereas x and y are connected in both directions. For this reason, in order to infer a model error targeting z we must measure this state variable directly. Under the assumption of invariable sparsity it is, however, sufficient to measure one of the state variables x and y. We chose x as observable, so that the measurement function takes the form of the matrix

$$C = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{7.20}$$

and the input-output map

$$\Phi(\boldsymbol{u})(t) = C\varphi_t^{\boldsymbol{u}}(\boldsymbol{x}_0)\,. \tag{7.21}$$

Linear P1 One will realise that the input-output map (7.21) is not linear but affine-linear as it takes the form

$$\Phi(\boldsymbol{u}) = \underbrace{C\mathrm{e}^{At}\boldsymbol{x}_0}_{=:\Phi_1(t)} + \underbrace{\int_0^t C\mathrm{e}^{A(t-\tau)}\boldsymbol{u}(\tau)\,\mathrm{d}\tau}_{=:\Phi_2(\boldsymbol{u})(t)} \tag{7.22}$$

where Φ_1 is constant with respect to $\boldsymbol{u}$. Since the convexity of P1 is only guaranteed for linear input-output maps, we have to perform the transformation

$$\begin{pmatrix} \tilde{x}^{\text{data}}(t) \\ \tilde{z}^{\text{data}}(t) \end{pmatrix} = \begin{pmatrix} x^{\text{data}}(t) \\ z^{\text{data}}(t) \end{pmatrix} - C\left(\boldsymbol{x}_0 + \int_0^t \mathrm{e}^{A(t-\tau)}\boldsymbol{x}_0\,\mathrm{d}\tau\right) \tag{7.23}$$

and solve the problem

$$\Phi_2(\boldsymbol{u}) = \begin{pmatrix} \tilde{x}^{\text{data}}(t) \\ \tilde{z}^{\text{data}}(t) \end{pmatrix} \tag{7.24}$$

where Φ_2 is a linear map.

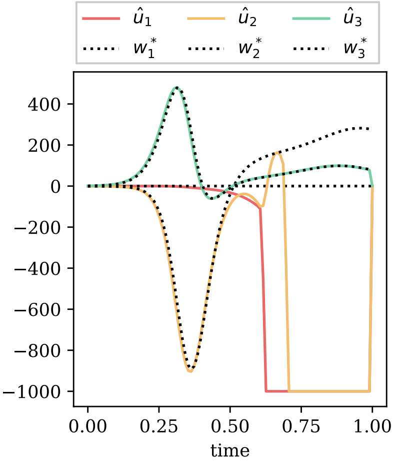

Figure 7.3: Result of the error reconstruction for the linearised Lorenz model with data transformation over a time range $[0, 1]$. One can see that the error estimates $\hat{u}_i$ fit the ground truth w_i^* between $t = 0$ and $t \approx 0.5$. For larger t, $\hat{u}_1$ and $\hat{u}_2$ differ significantly from the ground truth.

Find the trajectory $(x(t), y(t), z(t))$ of the Lorenz system for the time interval $[0, 15]$ before and after transformation in figure 7.4(a) and (b), respectively. To compute estimates for the model error we minimise the cost functional

$$J[\boldsymbol{u}] = \alpha \left\| \begin{pmatrix} \tilde{x}^{\text{data}}(t) \\ \tilde{z}^{\text{data}}(t) \end{pmatrix} - \Phi_2(\boldsymbol{u}) \right\|_2^2 + \beta \|\boldsymbol{u}\|_1 \tag{7.25}$$

with $\alpha = 1$ and $\beta = 10^{-3}$. See figure 7.3 for the estimate in the time interval $[0, 1]$.

One will realise that for $t \lessapprox 0.5$ the estimates $\hat{u}_i$ approximate the true model errors w_i^* accurately. For times later than $t \approx 0.5$, however, the estimates do not match the ground truth any more. Figure 7.4(b) and (d) will clarify thus issue. In subfigure (b) one will realise that, after data transformation, the trajectory diverges. Subfigure (d) shows that around $t \approx 5$ the magnitude of the state variables x and y start to leave any bounded region.

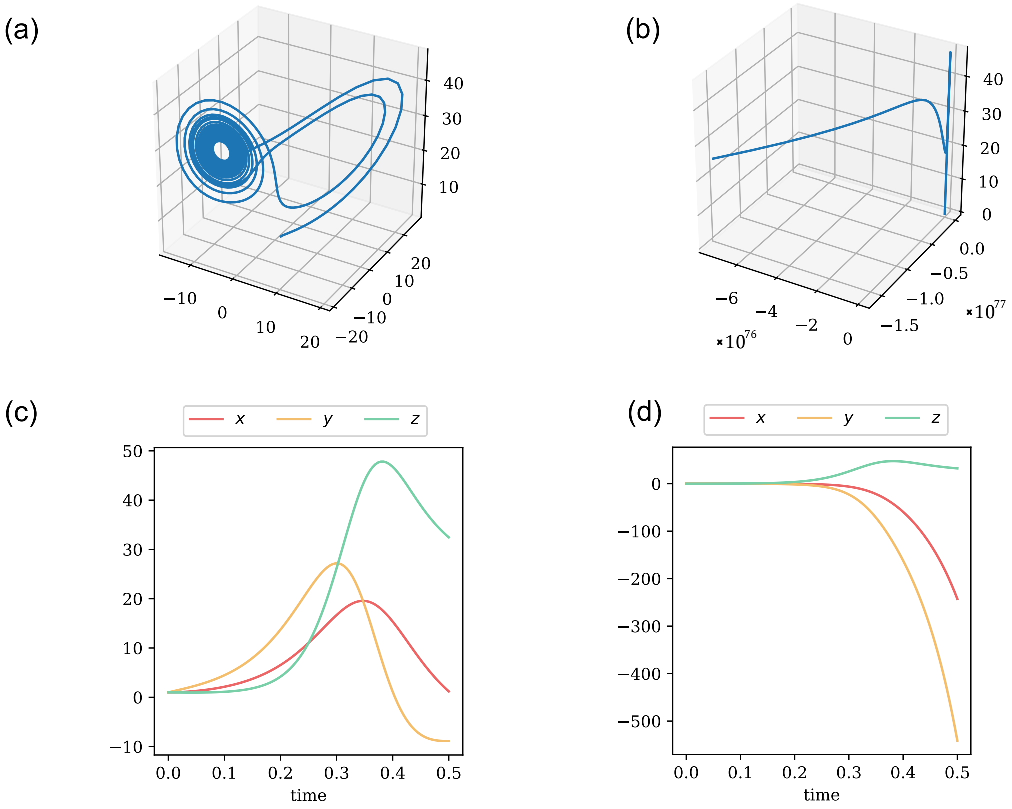

Figure 7.4: Trajectories of the Lorenz system before and after data transformation. (a) The trajectory before data transformation over the time range $[0,15]$. One can identify the well known Lorenz-attractor. (b) The trajectory after data transformation over the time range $[0,15]$. The trajectory is unbounded as time increases. (c) The three state variables before data transformation over a smaller time range. The values stay in a bounded range. (d) The three state variables after data transformation. The absolute value of the state variables x and y grow over limitless as the time exceeds $t \approx 0.5$.

Non-linear P1 Though our theoretical results are only proven for linear input-output maps, let us consider the cost functional

$$J[\boldsymbol{u}] = \alpha \left\| \begin{pmatrix} x^{\text{data}}(t) \\ z^{\text{data}}(t) \end{pmatrix} - \Phi(\boldsymbol{u}) \right\|_2^2 + \beta \|\boldsymbol{u}\|_1 \tag{7.26}$$

with non-linear Φ. Figures 7.5(a-c) show the results of the input estimation with a regularisation parameter of $\beta = 10^{-5}$ for the time interval $[0,15]$.

The estimate $\hat{w}_1$ is relatively small in magnitude, compared to the estimates $\hat{w}_2$ and $\hat{w}_3$ but it is not zero, though this would match the ground truth $w_1^* = 0 \forall t$. Remember that in the 2-norm on the output space

$$\|\Phi(\boldsymbol{u})\|_2^2 = \|\underline{\Phi(\boldsymbol{u})}\|_2^2 = \sum_{i=x,z} \|\Phi_i(\boldsymbol{u})\|_{L^p}^2 \tag{7.27}$$

there is a L^p-norm hidden in the underline. While our theoretical results for sparse input estimation

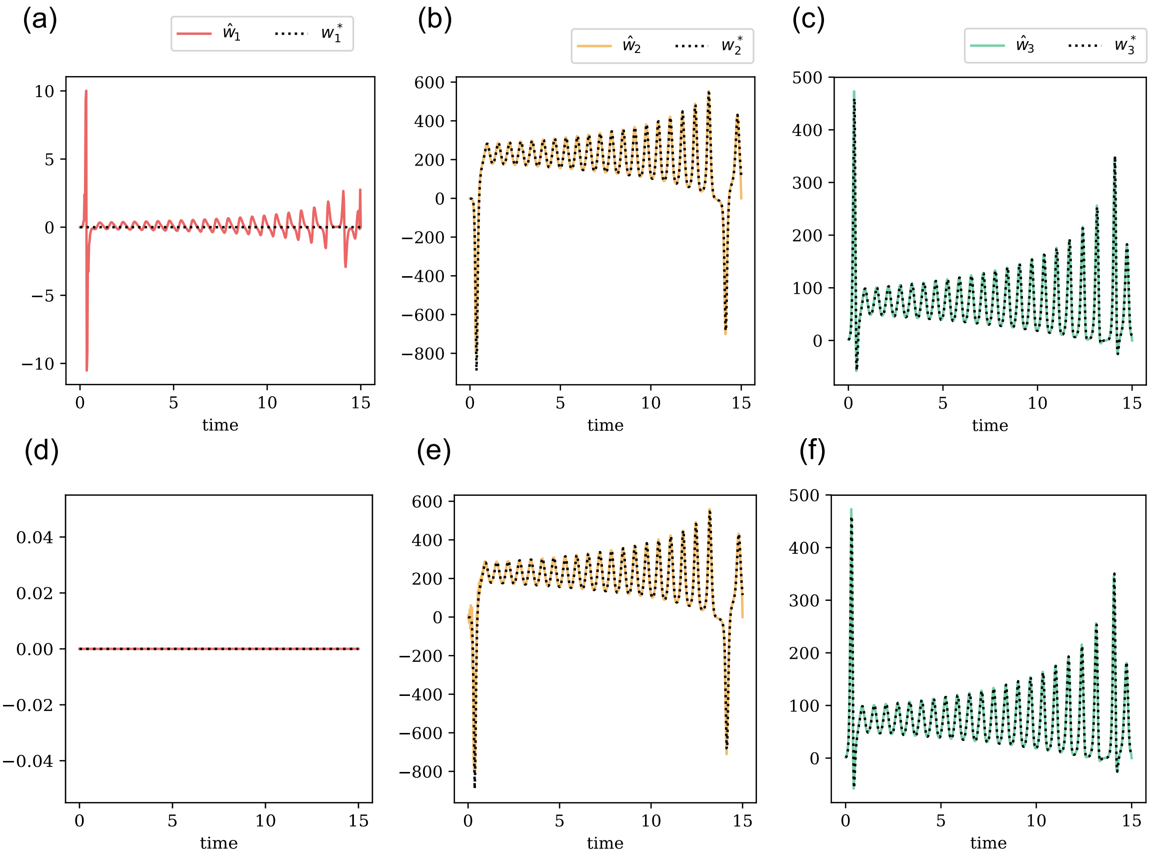

Figure 7.5: Error reconstruction for the linearised Lorenz system without data transformation over the time range [0, 15]. (a) Estimate $\hat{u}_1$ and ground truth w_1^*. (a) Estimate $\hat{u}_2$ and ground truth w_2^*. (a) Estimate $\hat{u}_3$ and ground truth w_3^*.

are valid for any $1 \leq p < \infty$, the choice $p = 2$ make the output space Hilbert. For inverse problems on a Hilbert space, sparsity can be achieved by Iterative-Threshold-Algorithms [145, 146, 147]. In this example, Iterative-Thresholding would imply that we force the smallest input estimate to zero. Figures 7.5(d-f) show the input estimates for $\hat{w}_1 \overset{!}{=} 0$. After one step of Iterative-Thresholding, all three input estimates $\hat{w}_i$ offer accurate descriptions of the true model errors w_i^*.

Note that the greedy approach to the *Dynamic Elastic Net* [7] implements a variety of Iterative-Thresholding which has already proven to produce good appropriate estimates of model errors in linear and non-linear systems.

Summary In this section, we have started with a nominal, linear model for the Lorenz system and inferred the non-linearities from data for the chaotic Lorenz attractor. In order to obtain a linear input-output map for the augmented nominal model, we had to perform a transformation of the data. This data transformation lead to the problem that the data start to diverge as time exceeds $t_1 \approx 0.5$. As a consequence, we found that only for $t < t_1$ we were able to reconstruct the model error accurately. Our theoretical results for the input estimates were proven only for a linear input-output map that fulfils the RIP2k condition to a sufficient order. In practice, it is hardly possible to verify the RIP2k, however, the divergence of the transformed trajectory for $t > t_1$ lead to the presumption that the RIP2k is clearly not fulfilled for large intervals of time. This presumption is consistent with the result that our input estimates do not match the ground truth for $t > t_1$.

In a second trial, we skipped the data transformation and worked with a non-linear input-output map. The non-linear map comes with the advantage that its image stays in a bounded region. We were able to infer accurate estimates for the model errors for a large interval of time. Though only proven for linear problems, we come to the conclusion that the optimisation approach for sparse input estimation can produce accurate results also for a non-linear input-output map. Engelhardt et al. [21, 22] used a related optimisation problem were also able to produce good error estimates for non-linear systems. Thus, one can be optimistic that the meaningfulness of the problem P1 will also be verified for non-linear systems in the future.

7.3 Cluster Approach

For a linear system, we have seen that the coherence $\mu_{ij}(s)$ makes a statement about the distinguishability of inputs attacking the input nodes $\mathtt{s}_i$ and $\mathtt{s}_j$. It will be convenient to reformulate the coherence $\mu_{ij}(s)$ as a distance $d_{ij}(s)$ between input nodes according to

$$d_{ij}(s) := \frac{1}{\mu(s)_{ij}} - 1\,. \tag{7.28}$$

One will see that two coherent nodes i and j have a short distance, $d_{ij}(s) \approx 0$, with respect to this metric whereas incoherence translates to a long distance, $d_{ij}(s) \to \infty$. As it is not unusual that a coherence tends to zero or vanishes exactly, the distances are likely to diverge. For practical purposes it makes sense, to define an upper bound $d^{\max}$. Furthermore it is again convenient to work with shortest paths, i.e.,

$$d_{ij}^{\text{short}} = \frac{1}{\mu_{ij}^{\text{short}}} - 1\,. \tag{7.29}$$

We call

$$D := \left(d_{ij}^{\text{short}}\right)_{ij} \tag{7.30}$$

the **(shortest path) distance matrix**, often we work with the normalised matrix

$$\hat{D} := \frac{D}{\max_{ij} d_{ij}}\,. \tag{7.31}$$

For this thesis, `python` with the packages `scipy` and `seaborn` were used to compute node clusters according to the distance matrix.

7.3.1 Caenorhabditis Elegans

Though biologists have gathered lots of information about the interplay of neurons and synapses on cellular level, a better understanding of the working principles of the brain or a neural network in general still remains a central task for modern sciences, find a comprehensive textbook which covers biophysical aspects as well as mathematical models and neural information coding in [148]. Inspired by generically grown neural networks in living organisms, modern computers enable us to implement artificial neural networks. This development makes the investigation of neural structures a relevant subject also for computer scientists in machine learning and artificial intelligence, find a contemporary review in [149]. Recently, dynamic systems were presented as a model for continuous neural networks [150].

The online resources [151] and [152] provide comprehensive information as well as peer-reviewed publications about the neural network of *C.elegans*. *C.elegans*, see figure 7.6, became a prominent research biologists managed to completely map its neural structure and by this made it possible to study a full connectome. We work with the data set from [153] provided online [152]. See figure 7.7(a) for a visualisation of the *C.elegans* connectome. The connectome has been the object of network scientific studies in recent years. In [154] and [153] the network principles and an optimisation of the network layout were discusses. The network controllability was addressed in [155] where it was shown that network control principles allow to connect the structure of the connectome to its function for the locomotion of the worm. The latter result was consistent with [156] where it was shown that the worm's degree of motion is 95% described by only four motions, the so called *eigenworms*.

Despite this consistency between the controllability framework and the experimentally observed four *eigenworms*, one will agree that the neural network is, from the biological point of view, not a control system. In a control system, external inputs are applied in order to steer the system into a desired behaviour. The neural network rather acts like a information processing or error detection system: If the worm encounters a barrier or receives a thermal stimulus [156], it reacts (or encodes this information) in form of a movement.

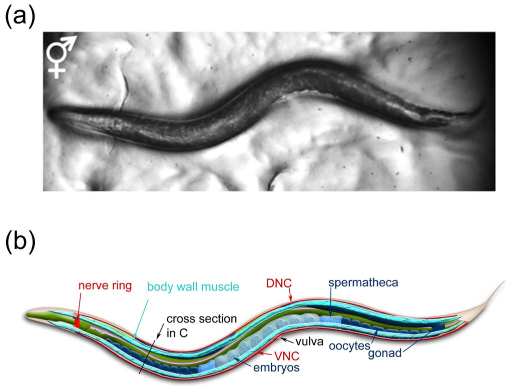

Figure 7.6: A photo and a schematic illustration of an *C.elegans* individual of hermaphrodite sex. 🅭🅯 2015 *Life Cycle of C.elegans* and *C.elegans anatomy* by Ann K. Korsi, Bruce Wightman, and Martin Chalfie, source and information [157]. The figure shows image sections of the original. (a) A photo of an adult *C.elegans* individual. (b) A schematic illustration of the worm's physique.

We can reinterpret the connectome of *C.elegans* in terms of a weighted gammoid: The connectome (the data set we use) comprises a total of 283 neurons (the node set $\mathtt{N}$) which split into 82 sensor neurons, 93 inter neurons, and 108 motor neurons. Furthermore, we have a set of 4693 neural junctions (the directed edges $\mathtt{E}$). The neural junctions take the form of synapses, more precisely, the data set provides the number of synapses between each pair of neurons. We can take this multiplicity of synapses as weight function $F : \mathtt{E} \to \mathbb{R}$. Finally, there are three kinds of neurons. Sensor neurons act as input nodes and can therefore be regarded the input ground set $\mathtt{L}$, inter neurons process the information, and motor neurons are finally connected to muscles and can be interpreted as the output ground set $\mathtt{M}$. Altogether, we obtain a representation of the *C.elegans* connectome in form of the weighted gammoid $\Gamma = (\mathtt{L}, \mathtt{g}, \mathtt{M})$. In [155] it was already argued why the dynamics of the neural network can be approximated by a linear system. It is therefore justified to compute coherences and distances for the input nodes of the gammoid, which we have so far only defined for linear systems.

Coherence Structure of the C.elegans Connectome According to (7.30) we can use the weighted *C.elegans* gammoid to compute a distance matrix for the sensor neurons L. A visualisation as heat map can be found in figure 7.7(a). Along the diagonal axis we identify four or five larger clusters, where the exact assignment of the clusters depends on the desired clustering depth, plus some smaller clusters. For stimuli within one input cluster, the generated output of the network, in this case the provoked movement, is indistinguishable. Whereas stimuli of sensor nodes of distinct clusters are distinguishable by their produced output. Though the precise number and size of the clusters contains some degree of arbitrariness, we find that our results are consistent with the results of [156], i.e., that the major part of the worm's locomotion is covered by very few distinct moves.

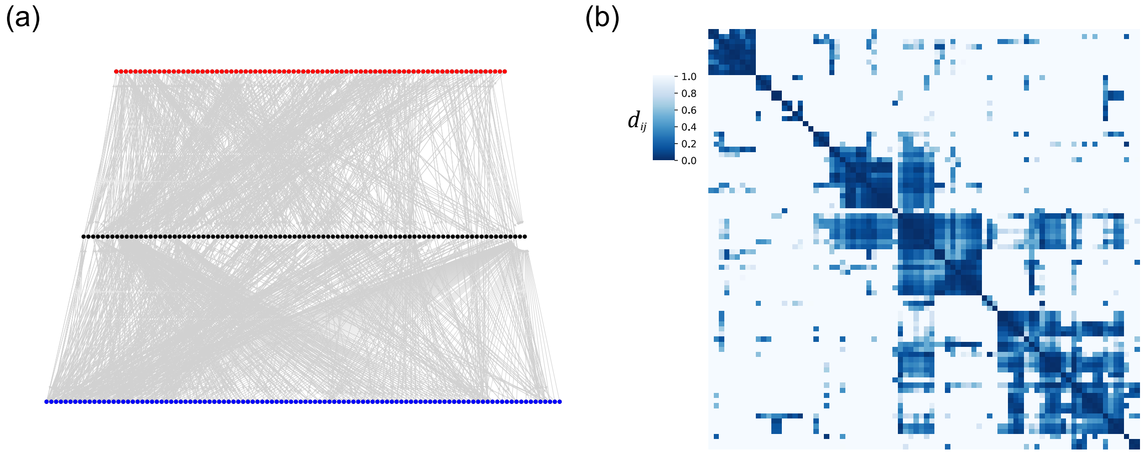

Figure 7.7: Neural structure of hermaphrodite *C.elegans*. (a) The neuronal network in a simplistic 3-layer illustration. The data set, published in [153] and online available [152], comprises 283 nodes comprising 82 input neuron (red nodes, top layer), 93 inter neurons (black nodes, middle layer), and 108 motor neurons (blue nodes, bottom layer). Furthermore there are 4693 synaptic links (grey arrows). The real neural network does not have a strict cascade structure, i.e., there are direct edges from the top to the bottom layer as well as edges directed upwards. (b) Heat map of the normalised distance matrix for the 82 sensor neurons. For the pairwise distance d_{ij} an upper bound of $d_{\max} = 10$ was introduced.

7.3.2 Iterative Error Detection

In the remainder we will consider a system of $N = 30$ nodes which we already know from the introductory example of chapter 5. The differential equations of the nominal model read as follows:

$$
\begin{aligned}
&\dot{x}_1(t) = x_7(t) + x_{26}(t) - \alpha x_1(t)\\
&\dot{x}_2(t) = x_3(t) + x_{14}(t) - \alpha x_2(t)\\
&\dot{x}_3(t) = x_2(t) + x_9(t) + x_{11}(t) + x_{22}(t) + x_{30}(t) - \alpha x_3(t)\\
&\dot{x}_4(t) = x_6(t) + x_{14}(t) + x_{16}(t) + x_{21}(t) + x_{28}(t) - \alpha x_4(t)\\
&\dot{x}_5(t) = x_8(t) + x_{16}(t) + x_{19}(t) + x_{20}(t) + x_{29}(t) - \alpha x_5(t)\\
&\dot{x}_6(t) = x_8(t) + x_{14}(t) + x_{15}(t) + x_{20}(t) + x_{30}(t) - \alpha x_6(t)\\
&\dot{x}_7(t) = x_{17}(t) + x_{24}(t) - \alpha x_7(t)\\
&\dot{x}_8(t) = x_{21}(t) - \alpha x_8(t)\\
&\dot{x}_9(t) = x_{15}(t) + x_{21}(t) + x_{30}(t) - \alpha x_9(t)\\
&\dot{x}_{10}(t) = x_1(t) + x_2(t) + x_{11}(t) + x_{13}(t) - \alpha x_{10}(t)\\
&\dot{x}_{11}(t) = x_9(t) + x_{13}(t) + x_{14}(t) + x_{21}(t) - \alpha x_{11}(t)\\
&\dot{x}_{12}(t) = x_7(t) + x_{14}(t) - \alpha x_{12}(t)\\
&\dot{x}_{13}(t) = x_6(t) + x_{12}(t) + x_{15}(t) + x_{22}(t) + x_{30}(t) - \alpha x_{13}(t)\\
&\dot{x}_{14}(t) = x_2(t) + x_{16}(t) + x_{18}(t) + x_{29}(t) - \alpha x_{14}(t)\\
&\dot{x}_{15}(t) = x_{23}(t) - \alpha x_{15}(t)
\end{aligned}
$$

$$
\begin{aligned}
&\dot{x}_{16}(t) = x_5(t) + x_7(t) + x_{14}(t) + x_{20}(t) + x_{23}(t) + x_{28}(t) - \alpha x_{16}(t)\\
&\dot{x}_{17}(t) = -\alpha x_{17}(t)\\
&\dot{x}_{18}(t) = x_{16}(t) + x_{25}(t) + x_{27}(t) + x_{28}(t) - \alpha x_{18}(t)\\
&\dot{x}_{19}(t) = x_4(t) + x_7(t) + x_9(t) + x_{27}(t) + x_{30}(t) - \alpha x_{19}(t)\\
&\dot{x}_{20}(t) = x_6(t) + x_{18}(t) + x_{19}(t) + x_{26}(t) + x_{27}(t) - \alpha x_{20}(t)\\
&\dot{x}_{21}(t) = x_6(t) + x_{12}(t) + x_{17}(t) + x_{20}(t) + x_{22}(t) + x_{23}(t) +\\
&\quad x_{24}(t) + x_{25}(t) + x_{27}(t) + x_{30}(t) - \alpha x_{21}(t)\\
&\dot{x}_{22}(t) = x_{26}(t) + x_{27}(t) + x_{29}(t) - \alpha x_{22}(t)\\
&\dot{x}_{23}(t) = x_3(t) + x_5(t) + x_{14}(t) + x_{21}(t) + x_{25}(t) + x_{30}(t) - \alpha x_{23}(t)\\
&\dot{x}_{24}(t) = x_{10}(t) + x_{27}(t) - \alpha x_{24}(t)\\
&\dot{x}_{25}(t) = x_5(t) + x_{10}(t) - \alpha x_{25}(t)\\
&\dot{x}_{26}(t) = x_6(t) + x_{10}(t) - \alpha x_{26}(t)\\
&\dot{x}_{27}(t) = x_4(t) + x_{30}(t) - \alpha x_{27}(t)\\
&\dot{x}_{28}(t) = x_6(t) + x_{14}(t) + x_{16}(t) + x_{22}(t) + x_{26}(t) + x_{27}(t) - \alpha x_{28}(t)\\
&\dot{x}_{29}(t) = x_5(t) + x_{13}(t) + x_{22}(t) - \alpha x_{29}(t)\\
&\dot{x}_{30}(t) = x_5(t) + x_7(t) + x_{15}(t) + x_{29}(t) - \alpha x_{30}(t)
\end{aligned}
$$

The observables are given as the output set

$$Z = \{13, 20, 7, 26, 21\} \tag{7.32}$$

and $\alpha = 4$ and the initial value by $\boldsymbol{x}_0 = 0$. To simulate a model error, we add a single additional input w_6^* to the equation for x_6, i.e.,

$$f_6^{\text{true}}(\boldsymbol{x}, t) = x_8(t) + x_{14}(t) + x_{15}(t) + x_{20}(t) + x_{30}(t) - \alpha x_6(t) + w_6^*(t)\,. \tag{7.33}$$

Furthermore we add Gaussian noise with relative standard deviation of $\sigma = 5\%$ of the current output value to the observables to simulate pseudo-experimental data $\boldsymbol{y}^{\text{data}}$. We now aim to recover the model error w_6^* of the nominal model using the pseudo-experimental data. We make no assumptions about the model error except for sparsity, thus we consider the input ground set $\mathsf{L} = \{1, \dots, 30\}$.

Detect the Active Cluster In figure 7.8(a) one can find the estimated inputs for the erroneous model described above. The estimates were computed via Δ with $\alpha = 1$ and $\beta = 0.01$. The estimated input $\hat{\boldsymbol{w}}$ is neither sparse nor does the correct component ($\hat{w}_6$) fit the ground truth w^*.

The reason for the failure of the error reconstruction lies in the high coherence between the input nodes, as presented in figure 7.8(b). One can see that there are clusters of indistinguishable nodes, whereas the distance between two clusters almost vanishes. It turns out that a number of $P = 5$ clusters is a robust choice. For the presented results, the hierarchical clustering from `python's scipy` package was used with the option *complete* which sets the distance $d(\mathsf{C}_1, \mathsf{C}_2)$ of two clusters as the maximum of $d(\mathsf{s}, \mathsf{t})$ over all $\mathsf{s} \in \mathsf{C}_1$ and $\mathsf{t} \in \mathsf{C}_2$.

Within one input cluster it is not possible to localise the source node of a model error. However, we might be able to detect the model error on the level of clusters. Figure 7.8(c) shows the influence graph of the system, the node colouring encoding the five input clusters $\mathsf{C}_1, \dots, \mathsf{C}_5$. We now compute

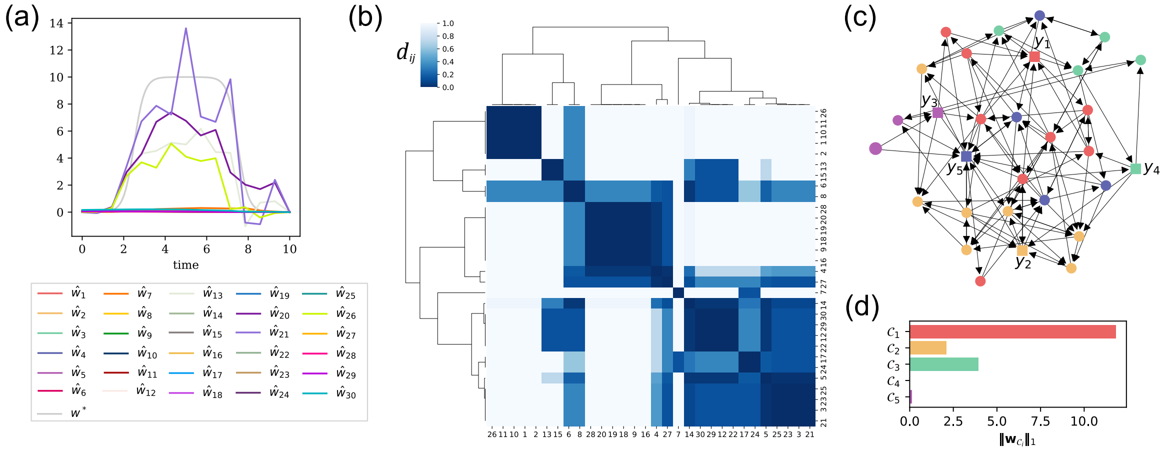

Figure 7.8: Detection of the error cluster. (a) Result of the direct application of the error estimation by minimising the cost functional. The thirty estimates $\hat{w}_i$ are shown as well as the ground truth w^*. (b) Heat map and dendrogram of the normalised distance matrix of $N = 30$ example system. For a compact illustration, the dendroram is slightly compressed, a closer quantitative analysis reveals that a clustering into five clusters is robust. (c) The influence graph of the system with node colouring indicating the affiliation of an input node to one of the five clusters. The square nodes indicate the outputs of the system. (d) The 1-norm of the partial input vector $\hat{\boldsymbol{w}}_{\mathtt{C}_i}$ which comprises the inputs attacking cluster $\mathtt{C}_i$.

the norm

$$\|\boldsymbol{w}_{\mathtt{C}_k}\|_1 = \sum_{i\in\mathtt{C}_k} \underline{w_i} \tag{7.34}$$

for each cluster. The result can be found in 7.8(d). As the overall input to cluster $\mathtt{C}_1$ is significantly larger in magnitude compared to the other clusters, we conclude that $\mathtt{C}_1$ is the *active* cluster, meaning, the cluster that is affected by (at least) one model error.

This argumentation again relies on the assumption of invariable sparsity. Without prior knowledge it is not possible to rigorously argue that, for instance, clusters $\mathtt{C}_2$ and $\mathtt{C}_3$ are affected by less severe model errors. The identification of active clusters should therefore be regarded as a heuristic approach for ill-conditioned problems, where the direct error detection does not succeed.

Sensor (Re-)Placement With the finding of the active cluster it makes sense to redefine the input ground set $\mathtt{L} := \mathtt{C}_1$. the nodes within this cluster have a high coherence which makes a more precise localisation of the model error impossible. However, for the $\Gamma = (\mathtt{C}_1, \mathtt{g}, \mathtt{Z})$ gammoid theory provides a transposed gammoid $\Gamma' = (\mathtt{Z}', \mathtt{g}', \mathtt{C}_1')$. And since we use Γ to compute the coherence of input nodes with respect to the output set, it becomes clear that we can utilise Γ' to obtain a coherence measure for output nodes with respect to the input set. So we can take the gammoid $\Gamma' = (\mathtt{M}', \mathtt{g}', \mathtt{C}_1)$ where $\mathtt{M} \subseteq \mathtt{N}$, to compute clusters of coherent output nodes (from the output ground set $\mathtt{M}$) with respect to the new input ground set $\mathtt{C}_1$. Figure 7.9(a) shows the influence graph of the system again, with the node colouring indicating clusters of coherent output nodes.

The figure unveils that the output set $\mathtt{Z}$ which has been used so far is not an optimal choice, as the observables y_2 and y_4 lie within the same output cluster, i.e., provide indistinguishable

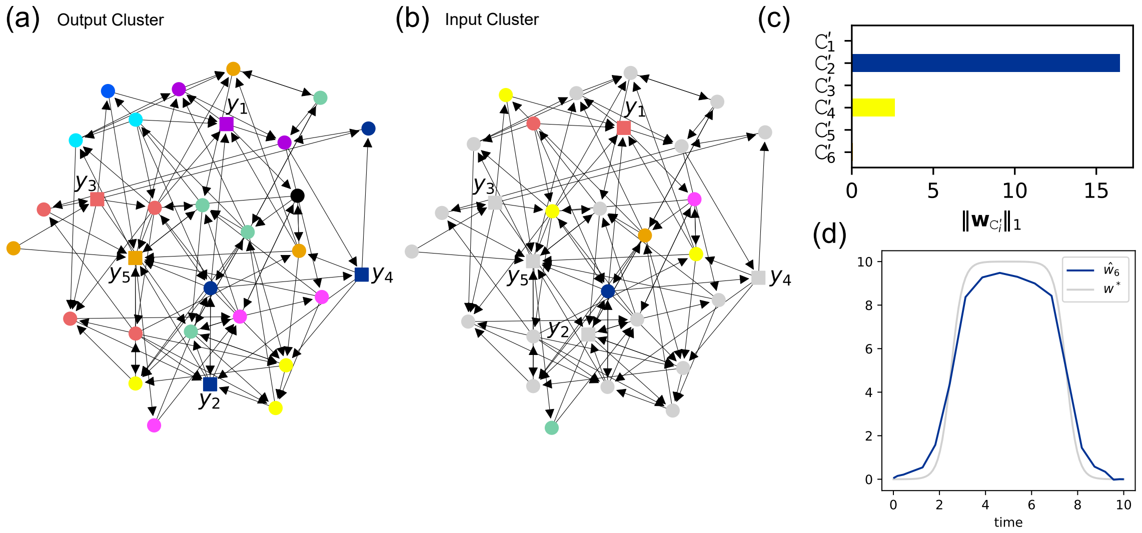

Figure 7.9: Error localisation and reconstruction in the $N = 30$ nodes system. (a) Output-clustering of the influence graph with respect to the active cluster C_1. Clusters are indicated by the node colouring. (b) After sensor replacement, an input clustering is applied to the active cluster C_1. The resulting clusters $\mathsf{C}'_1, \dots, \mathsf{C}'_2$ are indicated by the node colouring. (c) The $\|\cdot\|_1$-norms for the computed estimates of the model errors for each new cluster $\mathsf{C}'_1, \dots, \mathsf{C}'_6$. (d) The result of the error reconstruction for the active cluster C'_2

information about C_1. In order to avoid redundancy in the measurements it makes therefore sense to replace the sensor such that many output clusters are covered. Compare subfigures (a) and (b) to see that we have replaced observable y_2.

Reconstruct the Model Error Now that we have reduced the input ground set L and replaced the sensor y_2 so that the measurements provide a higher information content, we can try to infer the model error once more. In the fist step, we cluster the input ground set as shown in figure 7.9(b). Subfigure (c) shows the total input magnitude for the six input clusters $\mathsf{C}'_1, \dots, \mathsf{C}'_6$. We identify C'_2 as the active cluster. One will see that the active cluster comprises only one node which corresponds to state variable x_6. Since x_6 is indeed the node which is is perturbed by the true input w^*, we have successfully localised the model error.

Subfigure (d) shows the estimate $\hat{w}_6$. We have reconstructed the true model error w^* with a good accuracy, keeping in mind that we worked with a highly under-determined system and measurement noise.

7.3.3 Summary

The considered system is affected by a 1-sparse model error w^*. Without prior knowledge, the input ground set L contains each of the 30 nodes, but it provides only 5 observables. For this system we find a spark of 3, which actually means that a 1-sparse model is unique. However, the shortest path coherence mutual coherence is $\mu^{\text{short}} = 1$ which indicates that a high coherence between the nodes will make the error reconstruction difficult in practice. Though we were not able to reconstruct the model error directly, we computed clusters $\mathsf{C}_1, \dots, \mathsf{C}_5$ of coherent input nodes and we were able to detect C_1 as the cluster that contains the model error. Applying the clustering approach to the transposed gammoid, we computed clusters of indistinguishable output nodes. We used this information to improve the sensor placement. With the knowledge about the active cluster and the replaced sensors, we finally succeeded to localise and reconstruct the true model error.

The example described above provides an iterative strategy which combines the coherence measure and the minimisation of the cost functional in order to narrow down the location of a model error step by step. Though this strategy works remarkably well in the presented example, some comments are in turn: The identification of an active cluster is plausible but still heuristic at the current state of the theory. When we choose one active cluster, we force the inputs in the clusters of smaller magnitude to zero. This idea corresponds to *Iterative-Hard-Thresholding* algorithms which was first presented for linear, finite-dimensional operators [147] but also discussed for general Hilbert spaces [146] as well as for non-linear problems [145]. Under which conditions this kind of algorithm converges to the desired solution is an open question. A related *greedy algorithm* was published in [7] and confirms that it is capable of producing good results for real problems. Furthermore, if the true model error is non-sparse, more precisely, if its support is larger than the size of the clusters, or if the true model error contains components of small magnitude, the identification of the active cluster won't be straightforward.

CHAPTER 8

A Concluding View on SEEDS

The aim of the present work was to shed light upon the topic of Structural Error Estimation in Dynamic Systems and provide a methodology for a data driven inference. From a modern naturalist's point of view, every investigation of natural phenomena must be made with the humble confession, that there is hardly any process fully understood. The emergence of *structural model errors* of *internal* or *external* origin is the unavoidable cost of growing complexity. Therefore it is high time to develop a methodology for a contemporary treatment of model errors.

Structural Model Errors are Unavoidable Biological, chemical, or physical processes, engineered systems or machines are never isolated, but *open systems.* Clearly, exceptions exist: Experimentalists put much effort into high precision experiments to achieve a physical isolation of the system. But more often than not, the investigated system stays in contact with its environment. Be it a part of a living organism, which obviously cannot be investigated separately, or a fundamental physical system where the very act of observation already intervenes in the system's future behaviour, see for instance [158] for Heisenberg's philosophical thoughts on quantum physics. The mathematical models we use today are often implicitly, or even unknowingly, blind to such *external* model errors. The *nominal systems* we work with are *mathematically isolated.*

For a biological system it is clear that it is practically impossible to model each and every process that keeps the system intact. For a physical system it is never clear, if the understanding we have attained is really fundamental, or merely a good approximation. There remains always the possibility of *internal* model errors, misspecified laws of nature, unknown reactions, or simply erroneous parameters. Internal and external model errors change the behaviour of the system and diminish our understanding of the system and our capability to estimate its current and future development. Therefore one important objectives of this thesis was to demonstrate that all kinds of model errors, notwithstanding their origin, can be incorporated into the model as additive *unknown inputs.*

Significance of the Model Error Estimation Knowing the unknown inputs aids our understanding of the system manifoldly: The true state $\boldsymbol{x}$ of the system at a certain point in time is usually not directly accessible, but it must be estimated from the few *observables* $\boldsymbol{y}$ for which we can collect data, in combination with a good theoretical model from which we can infer the state of the system. An erroneous model leads to erroneous state estimates, and finally to a complete loss of control over the system.

In a model for a physical system the single terms of the equations represent the fundamental (meaning fundamental relative to the level one wants work with) interactions of the system. The existence of a model error can therefore indicate the need for new or revised interaction laws. It may also indicate that external perturbations which were deemed to be negligible are potentially significant. The inference of unknown inputs from given data for a system of consideration can help to build better mathematical models and by this lead to a better understanding of the laws that govern the behaviour of the system.

Both the practical issue of estimating the state of a system and the theoretical process of developing more competent models benefit from a credible model error reconstruction.

8.1 Result: A Theoretical Framework

The ways to formulate dynamic systems are only bounded by creativity, but three representations have proven convenient in the treatment of dynamic input-output system:

The formulation in *state space* is an intuitive and common understanding when one speaks of dynamic systems. A state variable $x_i(t)$ gives a numerical value, e.g., a specimen number, concentration, or voltage, at some point t in time, and the governing laws are formulated as differential equation

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t)) \quad \forall\, t \in [0,T]\,. \tag{8.1}$$

The *d-algebraic* formulation represents a dynamic system as d-polynomials

$$q_i(X_i;\boldsymbol{X}) = X_i^{(1)} - p_i(\boldsymbol{X}) \tag{8.2}$$

from a d-ring $\mathbb{K}\{\boldsymbol{X}\}$ together with a d-ideal $\Im(q_1,\ldots,q_N)$ that ensures the conformity with the governing equations. In this formulation the time resolution is omitted providing a single d-indeterminate X_i that represents the whole trajectory $x_i(t)\forall t$. It is insensitive to singular time events, such as fixed initial or final state constraints, but therefore naturally integrates the complete information of a state or observable trajectory.

Finally, the *structural* or *network* approach reduces the dynamic behaviour of a system to directed links between nodes, see figure 8.1 on the right. It lacks the information about the precise interactions, but therefore yields efficient algorithms.

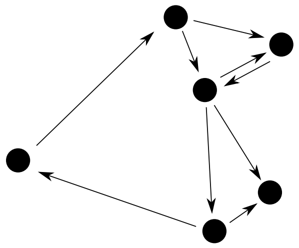

Figure 8.1: An exemplary structural system.

Invertibility The state space, d-algebraic, and structural approaches to dynamic systems can be seen as three perspectives of the same system, with focus on its geometric, algebraic and structural properties, respectively. In each representation some characteristics are obvious, others get upstaged. This phenomenon facilitated the initial formulation of the Model Error Reconstruction Problem in state space, and thus enabled the subsequent step of utilising the d-algebraic way to finally proof the almost (up to pathological cases) equivalence of analytical, d-algebraic and structural invertibility. The demonstration of the equivalence of the three invertibilities is the second important achievement of this thesis.

In the end, analytical, d-algebraic and structural invertibility provide answers to the same question: Is it possible to reconstruct the model errors from data? The three notions of invertibility are supposed to start in their own right; nevertheless a combination of them may also prove itself purposeful when intending to extract the maximum of information about a system. In the retrospective, the goal of the SEEDS project and of this thesis was to pour the different representations into one consistent theoretical framework to solve the Model Error Reconstruction Problem.

8.1.1 How to: Model Error Reconstruction

Along with the theoretical development, a methodology for practical error reconstruction evolved as a side product and shall be summarised at this point.

Structural Invertibility of Complex Networks Structural invertibility enabled us to perform a purely structure-focused investigation to understand the principles of complex networks. It has been shown that density and homogeneity support structural invertibility. Inhomogeneity in the degree-distribution, meaning the emergence of hubs and satellites, and inhomogeneity in the in- and out-degree of the hubs, meaning the emergence of super-collectors and super-spreaders, diminish the chance for a trustworthy error estimation. However, with a certain input and output placement that respects hubs and satellites of the networks, it was possible to increase the probability for structural invertibility significantly.

Real networks are often highly inhomogeneous. The first *sensor placement* respects the aforementioned network principles and helps to find an *experimental design* with enhanced invertibility behaviour.

Sparse Sensing The discovery of an intrinsic gammoid structure enabled us to formulate a sparse model error reconstruction: In a non-invertible system, there are many input vectors $\boldsymbol{u}, \boldsymbol{u}', \boldsymbol{u}'', \ldots \in \mathcal{U}$ that reproduce the data, hence serve as a candidate for a model error estimate. But in many realistic scenarios it is plausible to assume *invariable sparsity*, meaning not each and every equation of the nominal system is erroneous, but only a small number. We were able to proof that the assumption of sparse model errors leads to the *spark* which yields a lower bound ensuring a unique and by this credible error reconstruction.

Error Localisation When the idea of dependence and independence reaches its limits, we can use the idea of coherences to achieve less rigorous yet practical approximations. Under very adverse conditions the bound obtained by the *spark* is practically pointless as is can happen that uniqueness is only guaranteed for the trivial solution $\boldsymbol{u} = 0$. In such a case the *coherence of input nodes* helps to find clusters of practically indistinguishable nodes. Within one cluster, it does not make sense to search for a higher resolution of the model error. The point is that on the level of *input clusters*, we are able to distinguish *active clusters* (i.e., those which contain model errors) from inactive clusters. Admittedly we are not able to reconstruct the model error, but we can at least *localise* it.

As a dual result, the *coherence of output nodes* allows to find clusters of coherent outputs or measurements. Two sensors that lie within the same output cluster yield redundant information about the mode errors. We deduce the second *sensor placement* principle which respects the coherence of output nodes.

Error Estimation The final goal of the invertibility question is the inference of a model error estimate from given data. A generalised *Restricted-Isometry-Property* was presented together with a LASSO-type optimisation problem. For linear systems it was possible to show that the LASSO optimisation problem produces the best approximation to a sparse solution. This enables us to handle the reconstruction of model errors as a *dynamic compressed sensing* problem.

8.2 Perspectives for Future Research

Given that this thesis opens up new views on mathematical models that suffer from endogenous or exogenous model errors, the boundaries of the presented theory are open in many directions. Three concrete questions, or possibilities, for future research shall finally be mentioned.

8.2.1 Non-linear Optimisation

Non-linear Gammoids Our results on the coherence measure as well as the results on the LASSO-type optimisation are restricted to linear systems over a constant field, usually $\mathbb{R}$. The coherence measure grounds on the structure of a *weighted gammoid*. The linearity of the system ensures that the weights are independent of the current state of the system. The way towards the coherence lead over the Laplace-transformation which is defined for classical fields and therefore our results on the coherence are not directly applicable to systems over a (non-constant) d-field. It would be an interesting and relevant investigation if it is possible extend the Laplace-transformation to d-fields. This would allow to define coherence measures for a system over an arbitrary d-field.

Already [51] briefly mentioned non-constant weights in a structural system, which, for instance, would emerge from a non-linear system. It seems plausible that at least a local coherence measure might be explored for non-linear system.

Non-linear Optimisation Also the convexity of the LASSO optimisation as well as the investigation of the Restricted-Isometry-Property were restricted to the linear case. Despite this fact, we have applied our methods to a nominal Lorenz model with non-linear input-output map. This resulted in remarkably accurate estimates for the non-linearities of the Lorenz system. Treatments of non-linear systems can be found in [21] and [22], where comparable non-linear optimisation problems lead to accurate error reconstructions. Though its optimality is not yet proven for the non-linear case, it is a heuristic result that the LASSO-regularisation works well in the non-linear case.

Even for the simpler non-dynamic case, meaning compressed sensing for a map Φ between two finite dimensional vector spaces $\Phi : \mathbb{R}^N \to \mathbb{R}^P$, the case of a non-linear map is subject to current research. For instance [146, 145] discusses an *Iterative-Hard-Threshold Algorithm.* Under the *bi-Lipschitz* condition, which is closely connected to the Restricted-Isometry-Property, this iterative approach yields sparse solutions and it was shown that this algorithm converges also in the non-linear case. Whether the Iterative-Hard-Threshold Algorithm can be reformulated for dynamic systems is an open and highly relevant question, as it not only shows that dynamic compressed sensing of invariable sparse model errors is possible in non-linear systems, but also because it presents an alternative way to perform the search for an invariable sparse solution.

8.2.2 Invertibility of Partial Differential Systems

This thesis focused on the classical understanding of dynamic systems as a system of ordinary differential equations. Another import class of problems are those formulated as *partial differential equations*. Take for instance the *equation of thermal conduction* [71]

$$\dot{\psi} = \Delta\psi \tag{8.3}$$

where $\psi : (\boldsymbol{x}, t) \mapsto \psi(\boldsymbol{x}, t)$ is the temperature as a function of space and time and all constants set to one. One will believe that such a model can be erroneous as well. For instance imagine an external heat source

$$\dot{\psi} = \Delta\psi + u \tag{8.4}$$

forces the system into a different behaviour.

We notice a difficulty which was not present before: The entities ψ and $u : (\boldsymbol{x}, t) \mapsto u(\boldsymbol{x}, t)$ are functions of space and time. In the example of a room temperature it seems natural that we use a thermometer at some position $\boldsymbol{x}_0$ and measure the temperature at this place over time

$$y(t) := \psi(\boldsymbol{x}_0, t)\,. \tag{8.5}$$

Or if we take a thermal image at time t_1, we may get

$$y(\boldsymbol{x}) := \psi(\boldsymbol{x}, t_1)\,. \tag{8.6}$$

While in the case of ordinary differential equations the unknown inputs, state variables and observables are all functions of time, partial differential models have the additional quality that the observables might be fixed to one point in space (8.5) or time (8.6).

Partial d-Field for Thermal Conduction The theory of d-algebra fortunately generalises to the partial differential case [66]. So $\mathbb{K}$ can be a partial d-field if there is a family of commuting derivation operators $\Delta = (\partial_1, \partial_2, \dots, \partial_D)$, each of which fulfils the rules for d-fields.

In the heat equation example we would have the (constant) Δ-field $\mathbb{R}$ and the derivations $\Delta = (\partial_t, \partial_x, \partial_y, \partial_z)$. For simplicity we will henceforth only discuss $\Delta = (\partial_t, \partial_x)$

Now, the perturbed heat equation can be written

$$\partial_t\psi = \partial_x^2\psi + u\,. \tag{8.7}$$

Or to adapt a more algebraic style

$$q(\Psi, U) = \partial_t\Psi - (\partial_x^2\Psi + U) \tag{8.8}$$

is a d-polynomial in the Δ-field $\mathbb{R}\{\Psi, U\}$.

Influence Pattern An influence structure like $\mathtt{i} \to \mathtt{j}$ represents an equation of the form

$$\partial_t x_j(t) = \ldots + x_j(t) + \ldots \tag{8.9}$$

where the interaction is linear only for simplicity. One could say, the directed edge $\mathtt{i} \to \mathtt{j}$ is a direct influence conveyed by ∂_t,

$$\mathtt{i} \xrightarrow{\partial_t} \mathtt{j}. \tag{8.10}$$

In a partial differential context like

$$\partial_t \psi(x,t) = \partial_x \psi(x,t) \tag{8.11}$$

however, several translations into the structural representation are possible. For instance a two node system $\mathtt{N} = \{\psi, \partial_x \psi\}$ with one edge

$$\partial_x \psi \xrightarrow{\partial_t} \psi \tag{8.12}$$

or a one node system $\mathtt{N} = \{\psi\}$ with two edges

$$\psi \xrightarrow{\partial_t} \psi \quad , \quad \psi \xrightarrow{\partial_x} \psi . \tag{8.13}$$

Let us focus on the first version. Differentiation of the thermal conduction equation with respect to space and commuting the derivations yield

$$\partial_t(\partial_x \psi) = \partial_x(\partial_x \psi) , \tag{8.14}$$

which can be translated to the structure

$$\partial_x^2 \psi \xrightarrow{\partial_t} \partial_x \psi , \tag{8.15}$$

which looks like a differentiation of the structure (8.12) with respect to x.

In the same manner, one can deduce arbitrarily many edges between the nodes $\psi, \partial_x \psi, \partial_x^2 \psi, \ldots$ and the influence graph would be infinite. It seems convenient to introduce the idea of an *influence pattern*. For instance, the influence pattern of the thermal conduction with input u and an observable given by (8.5) would be

$$\mathtt{g}(x) := \begin{tikzcd} u(t,x) \arrow[r] & \psi(t,x) \arrow[r, "\delta(x-x_0)"] & y(t) \\ & \partial_x^2 \psi(t,x) \arrow[u] & \end{tikzcd} \tag{8.16}$$

with an Kronecker-delta meaning y is connected only at $x = x_0$.

Thermal Conduction as a Structural System For the influence pattern above we can derive a general formula

$$\partial_x^n \mathtt{g}(x) = \begin{tikzcd} \partial_x^n u(t,x) \arrow[r] & \partial_x^n \psi(t,x) \\ & \partial_x^{n+2} \psi(t,x) \arrow[u] \end{tikzcd} , \tag{8.17}$$

which helps us to construct the influence graph $\mathsf{G}(x)$ (with dependencies neglected)

$$\mathsf{G} = \begin{tikzcd}
 & & u \arrow[r] & \psi \arrow[r, "\delta(x-x_0)"] & y \\
\partial_x u \arrow[r] & \partial_x \psi & \partial_x^2 u \arrow[r] & \partial_x^2 \psi \arrow[u] & \\
\partial_x^3 u \arrow[r] & \partial_x^3 \psi \arrow[u] & \partial_x^4 u \arrow[r] & \partial_x^4 \psi \arrow[u] & \\
\partial_x^5 u \arrow[r] & \partial_x^5 \psi \arrow[u] & \partial_x^6 u \arrow[r] & \partial_x^6 \psi \arrow[u] & \\
\partial_x^7 u \arrow[r] & \partial_x^7 \psi \arrow[u] & \partial_x^8 u \arrow[r] & \partial_x^8 \psi \arrow[u] & \\
 & \vdots \arrow[u] & & \vdots \arrow[u] &
\end{tikzcd} . \tag{8.18}$$

One will soon realise that this system has infinitely many input nodes $u, \partial_x u, \ldots$ but only one output node. However, if we make the assumption that the external heat u is spatially constant, we find that $\partial_x u = 0$. It is then without issue to collect data only at $x = x_0$ and the system becomes invertible as it reduces to

$$\mathsf{G}(x_0) = \begin{tikzcd}
 & u(t) \arrow[r] & \psi(x_0, t) \arrow[r] & y(t) \\
\partial_x \psi & & \partial_x^2 \psi \arrow[u] & \\
\vdots \arrow[u] & & \vdots \arrow[u] &
\end{tikzcd} . \tag{8.19}$$

One will see that also a system of partial differential equations can be treated with d-algebraic and structural methods, but with issues which are not present in the case of ordinary differential equations. Namely, a more elaborated discussion of the observables is necessary as well as a treatment of infinite gammoids. It would be a significant contribution to the theory of model error reconstruction, to examine how the results presented in this thesis can be adapted to the partial differential case.

8.2.3 Model Discovery

In this thesis, we considered the case that one has a (slightly erroneous) nominal model of a system in conjunction with observation data. Then an attempt to estimate the model error is undertaken. Such a process can be viewed as a *model correction* strategy. Another approach is the discovery or *identification* of a model: Given observation data, we aim to construct a model that is simple yet reproduces the data accurately. For instance *SINDy* [33] tries to find the simplest model in the sense that the vector field of the system is sparse with respect to a given dictionary.

As both approaches aim for a precise mathematical description of the system, it might be in the focus of future research to find a complementary method: Say, we have a nominal model, data, and we have estimated the model error. Instead of starting a model identification from scratch, which would neglect all prior knowledge about the system, we first localise and estimate the model errors with the methods presented in this thesis. Then, we apply the model identification methods only to the model errors to learn about their intrinsic dynamics.

Especially in the case of large systems, a direct model identification from scratch would be computationally infeasible. The identification of the few (invariable sparse) model errors, however, is much more realistic. A combination of the two methods would be capable of correcting the vector field with functional terms and by this produce an improved and *predictive model*.

Bibliography

[1] A. von Humboldt. *Ansichten der Natur : mit wissenschaftlichen Erläuterungen /*. J.G. Cotta, Stuttgart, 3. verb. und verm. ausg. edition, 1849.

[2] P. Guyer and A. W. Wood, editors. *Critique of Pure Reason*. The Cambridge Edition of the Works of Immanuel Kant. Cambridge University Press, 1998.

[3] I. Kant. *Die drei Kritiken - Kritik der reinen Vernunft. Kritik der praktischen Vernunft. Kritik der Urteilskraft*. Anaconda Verlag, Colone, 2015. Sammelband.

[4] J. Pearl and D. Mackenzie. *The Book of Why*. Basic Books, New York, 2018.

[5] D. Kahl, P. Wendland, M. Neidhardt, A. Weber, and M. Kschischo. Structural invertibility and optimal sensor node placement for error and input reconstruction in dynamic systems. *Phys. Rev. X*, 9:041046, Dec 2019.

[6] D. Kahl, A. Weber, and M. Kschischo. Sparse Error Localization in Complex Dynamic Networks. *arXiv:2006.04694 [math]*, Jun 2020.

[7] T. Newmiwaka, B. Engelhardt, P. Wendland, D. Kahl, H. Fröhlich, and M. Kschischo. Seeds: Data driven inference of structural model errors and unknown inputs for dynamic systems biology. *Bioinformatics*, Sep 2020.

[8] `https://CRAN.R-project.org/package=seeds`, access date 2020-10-06.

[9] A. Raue, M. Schilling, J. Bachmann, A. Matteson, M. Schelke, D. Kaschek, S. Hug, C. Kreutz, B. D. Harms, F. J. Theis, U. Klingmüller, and J. Timmer. Lessons learned from quantitative dynamical modeling in systems biology. *PLOS ONE*, 8:1–17, 09 2013.

[10] A. Tsigkinopoulou, S. M. Baker, and R. Breitling. Respectful modeling: Addressing uncertainty in dynamic system models for molecular biology. *Trends in Biotechnology*, 35:518 – 529, Jan 2017.

[11] M. Almog and A. Korngreen. Is realistic neuronal modeling realistic? *Journal of Neurophysiology*, 116:2180–2209, Nov 2016.

[12] J. A. Papin, J. J. Saucerman, S. M. Peirce, K. A. Janes, P. L. Chandran, M. J. Lazzara, R. M. Ford, and D. A. Lauffenburger. An engineering design approach to systems biology. *Integrative Biology*, 9:574–583, Jun 2017.

[13] R. A. Rossi and N. K. Ahmed. The network data repository with interactive graph analytics and visualization. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, pages 4292–4293, Austin, Texas, 2015. AAAI Press.

[14] V. Batagelj and A. Mrvar. Pajek datasets. `http://vlado.fmf.uni-lj.si/pub/networks/data/`, access date 2018-12-16.

[15] J. Kunegis. Konect - the koblenz network collection. In *Proc. Int. Web Observatory Workshop*, pages 1343–1350, 2013.

[16] J. Leskovec and A. Krevl. Snap datasets: Stanford large network dataset collection. `http://snap.stanford.edu/data`, access date 2018-12-16.

[17] H. Haller. *Zur Ökologie des Luchses Lynx lynx im Verlauf seiner Wiederansiedlung in den Walliser Alpen.* P. Parey, Hamburg, 1992.

[18] U. Breitenmoser, A. Ryser, A. Moliari-Jobin, F. Zimmermann, H. Haller, P. Molinari, and C. Breitenmoser-Würsten. The changing impact of predation as a source of conflict between hunters and reintroduced lynx in Switzerland. In *Biology and Conservation of Wild Felids*, pages 493–505, Oxford, 2010. Oxford University Press.

[19] U. Breitenmoser and M. Baettig. Wiederansiedlung und ausbreitung des luchses (lynx lynx) im schweizer jura. *Revue suisse de Zoologie*, 122:163–176, 1992.

[20] A. Molinari-Jobin, F. Zimmermann, A. Ryser, C. Breitenmoser-Würsten, S. Capt, U. Breitenmoser, P. Molinari, H. Haller, and R. Eyholzer. Variation in diet, prey selectivity and home-range size of Eurasian lynx Lynx lynx in Switzerland. *Wildlife Biology*, 13:393–405, December 2007.

[21] B. Engelhardt, H. Fröhlich, and M. Kschischo. Learning (from) the errors of a systems biology model. *Scientific Reports*, 6, Nov 2016.

[22] B. Engelhardt, M. Kschischo, and H. Fröhlich. A Bayesian approach to estimating hidden variables as well as missing and wrong molecular interactions in ordinary differential equation-based mathematical models. *Journal of The Royal Society Interface*, 14:20170332, June 2017.

[23] J. J. O'Connor and E. F. Robertson. Mathematical discovery of planets. `https://mathshistory.st-andrews.ac.uk/HistTopics/Neptune_and_Pluto/`,access date 2020-07-07, Sep 1996.

[24] I. Netwon. Isaac Newton: Principia. `http://www.17centurymaths.com/contents/newtoncontents.html`, access date 2020-07-07. translated and annotated by Ian Bruce.

[25] G. C. Layek. *An Introduction to Dynamical Systems and Chaos.* Springer India, 1st ed. 2015 edition, 2015.

[26] H. N. Salas. Gershgorin's theorem for matrices of operators. *Linear Algebra and its Applications*, 291:15–36, Apr 1999.

[27] D. G. Luenberger. *Introduction to Dynamic Systems: Theory, Models, and Applications.* Wiley, Hoboken, New Jersey, May 1979.

[28] M. E. Lindelöf. Sur l'application des méthodes d'approxiations successives à l'étude des intégrales réeles des équations différentielles ordinaires. *Journal des Mathématiques pures et appliquées*, 4:117–128, 1894.

[29] E. Picard. Sur l'application des méthodes d'approxiations successives à l'étude des certaines équations différentielles ordinaires. *Journal des Mathématiques pures et appliquées*, 4:217–272, 1893.

[30] L. A. Aguirre, L. L. Leonardo, and C. Letellier. Structural, dynamical and symbolic observability: From dynamical systems to networks. *PLoS One*, 13:e0206180, Oct 2018.

[31] `http://svgsilh.com`, access date 2020-10-02.

[32] D. D. Mordykhai-Boltovskoi. On hypertranscendence of the function ξ (x, s). *Izv. Politekh. Inst. Warsaw*, 2:1–16, 1914.

[33] S. L. Brunton, J. L. Proctor, and J. N. Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences of the United States of America*, 113:3932–3937, Apr 2016.

[34] D. J. Mook and J. L. Junkins. Minimum model error estimation for poorly modeled dynamic systems. *Journal of Guidance, Control, and Dynamics*, 11:256–261, May 1988.

[35] J. Lunze. Einführung in die Mehrgrößenregelung. In *Regelungstechnik 2: Mehrgrößensysteme, Digitale Regelung*, Heidelberg, 2016. Springer Berlin Heidelberg.

[36] L. Silverman. Inversion of multivariable linear systems. *IEEE Transactions on Automatic Control*, 14:270–276, Jun 1969.

[37] E. D. Sontag. Mathematical Control Theory. In *Texts in Applied Mathematics*, volume 6, New York, New York, 1998. Springer New York.

[38] M. Fliess. A note on the invertibility of nonlinear input-output differential systems. *Systems & Control Letters*, 8:147–151, December 1986.

[39] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki. *Diagnosis and Fault-Tolerant Control*. Springer, Heidelberg, Germany, 2016.

[40] P. Kühl, M. Diehl, T. Kraus, J. P. Schlöder, and H. G. Bock. A real-time algorithm for moving horizon state and parameter estimation. *Computers & Chemical Engineering*, 35:71–83, Jan 2011.

[41] R. Fonod, D. Henry, C. Charbonnel, and E. Bornschlegl. A class of nonlinear unknown input observer for fault diagnosis: Application to fault tolerant control of an autonomous spacecraft. In *2014 UKACC International Conference on Control (CONTROL)*, pages 13–18, Loughborough, UK, Jul 2014. IEEE.

[42] A. Chakrabarty, M. J. Corless, G. T. Buzzard, S. H. Zak, and A. E. Rundell. State and Unknown Input Observers for Nonlinear Systems With Bounded Exogenous Inputs. *IEEE Transactions on Automatic Control*, 62:5497–5510, Nov 2017.

[43] R. W. Brockett and M. D. Mesarović. The reproducibility of multivariable systems. *Journal of Mathematical Analysis and Applications*, 11:548–563, Jan 1965.

[44] M. Sain and J. Massey. Invertibility of linear time-invariant dynamical systems. *IEEE Transactions on Automatic Control*, 14:141–149, Apr 1969.

[45] H. Nijmeijer. Invertibility of affine nonlinear control systems: A geometric approach. *Systems & Control Letters*, 2:163–168, Oct 1982.

[46] H. Nijmeijer. Right-invertibility for a class of nonlinear control systems: A geometric approach. *Systems & Control Letters*, 7:125–132, Apr 1986.

[47] R. M. Hirschorn. Invertibility of Nonlinear Control Systems. *SIAM Journal on Control and Optimization*, 17:289–297, Mar 1979.

[48] A. Martinelli. Nonlinear Unknown Input Observability: The General Analytic Solution. *IEEE Transactions on Automatic Control*, 64:222–237, Jan 2019.

[49] M. Fliess. Nonlinear Control Theory and Differential Algebra: Some Illustrative Examples. In *10th Triennial IFAC Congress on Automatic Control*, volume 20, pages 103–107. Elsevier, Jul 1987.

[50] R. Isermann. *Fault-diagnosis applications*. Springer, Heidelberg, 2011.

[51] T. Wey. Rank and Regular Invertibility of Nonlinear Systems: A Graph Theoretic Approach. *IFAC Proceedings Volumes*, 31:257–262, Jul 1998.

[52] J.-M. Dion, C. Commault, and J. van der Woude. Generic properties and control of linear structured systems: a survey. *Automatica*, 39:1125–1144, Jul 2003.

[53] T. Boukhobza, Frédéric H., and S. Martinez-Martinez. State and input observability for structured linear systems: A graph-theoretic approach. *Automatica*, 43:1204–1210, Jul 2007.

[54] G. Basile and G. Marrq. A new characterization of some structural properties of linear systems: unknown-input observability, invertibility and functional controllability. *International Journal of Control*, 17:931–943, May 1973.

[55] M. Schelker, A. Raue, J. Timmer, and C. Kreutz. Comprehensive estimation of input signals and dynamics in biochemical reaction networks. *Bioinformatics*, pages i529–i534, Sep 2012.

[56] N. Tsiantis, E. Balsa-Canto, and J. R. Banga. Optimality and identification of dynamic models in systems biology: an inverse optimal control framework. *Bioinformatics*, 34:2433–2440, Jul 2018.

[57] J. A. Moreno, E. Rocha-Cózatl, and A. V. Wouwer. A dynamical interpretation of strong observability and detectability concepts for nonlinear systems with unknown inputs: application to biochemical processes. *Bioprocess and Biosystems Engineering*, 37:37–49, Jan 2014.

[58] J. Moreno, E. Rocha-Cozatl, and A. Vande Wouwer. Observability/detectability analysis for nonlinear systems with unknown inputs - Application to biochemical processes. In *2012 20th Mediterranean Conference on Control and Automation, MED 2012 - Conference Proceedings*, pages 151–156. IEEE, Jul 2012.

[59] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67:301–320, 2005.

[60] E. J. Candes, J. Romberg, and T. Tao. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52:489–509, Feb 2006.

[61] E. J. Candes and T. Tao. Decoding by linear programming. *IEEE Transactions on Information Theory*, 51:4203–4215, Dec 2005.

[62] D. L. Donoho and P. B. Stark. Uncertainty Principles and Signal Recovery. *SIAM Journal on Applied Mathematics*, 49:906–931, Jun 1989.

[63] J. Hadamard. Sur les problèmes aux dérivées partielles et leur signification physique. *Princeton University Bulletin*, 13:49–52, 1902.

[64] H. Heuser. *Funktionalanalysis: Theorie und Anwendung*. Mathematische Leitfäden. Vieweg+Teubner Verlag, Wiesbaden, 4 edition, 2006.

[65] R. Recorde. *The Whetstone of Witte*. Ihon Kyngston, London, 1557. `http://archive.org/details/TheWhetstoneOfWitte`, access date 2020-07-08.

[66] J. F. Ritt. *Differential Algebra*. American Mathematical Society, Providence, Rhode Island, 1950.

[67] E. R. Kolchin. *Differential Algebra & Algebraic Groups*, volume 54. Academic Press, Cambridge, Massachusetts, 1 edition, 1973.

[68] I. Kaplansky. *An Introduction to Differential Algebra*. Actualités scientifiques et industrielles. Hermann, Paris, 1957.

[69] M. Fliess. Nonlinear control theory and differential algebra. In *Modelling and Adaptive Control*, Lecture Notes in Control and Information Sciences, pages 134–145, Berlin, Heidelberg, 1988. Springer.

[70] W. Nolting. *Theoretical Physics 5: Thermodynamics*. Springer International Publishing, Cham, 2017.

[71] L. D. Landau and E. M. Lifshitz. *Fluid Mechanics: Volume 6*. Butterworth-Heinemann, Amsterdam, 2 edition, 1987.

[72] D. J. Griffiths. *Introduction to Electrodynamics*. Cambridge University Press, Cambridge, 4 edition, 2017.

[73] D. Carothers, G. Parker, J. Sochacki, and P. Warne. Some properties of solutions to polynomial systems of differential equations. *Electronic Journal of Differential Equations*, 04 2005.

[74] S. Lang. *Algebra.* Graduate Texts in Mathematics. Springer-Verlag, New York, 3 edition, 2002.

[75] C.-T. Lin. Structural controllability. *IEEE Transactions on Automatic Control*, 19:201–208, Jun 1974.

[76] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási. Controllability of complex networks. *Nature*, 473:167–173, May 2011.

[77] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási. Observability of complex systems. In *Proceedings of the National Academy of Sciences*, volume 110, pages 2460–2465, Washigton, D.C., Feb 2013. United States National Academy of Sciences.

[78] J. Gao, Y.-Y. Liu, R. M. D'Souza, and A.-L. Barabási. Target control of complex networks. *Nature Communications*, 5:5415, Nov 2014.

[79] A. E. Motter. Networkcontrology. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 25:097621, Sep 2015.

[80] Y.-Y. Liu and A.-L. Barabási. Control principles of complex systems. *Reviews of Modern Physics*, 88:035006, Sep 2016.

[81] H. D. I. Abarbanel. *Predicting the future: completing models of observed complex systems.* Understanding complex systems. Springer, New York, NY, 2013.

[82] J. Sun and A. E. Motter. Controllability transition and nonlocality in network control. *Physical Review Letters*, 110:208701, May 2013.

[83] S. P. Cornelius, W. L. Kath, and A. E. Motter. Controlling complex networks with compensatory perturbations. *arXiv:1105.3726 [cond-mat, physics:physics, q-bio]*, May 2011.

[84] G. Yan, G. Tsekenis, B. Barzel, J.-J. Slotine, Y.-Y. Liu, and A.-L. Barabási. Spectrum of controlling and observing complex networks. *Nature Physics*, 11:779–786, Sep 2015.

[85] S. P. Cornelius, W. L. Kath, and A. E. Motter. Realistic control of network dynamics. *Nature Communications*, 4:1942, Dec 2013.

[86] T. H. Summers, F. L. Cortesi, and J. Lygeros. On Submodularity and Controllability in Complex Dynamical Networks. *IEEE Transactions on Control of Network Systems*, 3:91–101, Mar 2016.

[87] M. Newman. *Networks: An Introduction.* Oxford University Press, Oxford, Mar 2010.

[88] K. Menger. Zur allgemeinen kurventheorie. *Fundamenta Mathematicae*, 10:96–115, 1927.

[89] R. E. Kalman. *Contributions to the Theory of Optimal Control*, volume 5. Sociedad Matematica Mexicana, 1960.

[90] R. E. Kalman. Mathematical Description of Linear Dynamical Systems. *Journal of the Society for Industrial and Applied Mathematics Series A Control*, 1:152–192, Jan 1963.

[91] P. Erdős and A. Rényi. On random graphs i. *Publicationes Mathematicae Debrecen*, 6:290–297, 1959.

[92] A.-L. Barabási and Albert R. Emergence of scaling in random networks. *Science*, 286:509–512, Oct 1999.

[93] P. Erdős and Rényi A. On the evolution of random graphs. *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, pages 17–60, 1960.

[94] K.-I. Goh, B. Kahng, and D. Kim. Universal Behavior of Load Distribution in Scale-Free Networks. *Physical Review Letters*, 87:278701, Dec 2001.

[95] B. Korte and J. Vygen. *Combinatorial optimization: theory and algorithms*. Algorithms and combinatorics. Springer, Berlin, sixth edition edition, 2018.

[96] A. V. Goldberg and R. E. Tarjan. A new approach to the maximum-flow problem. *Journal of the ACM*, 35:921–940, Oct 1988.

[97] R. Cohen and S. Havlin. Scale-Free Networks Are Ultrasmall. *Physical Review Letters*, 90:058701, Feb 2003.

[98] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs: Simple building blocks of complex networks. *Science*, 298:824–827, 2002.

[99] U. Alon. Collection of complex networks. `http://www.weizmann.ac.il/mcb/UriAlon/download/collection-complex-networks`, access date 2018-12-16.

[100] K. Norlen, G. Lucas, M. Gebbie, and J. Chuang. EVA: Extraction, Visualization and Analysis of the Telecommunications and Media Ownership Network. In *Proceedings of International Telecommunications Society 14th Biennial Conference (ITS2002)*, pages 27–129. n.p., 2002.

[101] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney. Community Structure in Large Networks: Natural Cluster Sizes and the Absence of Large Well-Defined Clusters. *Internet Mathematics*, 6:29–123, Jan 2009.

[102] M. Richardson, R. Agrawal, and P. Domingos. Trust management for the semantic web. In *The Semantic Web - ISWC 2003*, pages 351–368, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.

[103] N. D. Martinez. Artifacts or Attributes? Effects of Resolution on the Little Rock Lake Food Web. *Ecological Monographs*, 61:367–392, Feb 1991.

[104] Mount Sinai Ma'ayan Laboratory Department of Pharmacology and Systems Therapeutics. `http://research.mssm.edu/maayan/datasets/qualitative_networks.shtml`, access date 2018-12-16.

[105] J. Schellenberger, J. O. Park, T. M. Conrad, and B. Ø. Palsson. BiGG: a Biochemical Genetic and Genomic knowledgebase of large scale metabolic reconstructions. *BMC Bioinformatics*, 11:213, Apr 2010.

[106] BiGG Models. `http://bigg.ucsd.edu`, access date 2018-12-16.

[107] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440, Jun 1998.

[108] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining - KDD '05*, page 177, New York City, 2005. Association for Computing Machinery.

[109] L. A. Adamic and N. Glance. The political blogosphere and the 2004 U.S. election: divided they blog. In *Proceedings of the 3rd international workshop on Link discovery - LinkKDD '05*, pages 36–43, New York City, 2005. Association for Computing Machinery.

[110] Datasets. `http://moreno.ss.uci.edu/data.html#blogs`, access date 2018-12-16.

[111] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graph evolution: Densification and shrinking diameters. *ACM Transactions on Knowledge Discovery from Data*, 1:2–es, Mar 2007.

[112] T. Opsahl and P. Panzarasa. Clustering in weighted networks. *Social Networks*, 31:155–163, May 2009.

[113] Datasets. `https://toreopsahl.com/datasets/`, access date 2018-12-16.

[114] H. Ebel, L.-I. Mielsch, and S. Bornholdt. Scale-free topology of e-mail networks. *Physical Review E*, 66, Sep 2002.

[115] R. L. Cross and A. Parker. *The hidden power of social networks: understanding how work really gets done in organizations*. Harvard Business School Press, Boston, Massachusetts, 2004.

[116] Y. C. Eldar and G. Kutyniok, editors. *Compressed sensing: theory and applications*. Cambridge University Press, Cambridge, New York, 2012.

[117] D. L. Donoho, M. Elad, and V. N. Temlyakov. Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Transactions on Information Theory*, 52:6–18, Jan 2006.

[118] S. Foucart and H. Rauhut. *A Mathematical Introduction to Compressive Sensing*. Applied and Numerical Harmonic Analysis. Birkhauser Boston Inc., Cambridge, Massachusetts, 2013.

[119] D. L. Donoho and M. Elad. Optimally sparse representation in general (nonorthogonal) dictionaries via l1 minimization. *Proceedings of the National Academy of Sciences*, 100:2197–2202, Mar 2003.

[120] N. Challapalli, M. Nagahara, and M. Vidyasagar. Continuous Hands-off Control by CLOT Norm Minimization. *IFAC-PapersOnLine*, 50:14454–14459, Jul 2017.

[121] H. Whitney. On the Abstract Properties of Linear Dependence. *American Journal of Mathematics*, 57:509–533, Jul 1935.

[122] K. Murota. *Matrices and Matroids for Systems Analysis*. Springer-Verlag Berlin Heidelberg, Heidelberg, 2009.

[123] D. J. A. Welsh. *Matroid Theory*. Academic Press, Cambridge, Massachusetts, 1976.

[124] J. J. Cho, Y. Chen, and Y. Ding. On the (co)girth of a connected matroid. *Discrete Applied Mathematics*, 155:2456–2470, Nov 2007.

[125] H. Perfect. Applications of Menger's graph theorem. *Journal of Mathematical Analysis and Applications*, 22:96–111, Apr 1968.

[126] J. S. Pym. The Linking of Sets in Graphs. *Journal of the London Mathematical Society*, s1-44:542–550, Jan 1969.

[127] H. Perfect. Independence Spaces and Combinatorial Problems. *Proceedings of the London Mathematical Society*, s3-19:17–30, Jan 1969.

[128] J. S. Pym. A proof of the linkage theorem. *Journal of Mathematical Analysis and Applications*, 27:636–638, September 1969.

[129] A. W Ingleton and M. J Piff. Gammoids and transversal matroids. *Journal of Combinatorial Theory, Series B*, 15:51–68, Aug 1973.

[130] J. H. Mason. On a Class of Matroids Arising From Paths in Graphs. *Proceedings of the London Mathematical Society*, s3-25:55–74, Jul 1972.

[131] S. Boyd, S.P. Boyd, and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, 2004.

[132] S. A. Gershgorin. Über die Abgrenzung der Eigenwerte einer Matrix. *Bulletin de l'Académie des Sciences de l'URSS. VII. Série*, 1931:749–754, 1931.

[133] Z. Jiang and Q. Hu. The complexity results of the sparse optimization problems and reverse convex optimization problems. *Optimization Letters*, Feb 2020.

[134] F. Santosa and W. W. Symes. Linear Inversion of Band-Limited Reflection Seismograms. *SIAM Journal on Scientific and Statistical Computing*, 7:1307–1330, Oct 1986.

[135] R. Tibshirani. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58:267–288, 1996.

[136] J. Bergh and J. Lofstrom. *Interpolation Spaces: An Introduction*. Grundlehren der mathematischen Wissenschaften. Springer, Berlin Heidelberg, 1976.

[137] D. L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52:1289–1306, Apr 2006.

[138] E. J. Candès. The restricted isometry property and its implications for compressed sensing. *Comptes Rendus Mathematique*, 346:589–592, May 2008.

[139] J. Chen and R. J. Patton. *Robust Model-Based Fault Diagnosis for Dynamic Systems*. The International Series on Asian Studies in Computer and Information Science. Springer US, 1999.

[140] E. N. Lorenz. Deterministic Nonperiodic Flow. *Journal of the Atmospheric Sciences*, 20:130–141, Mar 1963.

[141] B. Saltzman. Finite Amplitude Free Convection as an Initial Value Problem—I. *Journal of the Atmospheric Sciences*, 19:329–341, Jul 1962.

[142] T. Haase. *Anforderungen an eine durch erneuerbare Energien geprägte Energieversorgung: Untersuchung des Regelverhaltens von Kraftwerken und Verbundnetzen.* 2006. PhD Thesis, University of Rostock.

[143] `https://www.freevector.com/map-of-europe#`, access date 2020-10-05.

[144] T. H. Summers, F. L. Cortesi, and J. Lygeros. On submodularity and controllability in complex dynamical networks. *IEEE Transactions on Control of Network Systems*, 3(1):91–101, 2016.

[145] T. Blumensath. Compressed sensing with nonlinear observations and related nonlinear optimization problems. *IEEE Transactions on Information Theory*, 59:3466–3474, 2013.

[146] T. Blumensath. Sampling and reconstructing signals from a union of linear subspaces. *IEEE Transactions on Information Theory*, 57:4660–4671, 2011.

[147] T. Blumensath and M. E. Davies. Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis*, 27:265–274, Nov 2009.

[148] R. Q. Quiroga and S. Panzeri, editors. *Principles of Neural Coding.* CRC Press, 0 edition, May 2013.

[149] F. Emmert-Streib, Z. Yang, H. Feng, S. Tripathi, and M. Dehmer. An Introductory Review of Deep Learning for Prediction Models With Big Data. *Frontiers in Artificial Intelligence*, 3, Feb 2020.

[150] R. T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud. Neural Ordinary Differential Equations. *arXiv:1806.07366 [cs, stat]*, Jun 2018. [Preprint].

[151] WormBook. `http://www.wormbook.org`, access date 2020-09-10.

[152] Z. F. Altun, L. A. Herndon, C. A. Wolkow, C. Crocker, R. Lints, and D.H. Hall (eds.). Wormatlas. 2002-2020.

[153] L. R. Varshney, B. L. Chen, E. Paniagua, D. H. Hall, and D. B. Chklovskii. Structural Properties of the Caenorhabditis elegans Neuronal Network. *PLOS Computational Biology*, 7, Feb 2011.

[154] B. L. Chen, D. H. Hall, and D. B. Chklovskii. Wiring optimization can relate neuronal structure and function. *Proceedings of the National Academy of Sciences*, 103:4723–4728, Mar 2006.

[155] G. Yan, P. E. Vértes, E. K. Towlson, Y. L. Chew, D. S. Walker, W. R. Schafer, and A.-L. Barabási. Network control principles predict neuron function in the Caenorhabditis elegans connectome. *Nature*, 550(7677):519–523, Oct 2017.

[156] G. J. Stephens, B. Johnson-Kerner, W. Bialek, and W. S. Ryu. Dimensionality and Dynamics in the Behavior of C. elegans. *PLOS Computational Biology*, 4(4):e1000028, Apr 2008.

[157] A. K. Corsi, B. Wightman, and M. Chalfie. A transparent window into biology: A primer on *Caenorhabditis elegans*. *WormBook*, Jun 2015.

[158] W. Heisenberg and J. Busche. *Quantentheorie und Philosophie: Vorlesungen und Aufsätze*. Reclams Universal-Bibliothek. Reclam, Stuttgart, 1979.

List of Figures

1.1 A visualisation of structural model errors. (a) A known subsystem (blue shaded region) is embedded into a larger unknown exo-system (grey shaded region). The subsystem interacts with the exo-system as indicated by the arrows. In addition, there exists one interaction (green dotted arrow) between two elements of the subsystem that leaves the known (blue) region. (b) A reformulation with model errors (red wiggly arrows). The interaction with the exo-system leads to an exogenous model error, the unknown interaction within the model leads to an endogenous model error. 3

1.2 A vicinity of a tree becomes a nominal system. Public domain (🅭∅) found at [31] were used. (a) A schematic illustration of the multiple state variables z_i of the complex system. We are interested in the vicinity of a single forest, illustratively described by the variables z_1, z_2, z_3. The boundary of the subsystem are drawn basing on our intuitive understanding, however, without mathematical reason. Furthermore, the boundary is an imaginary line but not a physical barrier. Hence, the interactions with the other state variables are not affected. (b) In the nominal model, the tree appears to be a standalone system. 9

1.3 An exemplary model error reconstruction. (a) The considered system. Each node x_i represents one state variable, each arrow represents a linear interaction with coefficient one. The wiggly arrows indicate the additive inputs w_i, the square nodes represents the measured outputs y_i. The initial value is set to zero, $\boldsymbol{x}(0) = 0$. (b) The pseudo-experimental data of the system (a) and the corrected model with inputs from (c, dotted lines) and (d, solid lines). (c) One choice of input functions applied as w_1, w_2, w_3 to the systems (a) and (e). The corresponding trajectory of the system is shown as dotted lines in (b) and (f). (d) Another choice of input functions applied to the systems (a) and (e). The corresponding trajectory of the system is shown as solid lines in (b) and (f). (e) A variation of the system (a). It differs in the output y_3. (f) The pseudo-experimental data of the system (f) and the corrected model with inputs from (c, dotted lines) and (d, solid lines). 13

3.1 Structural invertibility analysis for the two systems from figure 1.3. (a) A system with three inputs w_1, w_2, w_3 and three outputs y_1, y_2, y_3. All paths from w_2 and w_3 to the outputs intersect at node x_9. A node-disjoint linking of inputs and outputs is not possible and the system structurally non-invertible. (b) After replacing the output y_3, there exists a node-disjoint linking as indicated by the colouring. The system is structurally invertible. 40

3.2 The influence graph of the exemplary system 3.20 to demonstrate the propagation of a d-transcendence basis. 41

4.1 The transformation from influence graph g to flow graph G for an exemplary system. 51

4.2 The probability ρ for an invertible ER- or SF-network with random input and output sets as a function of the average degree $\bar{d}$. The network size is fixed to $N = 1000$, the input and output sets have a size of M. For each value of $\bar{d}$, ρ is computed as the relative number of structurally invertible networks among 100 network realisations. Each data point has an empirical standard deviation of $\sqrt{N\rho(1-\rho)}$ which is not plotted for better readability. (a) ER-networks. (b) SF-networks with a scaling parameter $\gamma = 2.8$. 53

4.3 The probability that an SF-network is structurally invertibility under preferential input and output placement. For each average node degree $\overline{d}$, ρ is the relative number of structurally invertible networks among 100 realisations of Scale-Free networks with $N = 1000$ nodes and $\gamma = 2.8$, where the input and output sets of size $M = 10$ were drawn uniformly. For ρ_{HubsIn} and ρ_{HubsOut} we took the 50 nodes with the highest node degree and randomly drew a subset of size $M = 10$ as input and output set, respectively. The output and input set, respectively, were drawn randomly from the remaining nodes of the network. 56

4.4 Convergence of the Sensor Node Placement Algorithm. The size P of the current output set as a function of the number of iterations for the satellite-deletion, hub-deletion and random-deletion versions of the algorithm for an exemplary ER-system with $N = 1000$ and $M = 100$ inputs. In this example, the minimal number of iterations is 500 and the maximal number of iterations 600. 57

4.5 Enhanced structural invertibility of real networks. (a) The probability ρ that the networks from table 4.1 are structurally invertible. The input set of size M is drawn randomly, the output set is obtained by the Sensor Node Placement Algorithm. For each M, ρ is the relative number of structurally invertible networks among 100 realisations. Each value is equipped with a standard deviation of $\sqrt{N\rho(\rho-1)}$ which s again not plotted for better readability. (b) The difference in the probabilities between the results of scenario I, see figure 4.6 (a) and scenario II, see subfigure (a). 58

4.6 Structural invertibility of real networks. (a) The probability ρ that the networks from table 4.1 with randomly placed inputs and outputs is structurally invertible as a function of the size M of the input and output sets. For each value of M one hundred realisations of input and output sets were simulated and ρ gives the fraction of structurally invertible networks. Each data point has an empirical standard deviation of $\sqrt{N\rho(\rho-1)}$ which is not plotted for better readability. (b) The probability from (a) is plotted against the probability computed analogously after a Degree Preserving Randomisation. 60

4.7 Impact of asymmetry. (a) The probability for structural invertibility for the *E.coli* network and a randomly generated SF-network with the same properties. Analogous to scenario I, see figure 4.6, for each M, ρ is the relative number of structurally invertible systems among 100 realisations. (b) The in- and out-degrees of the nodes of the *E.coli* network. (c) The in- and out-degrees of the nodes of a randomly generated Scale-Free network. 61

4.8 A comparison of the probability for structural invertibility of real networks for the three considered scenarios. 62

5.1 Example for a successful model error reconstruction. (a) The influence graph of an exemplary system of size $N = 30$. Each node represents one state variable. The black squares represent the $P = 10$ state variables that are considered as the observables $y_1, \dots, y_{10}$. The red wiggly arrow indicates that an additional input w^*, representing a model error, affects state variable x_6. (b) Pseudo-experimental and predictions of the augmented model with the input shown in (c) (c) The estimated input vector $\hat{\boldsymbol{w}} = (\hat{w}_1, \dots, \hat{w}_{30})^T$ is shown together with the ground truth w^*. 67

5.2 Construction of the Gramian gammoid. (a) An exemplary gammoid Γ. The red round nodes are the input set, the blue squared nodes represent the outputs. (b) The transposed gammoid Γ'. The round blue nodes are the output set and the red squares represent the input nodes. (c) The outputs of Γ are identified with the inputs of Γ'. We obtain the Gramian gammoid $\Gamma \circ \Gamma'$ with inputs indicated in red and outputs indicated in blue. 86

5.3 The result of the numerical inverse Laplace-transform of the coherence of the exemplary 2-in-2-out system. 94

7.1 A map of the European power grid UCTE, status as of summer 2004. The nodes represent large power and transformer stations as presented in [142]. Each edge represents a direct connection in both directions, for a better visualisation represented by a single undirected edge. 🅭🅯 *Map of Europe* by albertoalvarez.es, source [143], image section and colour adjusted. Graph illustrating the UCTE as presented in [142]. 128

7.2 The influence graph of the linearised Lorenz system. 130

7.3 Result of the error reconstruction for the linearised Lorenz model with data transformation over a time range $[0,1]$. One can see that the error estimates $\hat{u}_i$ fit the ground truth w_i^* between $t = 0$ and $t \approx 0.5$. For larger t, $\hat{u}_1$ and $\hat{u}_2$ differ significantly from the ground truth. 131

7.4 Trajectories of the Lorenz system before and after data transformation. (a) The trajectory before data transformation over the time range $[0,15]$. One can identify the well known Lorenz-attractor. (b) The trajectory after data transformation over the time range $[0,15]$. The trajectory is unbounded as time increases. (c) The three state variables before data transformation over a smaller time range. The values stay in a bounded range. (d) The three state variables after data transformation. The absolute value of the state variables x and y grow over limitless as the time exceeds $t \approx 0.5$. . 132

7.5 Error reconstruction for the linearised Lorenz system without data transformation over the time range $[0,15]$. (a) Estimate $\hat{u}_1$ and ground truth w_1^*. (a) Estimate $\hat{u}_2$ and ground truth w_2^*. (a) Estimate $\hat{u}_3$ and ground truth w_3^*. 133

7.6 A photo and a schematic illustration of an *C.elegans* individual of hermaphrodite sex. 🅭🅯 2015 *Life Cycle of C.elegans* and *C.elegans anatomy* by Ann K. Korsi, Bruce Wightman, and Martin Chalfie, source and information [157]. The figure shows image sections of the original. (a) A photo of an adult *C.elegans* individual. (b) A schematic illustration of the worm's physique. 135

7.7 Neural structure of hermaphrodite *C.elegans*. (a) The neuronal network in a simplistic 3-layer illustration. The data set, published in [153] and online available [152], comprises 283 nodes comprising 82 input neuron (red nodes, top layer), 93 inter neurons (black nodes, middle layer), and 108 motor neurons (blue nodes, bottom layer). Furthermore there are 4693 synaptic links (grey arrows). The real neural network does not have a strict cascade structure, i.e., there are direct edges from the top to the bottom layer as well as edges directed upwards. (b) Heat map of the normalised distance matrix for the 82 sensor neurons. For the pairwise distance d_{ij} an upper bound of $d_{\max} = 10$ was introduced. 136

7.8 Detection of the error cluster. (a) Result of the direct application of the error estimation by minimising the cost functional. The thirty estimates $\hat{w}_i$ are shown as well as the ground truth w^*. (b) Heat map and dendrogram of the normalised distance matrix of $N = 30$ example system. For a compact illustration, the dendroram is slightly compressed, a closer quantitative analysis reveals that a clustering into five clusters is robust. (c) The influence graph of the system with node colouring indicating the affiliation of an input node to one of the five clusters. The square nodes indicate the outputs of the system. (d) The 1-norm of the partial input vector $\hat{\boldsymbol{w}}_{\mathsf{C}_i}$ which comprises the inputs attacking cluster C_i. 138

7.9 Error localisation and reconstruction in the $N = 30$ nodes system. (a) Output-clustering of the influence graph with respect to the active cluster C_1. Clusters are indicated by the node colouring. (b) After sensor replacement, an input clustering is applied to the active cluster C_1. The resulting clusters $\mathsf{C}'_1, \ldots, \mathsf{C}'_2$ are indicated by the node colouring. (c) The $\|\cdot\|_1$-norms for the computed estimates of the model errors for each new cluster $\mathsf{C}'_1, \ldots, \mathsf{C}'_6$. (d) The result of the error reconstruction for the active cluster C'_2 139

8.1 An exemplary structural system. 142

List of Tables

4.1 A compilation of networks from various fields, also examined by other authors ([77]). Here N is the number of nodes and $|\mathtt{E}|$ the number of edges. The column *SF* indicates whether the degree distribution shows a power law and if so, the power law exponent γ was computed. 59