Structural Invertibility and Optimal Sensor Node Placement for Error and Input Reconstruction in Dynamic Systems

Dominik Kahl,¹ Philipp Wendland,¹ Matthias Neidhardt,² Andreas Weber,² and Maik Kschischo¹,*

¹Department of Mathematics and Technology, University of Applied Sciences Koblenz, Joseph-Rovan-Allee 2, 53424 Remagen, Germany
²Institute for Computer Science, University of Bonn, Endenicher Allee 19a, 53115 Bonn, Germany
(Dated: March 18, 2022)

Despite recent progress in our understanding of complex dynamic networks, it remains challenging to devise sufficiently accurate models to observe, control or predict the state of real systems in biology, economics or other fields. A largely overlooked fact is that these systems are typically open and receive unknown inputs from their environment. A further fundamental obstacle are structural model errors caused by insufficient or inaccurate knowledge about the quantitative interactions in the real system.

Here, we show that unknown inputs to open systems and model errors can be treated under the common framework of invertibility, which is a requirement for reconstructing these disturbances from output measurements. By exploiting the fact that invertibility can be decided from the influence graph of the system, we analyse the relationship between structural network properties and invertibility under different realistic scenarios. We show that sparsely connected scale free networks are the most difficult to invert. We introduce a new sensor node placement algorithm to select a minimum set of measurement positions in the network required for invertibility. This algorithm facilitates optimal experimental design for the reconstruction of inputs or model errors from output measurements. Our results have both fundamental and practical implications for nonlinear systems analysis, modelling and design.

I. INTRODUCTION

Dynamic systems in such diverse areas like physics, biology, economics or engineering are often composed of many interacting components. Developing useful and sufficiently accurate models of such complex dynamic networks having many degrees of freedom remains challenging [1–4] despite the ever increasing size of network data sets providing the wiring diagrams of diverse systems [5–8].

One important challenge for modelling real dynamic networks is that they are open systems receiving inputs from their environment, see Fig. 1. These inputs need to be either known or under experimental control to fully characterise the dynamic state of the network. For example, a biological cell is a system with a certain autonomy, but at the same time is crucially dependent on signals and nutrients received from the exterior. It is practically impossible to simultaneously detect or control all signals received by a living cell in their natural environment and to measure all compounds exchanged with the extracellular space. As another example, consider a population dynamic system in a certain geographical area. The state, i.e. the population count of the different species in this area, is not only determined by the inner dynamic interactions (e.g. pray and predator relationships) between the species, but also by migration and by environmental factors. Again, for state estimation it is typically neither feasible to directly quantify all these inputs nor is it possible to ignore them. The same applies to physical, engineering, or economic systems, which will always be subject to inputs and disturbances from the environment.

If the inputs to the system cannot directly be obtained, then we might want to infer the inputs from the measurable outputs. For the biological cell, we can try to estimate the transport fluxes across the membrane and the signals received by the cell from time series of measured protein concentrations. For a population, the number of certain species will be monitored and we will try to estimate dynamic changes of birth and migration rates for other, not directly observed species. Algorithms to estimate the inputs from the outputs of systems described by ordinary differential equations (ODEs) are an ongoing research topic, see e.g. [9–15]. However, no such algorithm can succeed, if the output doesn’t provide sufficient information about the input. Mathematically, this means that the map from input to output is not invertible and thus, systems inversion is bound to fail.

The situation is illustrated in Fig. 2(a), where a hypothetical system is represented by an influence graph. The nodes correspond to the systems states \(x = (x_1, \ldots, x_{11})\) and the black arrows indicate the endogenous interactions amongst them. The open system receives three input signals \(w(t) = (w_1(t), w_2(t), w_3(t))\) (wiggly arrows) from its exosystem, targeting the set \(S = \{x_3, x_5, x_8\}\) of three input nodes. The output signal \(y(t) = (y_1(t), y_2(t), y_3(t))\) in Fig. 2(b) is formed by measuring the time course of the sensor node set \(Z = \{x_2, x_4, x_9\}\), i.e. \(y_1(t) = x_2(t), y_2(t) = x_4(t)\) and \(y_3(t) = x_9(t)\). Nonetheless, these output observations are insufficient to uniquely reconstruct the corresponding input signals. Indeed, the two different input signals (Fig. 2(c,d)) gen-

* kschischo@rheinahrCampus.de
Figure 1. Schematic illustration of an open system and its environment. (a) The open system (blue shadowed region) interacts with its dynamic environment. The state (black nodes) of the open system is determined by both internal interactions and by the interactions with the (unknown) state of the environment (grey nodes). There are also systematic model errors, as indicated by a missing interaction between state variables (green dashed line). (b) Both the effects of the environment on the state of the open system and model errors can be regarded as unknown inputs.

erate identical outputs (solid and dashed lines Fig. 2(b)) and both reproduce the data points with good accuracy. Thus, the system is not invertible.

The lack of invertibility can be remedied by a careful experimental design. In the example of Fig. 2, the system can be made invertible by measuring the state \( y_3(t) = x_6(t) \) instead of \( x_9(t) \), compare Fig. 2(a) and (e). As explained below, the modified output from the sensor node set \( Z = \{x_2, x_4, x_6\} \) provides sufficient information to uniquely identify the input signal in Fig. 2(c) as the cause for the observed data (Fig. 2(f)). This example highlights the need for a sensor node placement algorithm for invertibility of open systems, which is one important result of this text.

Systematic model errors are another important source of potential discrepancies between measurements and model outputs. One type of model errors is caused by an incorrect influence graph, i.e. by missing or spurious interactions between the state variables (Fig. 1). Another type of model errors originates from misspecifications of the functional form or parameters of the interactions. However, both types of these endogenous model errors can effectively be treated as unknown inputs to the model system at hand, see Sec. II. As for genuine exogenous inputs, a unique reconstruction of these unknown inputs caused by endogenous model errors is again only possible if the systems model is invertible.

Controllability and observability are other important systems properties, which have attracted renewed interest from a complex systems point of view [18–22, 27–33]. Structural approaches use only the influence graph [17–22] and are therefore well suited to examine the controllability/observability properties related to topological network features. Structural controllability (observability) analysis provide binary decisions whether a system is controllable (observable) or not. Later, it was emphasised that realistic control and state estimation often require quantitative information about the network interactions and parameters of the system [24–33]. The underlying reason is that the specific functional form of the couplings might require huge energies for control or very sensitive measurements for state estimation. Continuous measures quantifying the degree of controllability or observability [24–33] were derived as alternatives to binary decisions about controllability or observability.

In contrast, the invertibility of open systems has not sufficiently been investigated in the context of large complex dynamic systems. Work in controllability and observability of complex networks builds on older results from the control engineering literature [17], providing algebraic and graphical conditions for both traits. The situation is similar for invertibility: Theorems providing algebraic conditions for the invertibility of linear systems were already published in the 1960s [35–37] and later extended to nonlinear systems [38–41]. These algebraic conditions require a full specification of the ODE system including all the parameters. Even in the rather exceptional case, where these data are available, the numerical test of these conditions is computationally very demanding for large networks. Fortunately, invertibility is a structural property [42, 43], which can for all practical purposes be inferred from the influence graph and the input and output node sets \( S \) and \( Z \) (see Fig. 2). This structural invertibility condition (see Sec. III) can efficiently be tested even for very large linear or nonlinear systems with thousands of state nodes.

In this text, we add invertibility as a systems property which is essential for understanding open and incompletely known systems. As our main new contributions we
Figure 2. Invertibility of a dynamic system. (a) The influence graph represents the states $\mathbf{x} = (x_1, \ldots, x_9)$ of a system as nodes (circles). Endogenous interactions are indicated by arrows between the states. This system receives three input signals $\mathbf{w}(t) = (w_1(t), w_2(t), w_3(t))$ (wiggly arrows) targeting the input nodes $S = \{x_3, x_5, x_8\}$. The output $\mathbf{y}(t) = (y_1(t), y_2(t), y_3(t))$ (squares) is given by measurements of the state variables in the sensor node set $Z = \{x_2, x_4, x_9\}$. (b) The measured output data (dots) of the open system can be explained by different input signals (c,d) causing identical outputs, as indicated by the dotted line for input (c) and by the solid line for input (d). Thus, the system is not invertible. (e) For $Z = \{x_2, x_4, x_6\}$, i.e. if the sensor for the output $y_3$ is moved to state $x_6$ instead of $x_9$ (compare (a)), we obtain an invertible system and the input signal in (d) is unique for the observed output in (f).
• show that unknown inputs to open systems and structural model errors can be treated under the same conceptual framework (Sec. II);
• establish invertibility as a necessary condition for unknown input observers and input reconstruction algorithms to work (Subsec. III G);
• provide a new recursive algebraic criterion for invertibility (Subsec. III C);
• discover important structural network properties influencing invertibility (Sec. IV);
• provide a simple but efficient algorithm for sensor node placement to achieve structural invertibility with a minimum number of measured outputs (Sec. V).

First we show, that structural model errors in nonlinear dynamic system can be treated as unknown inputs (Sec. II). Thus, the question of whether it is possible to reconstruct model errors and unknown inputs to open systems can be treated under the common framework of invertibility. Second, we provide a novel criterion for the invertibility of linear systems, which can be used for medium sized networks up to a few hundred state nodes due to its recursive nature (Sec. III). For large systems we exploit the structural invertibility criterion, which uses only the graph structure encoded in the interactions of the system. We will also briefly touch upon the topic of practical systems inversion, which requires careful regularisation schemes even for invertible systems. Throughout this text, we will consider three different scenarios (SC I-III). Scenario SC I refers to the case, that the positions of the inputs and outputs of the system are given and that we have no opportunity to deliberately choose these positions (Sec. IV). For SC I we show that invertibility depends largely on the degree distribution of the influence graph and that many sparse and scale free networks tend to be non-invertible. Since many real networks show this characteristics, we assume in scenario SC II, that the inputs are given, but the output positions can be chosen. As an important result we present a sensor node placement algorithm in Sec. V, which provides a minimum set of outputs required to uniquely reconstruct the inputs. This algorithm is also useful in scenario SC III, where we assume that the positions of both inputs and outputs can be manipulated. We show that placing the inputs at hubs with a high out-degree can drastically increase the probability that a certain dynamic network can be made invertible, if in addition the outputs are suitably chosen, e.g. by our sensor node placement algorithm. In Sec. VI we discuss the far reaching implications of invertibility for nonlinear systems analysis and some open questions for further research.

II. OPEN SYSTEMS, UNKNOWN INPUTS AND MODEL ERRORS

There are two basic reasons for discrepancies between observed time series data from a real world open system and the output of a mathematical model for the system: First, the system might receive unknown inputs [12–14, 34, 44, 45] from the environment, which are not covered by the model, but nevertheless influence the state and the resulting measured output. Second, there might be model errors, i.e. misspecified functional descriptions or missing interactions between internal model states or inaccurate parameter values. In this section, we define open systems with unknown inputs and then show that for ODE models, both types of error can be treated as additive unknown inputs to the model.

A. Open systems with unknown inputs

Consider a dynamic system $\mathcal{S}_0$ with time dependent state vector $x(t) = (x_1(t), \ldots, x_N(t))^T \in \mathcal{X}$. The state space $\mathcal{X}$ is either $\mathbb{R}^N$, a subset of it, or an $N$-manifold. A dynamic model of the open system can be formulated as a system of ordinary differential equations (ODEs)

$$\dot{x}(t) = f(x(t)) + Dw(t)$$  \hspace{1cm} (1a)
$$x(0) = x_0$$ \hspace{1cm} (1b)
$$y(t) = c(x(t))$$ \hspace{1cm} (1c)

The vector field $f(x) = (f_1(x), \ldots, f_N(x))^T$ represents the internal dynamic interactions between the state variables. In addition, the system receives $M$ unknown inputs, which are collected in the vector function

$$w(t) = (w_1(t), \ldots, w_M(t))^T.$$ \hspace{1cm} (2)

The $N \times M$ matrix $D$ describes, how the unknown input signals are distributed over the $N$ states $x$. The rows of $D$ provide information about the inputs acting on the respective state. Zero rows correspond to states which are not directly affected by model errors. As discussed below, the unknown input $u$ incorporates genuine inputs from the exterior as well as all possible types of model errors, including incorrect interactions between internal states $x$ and incorrect parameter values. The unknown input aka model error $w$ needs to be estimated from data. This would not be a problem, if the internal system state $x(t)$ could directly be measured. However, typically only a smaller set of $P$ scalar output signals

$$y(t) = (y_1(t), \ldots, y_P(t))^T = c(x(t))$$ \hspace{1cm} (3)

is directly accessible to measurements. The map from the state to the output is here assumed to be given by the function $c(x)$. We assume that both $f$ and $c$ satisfy a Lipschitz condition. In addition, the initial state $x_0$ needs to be specified.
It is often useful to represent systems of the form in Eq. (1) as an influence graph, see Fig. 2(a) for an example. The nodes of this directed graph correspond to the states \( x_i, i = 1, \ldots, N \), and the edges represent the interactions encoded by \( f \). More precisely, a directed edge \( x_i \rightarrow x_j \) indicates that \( \frac{\partial f_j}{\partial x_i} \neq 0 \) for some \( x \) in the state space \( X \). The state nodes targeted by unknown inputs are determined by the nonzero rows of the matrix \( D \). Please note, that these input nodes are sometimes also called driver nodes, in particular in the context of control. In the special case were each unknown input component \( w_k, k \in \{1, 2, \ldots, M\} \) affects only one state node we can choose each element \( D_{ik} \) to be either zero or one and we have

\[
D_{ik} \in \{0, 1\} \tag{10a}
\]

and

\[
\sum_{k=1}^{M} D_{ik} = \begin{cases} \ 1 & \text{if input } w_k \text{ affects state } x_i \\ \ 0 & \text{otherwise} \end{cases}, \tag{10b}
\]

as indicated in Fig. 2(a) by red arrows. Similarly, if the output function is given by the linear relationship \( e(x) = Cx \) with the \( P \times N \)-matrix \( C \) and if, in addition, we have \( C_{ij} \in \{0, 1\} \) and \( \sum_j C_{ij} \in \{0, 1\} \), we can indicate the subset of directly measured states by blue arrows. These correspond to the nonzero columns of \( C \).

### B. Model errors and unknown inputs

We now show that the unknown input function \( w(t) \) in Eq. (1) can represent both the effect of an exterior dynamic system to our open system \( S \), and the effect of structural model errors in \( f \), i.e. erroneous descriptions of the interactions between the internal state variables.

Let us start with a closed system \( S \) modelled by

\[
\dot{x}(t) = f(x(t)), \tag{11}
\]

where only the internal dynamics of the system is described, but unknown inputs are not taken into account. In reality, however, the system \( S \) might be embedded into a larger system (see Fig. 1) and interact also with external state variables \( \xi(t) = (\xi_1(t), \ldots, \xi_N(t))^T \), which are not included in the model in Eq. (11). Instead, the joint dynamics of the system \( S \) and its exterior should rather be described by

\[
\dot{x}(t) = \phi(x(t), \xi(t)) \tag{12a}
\]

\[
\dot{\xi}(t) = \psi(x(t), \xi(t)) \tag{12b}.
\]

The function \( \phi \) combines the effect of interactions between internal states \( x \) and the effect of the external states \( \xi \) on the dynamics of the internal states. The dynamics of the external system is determined by \( \psi \), which is usually not known and thus difficult to include in the model. This unknown dynamics leads to the structural model error \( \eta \), which can formally be defined as the discrepancy between the model in Eq. (11) and the system Eq. (12a)

\[
\eta(x(t), \xi(t)) := \phi(x(t), \xi(t)) - f(x(t)) \tag{13}\]

Thus, instead of explicitly extending the simpler model Eq. (11) to the potentially complicated joint system Eq. (12), we could correct the former by adding an unknown input

\[
\eta(t) := \eta(x(t), \xi(t)) \tag{14}
\]

to obtain

\[
\dot{x}(t) = f(x(t)) + \eta(t). \tag{15}\]

Replacing \( \eta(t) = Dw(t) \) in Eq. (15) leads to Eq. (1).

Let us illustrate the relationship between model errors and unknown inputs by a simple concrete example. Consider the following model

\[
\begin{align}
\dot{x}_1(t) &= -0.2 x_1(t) \tag{16a} \\
\dot{x}_2(t) &= \frac{x_1(t)}{1 + x_4(t)} - x_2(t) \tag{16b} \\
\dot{x}_3(t) &= x_2(t) - x_3(t) \tag{16c} \\
\dot{x}_4(t) &= x_3(t) - x_4(t) \tag{16d}
\end{align}
\]

for a protein cascade [46] and assume that Eq. 16c in the model is misspecified. Instead, assume that the correct description is

\[
\dot{x}_3(t) = x_2(t) - \frac{x_3(t)}{a(t) + x_3(t)} \tag{17}
\]

with a time dependent function \( a(t) \). Thus, the degradation of \( x_3 \) is an enzyme catalysed reaction with a time dependent affinity \( a(t) \), which can not be described by the mass action term in Eq. 16c. In addition, the affinity \( a(t) \) is controlled by an exogenous regulatory process, which is not covered by the model in Eq. (16). The model error (compare Eq. (13)) is given by

\[
\eta(t) = \begin{pmatrix} 0, 0, x_3(t) - \frac{x_3(t)}{a(t) + x_3(t)} \end{pmatrix}^T \tag{19}
\]

\[
= (Dw_1(t))^T
\]

with \( w_1(t) = x_3(t) - \frac{x_3(t)}{a(t) + x_3(t)} \) and \( D = (0, 0, 1)^T \). If the correct relationship in Eq. (17) is unknown, we can replace Eq. (16c) by

\[
\dot{x}_3(t) = x_2(t) - x_3(t) + w_1(t) \tag{18}
\]

and treat \( w_1(t) \) as an external input, which needs to be estimated from measurement data. An example for such an unknown input estimate is provided in Subsec. III G.
III. CRITERIA FOR INVERTIBILITY

If the unknown input function \( w(t) \) can uniquely be reconstructed from the output signal \( y(t) \), we call the system invertible. An algebraic criterion to check for invertibility of a system was first derived for linear systems [37], see Subsec. III D for details. An algorithm to invert the system, which terminates in the case of a non-invertibility, was also first devised for the linear case [36], but later extended [38] to nonlinear affine models of the form given in Eq. (1). Another type of results is based on differential geometric or differential algebraic criteria for the invariant control distributions [39–41]. All these criteria involve algebraic manipulations of the systems equations, which makes them useful for smaller models, but limits their utility for large networks. In addition, the invertibility tests require a full specification of the system (Eq. (1)), including the complete functional form and the parameters of the interaction terms encoded by \( f \).

Here, we state the exact mathematical definition for invertibility of dynamic systems [38]. Then, we provide a new recursive algorithm to check invertibility for linear systems, which might be easier to apply to systems of moderate size, in contrast to the mathematically equivalent algebraic rank condition [37]. However, for large networks, the structural invertibility algorithm has the huge advantages of scaling to large systems and of only requiring the topology of the influence graph. Invertibility is only a necessary condition and the robustness of unknown input reconstruction to measurement noise might be influenced by other factors than just the network structure. We briefly touch upon this important problem, discuss the role of regularisation and provide a simple, illustrative example for an unknown input estimate.

A. Invertibility

Mathematically, invertibility means that the map from the unknown input signal \( w \) to the output \( y \) is injective, which can be expressed as [38]:

Definition 1. The system (1) is invertible at the initial state \( x_0 \), if two distinct input signals \( w(t) \) and \( \bar{w}(t) \) always induce two distinct outputs \( y(t) \neq \bar{y}(t) \). If the system is invertible in an open neighbourhood of \( x_0 \), it is called strongly invertible at \( x_0 \). The system is strongly invertible, if there exists a dense submanifold \( M \) of \( X \), such that the system is strongly invertible for any \( x_0 \in M \).

For linear systems

\[
\begin{align*}
x(t) &= Ax(t) + Dw(t) \\
x(0) &= x_0 \\
y(t) &= Cx(t)
\end{align*}
\]

(19a)

(19b)

(19c)

with a real \( N \times N \) matrix \( A \) and an \( P \times N \) output matrix \( C \), all the three definitions are equivalent [38], since invertibility at some \( x_0 \in \mathbb{R}^N \) implies invertibility at all points in their state space \( X = \mathbb{R}^N \). Such linear systems are typically obtained as local approximations of the nonlinear model in Eq. (1), where \( A \) and \( C \) are given by the Jacobi matrices of \( f \) and \( c \), respectively, taken at a certain reference point.

For linear systems (Eq. (19)), invertibility is a global property and thus it is sufficient to consider the initial condition \( x_0 = 0 \). Then, the input-output map \( \Phi \) is given by the linear operator

\[
y(t) = (\Phi w)(t) = \int_0^t C \exp[A(t-s)]Dw(s) \, ds. \quad (20)
\]

The linear system is invertible, if this operator is one-to-one. Below, we state two different versions of an algebraic criterion to decide invertibility for linear systems. More details on the three criteria are given in the Appendix. Here, we only motivate the structure of the algebraic criteria below: Taking successive time derivatives \( y^{(l)}(t) \) of the output (Eq. (20)), for \( l \in \{1, 2, \ldots \} \)

\[
y^{(l)}(t) = CA^l x(t) + \sum_{k=0}^{l-1} CA^{l-k-1} Dw^{(k)}(t), \quad (21)
\]

and evaluating at \( t = 0 \), we obtain the sequence of linear equations

\[
\begin{align*}
\dot{y}(0) &= CDw(0) \\
\ddot{y}(0) &= CAdw(0) + CD\dot{w}(0) \\
\vdots \\
y^{(l)}(0) &= CA^{l-1} Dw(0) + CA^{l-2} D\dot{w}(0) + \cdots + CDw^{(l-1)}(0).
\end{align*}
\]

Invertibility implies, that we can solve these linear equations uniquely for \( w^{(l)}(0) \) for given \( y^{(l)}(0) \), or, equivalently, that \( y^{(l)}(0) = 0 \) implies \( w^{(l)}(0) = 0 \). Basically, the two equivalent algebraic criteria in Subsecs. III D and III C provide conditions for unique solutions of this system.

B. Invertibility versus Unknown Input Observability

Let us clarify the relationship between unknown input observability and invertibility. The notion of unknown input observability is not uniquely defined in the literature. Some authors call a system unknown input observable, if the state is completely or partially observable even in the presence of unknown inputs (see e.g. [47]). This weaker definition does not necessarily imply that the unknown input itself can be reconstructed. Other authors define unknown input observability in the stricter sense that both the systems state as well as the unknown input can be inferred from the outputs [44].

Invertibility or structural invertibility ensures, that the unknown input can uniquely be reconstructed from the
output. It does not necessarily imply that the system state can simultaneously be observed [44]. Only if the initial state \( x_0 \) is known, then invertibility is not only necessary but also sufficient for simultaneous state and input observability.

It is only recently, that a general algorithm for testing the weaker form of unknown input observability of nonlinear systems exists [47]. This algorithm is based on symbolic computation and thus restricted to very small systems with only a few state variables.

C. Recursive Algebraic Criterion for Invertibility of Linear Systems

For the first version of the algebraic criterion we define a sequence of block matrices

\[
R_l := \left[ \begin{array}{c|c|c} \cdots & CD & \cdots \\ \hline 0 & CD & \cdots \\ \hline \vdots & \ddots & \vdots \\ \hline 0 & \cdots & CD \\
\end{array} \right], \quad l \in \mathbb{N}_0 ,
\]

which appear in the derivatives of the output (Eq. (21)). Each matrix \( R_l \) in the sequence has \( P \) rows and \((l+1)M\) columns. Recall, that \( P \) was the number of measurement signals and \( M \) the number of unknown inputs. For each matrix \( R_l \) in the sequence, the null space is defined as

\[ \ker R_l = \left\{ v \in \mathbb{R}^{(l+1)M} \mid R_l v = 0 \right\}, \quad l \in \mathbb{N}_0. \]

In addition, we recursively define the following sets

\[
\begin{align*}
K_0 & := \ker R_0 \backslash \{0\} \\
K_l & := \ker R_l \cap (\mathbb{R}^M \times K_{l-1}) .
\end{align*}
\]

Here, “\( \times \)” indicates the cartesian product of two sets. Now we can state the invertibility criterion: The linear system in Eq. (19) is invertible, if and only if \( K_l = \emptyset \) for some \( l \leq N - 1 \).

To apply this criterion we have to calculate the null spaces of all \( R_l \) and then iteratively to evaluate the sequence of sets \( K_l \), starting from \( l = 0 \). The iteration terminates if \( K_l = \emptyset \) for some \( l \leq N - 1 \), indicating invertibility. If no such \( l \) can be found, the system is not invertible.

D. Rank Condition for Invertibility of Linear Systems

The iterative criterion above is equivalent to the following algebraic rank condition proofed by Sain and Massey [37]: Consider the sequence of matrices

\[
Q_l := \left[ \begin{array}{c|c|c} \cdots & CD & \cdots \\ \hline 0 & CD & \cdots \\ \hline \vdots & \ddots & \vdots \\ \hline 0 & \cdots & CD \\
\end{array} \right].
\]

(24)

for \( l \in \mathbb{N}_0 \). The linear system in Eq. (19) is invertible, if and only if

\[ \text{rank } Q_{N-1} - \text{rank } Q_{N-2} = M . \]

As before, \( N \) is the number of states in the system and \( M \) the number of inputs. Thus, the criterion requires to compute the rank of two matrices. The size of these matrices increases quadratically with the number of states \( N \) in the system. Such rank computations can be very memory intensive for large networks with many nodes.

It is worthwhile remarking on an interesting property of invertibility. For an invertible system, the null space of the operator \( \Phi \) is zero dimensional, containing as its single element the zero input \( \mathbf{w}(t) \equiv \mathbf{0} \). For a non-invertible system, the null space of \( \Phi \) is always infinite dimensional (see Lemma 1 in the Appendix). This means that for non-invertible systems there are infinitely many independent inputs which cannot be distinguished from each other. This shows, that there is no such thing like “nearly invertible”. Thus, any algorithm attempting to infer the inputs for the outputs is bound to fail without further assumptions about the inputs. Assumptions like smoothness and sparsity of the input signals can be encoded into these inversion algorithms by using suitable regularisation schemes [12] or Bayesian priors [13]. However, even these additional smoothness and sparsity assumptions restricting the domain of the input-output map are not always sufficient for invertibility [12].

E. Graphical Criterion for Linear and Nonlinear Systems

The quite intricate algebraic conditions can be replaced by a simple graphical criterion [43], see Fig. 3. Recall, that in the influence graph \( g \) each state variable \( x_i \) is represented as a node and the edges are determined by the adjacency matrix \( A \): For each \( A_{ij} \neq 0 \) draw an directed edge \( i \to j \). Now assume, that the columns of the input matrix consist only of a subset of \( M \) canonical basis vectors \( \mathbf{e}_k \in \mathbb{R}^N \). Thus \( D_{ik} \in \{0,1\} \) with \( \sum_k D_{ik} \in \{0,1\} \). Then, the nonzero rows of \( D \) indicate the states receiving an input signal. Denote these \( M \) input nodes as

\[ S = \{s_1, \ldots, s_M\} \subseteq \{x_1, \ldots, x_N\} . \]

(26)

Similarly, we assume that the output matrix \( C \) has only \( P \) rows which are a subset of the canonical basis of \( \mathbb{R}^P \) and thus \( C_{ji} \in \{0,1\} \) with \( \sum_j C_{ji} \in \{0,1\} \). Then, the nonzero columns of \( C \) indicate the output or sensor nodes

\[ Z = \{z_1, \ldots, z_P\} \subseteq \{x_1, \ldots, x_N\} , \]

(27)

i.e. the state nodes for which direct measurements are available.

Now, the graphical criterion can be stated as follows: The linear system in Eq. (19) is structurally invertible, if and only if there is a family \( \Pi \) of \( M \) directed paths in the influence graph \( g \) fulfilling the following conditions

(i) Each path in \( \Pi \) starts in \( S \) and terminates in \( Z \).

(ii) All paths of \( \Pi \) are pairwise node-disjoint.
In the following, we will call the triplet \((S, g, Z)\) consisting of the influence graph \(g\), the input node set \(S\) and the output node set \(Z\) invertible, if the structural invertibility criterion is fulfilled.

To put it differently: There must be a family \(\Pi\) of directed paths in \(g\) connecting each input node in \(S\) with an output node in \(Z\) and no two paths in the family intersect at any node of the influence graph. If no such family of node-disjoint paths exists, the system is structurally non-invertible. For the example graphs in Fig. 2(a,b), this path condition is illustrated in Fig. 3(a,b) for the non-invertible (a) and invertible case (b), respectively. Obviously, a system with fewer output nodes than input nodes is never invertible.

This intuitive graphical condition implies, that only the structure of the influence graph and the position of the inputs and outputs, i.e. only the patterns of nonzero entries in the systems matrices \(A\), \(D\), and \(C\), are relevant for invertibility [43]. Indeed, structural invertibility and algebraic invertibility coincide up to pathological cases, where the graphical condition could indicate structural invertibility, whereas none of the algebraic conditions would be fulfilled due to an exact cancellation of numerical terms. These pathological conditions are irrelevant in practice, since any arbitrarily small numerical perturbation of one of the nonzero terms in the systems matrices would repair invertibility. Or, in mathematical language: The set of systems matrices \(A\), \(D\), and \(C\), for which the graphical and the algebraic conditions give contradictory results is a set of measure zero. The situation is completely analogous to the structural and algebraic controllability or observability conditions [17–19, 43].

The structural invertibility condition was extended to nonlinear systems of the form in Eq. (1), see [42] for details. This means, that we can replace the matrix \(A\) by the Jacobi matrix of \(f\) and the output matrix \(C\) by the
Jacobi matrix of \( c \) at some point of the state space \( X \) to obtain the systems graph and the output node set \( Z \). Thus, the structural properties of the linearisation of Eq. (1) are also sufficient to detect the invertibility of a nonlinear system. There is one subtlety for nonlinear systems: The structural invertibility condition does not imply regularity of the system, which is relevant for feedback systems, see [42] for further details.

F. Structural Invertibility Algorithm

The graphical criterion for (structural) invertibility requires to count the number of node-disjoint paths connecting the input nodes \( S \) with the output nodes \( Z \). Counting all these paths in a combinatorial manner is not feasible for larger systems. Thanks to the Max-Flow-Min-Cut-Theorem [43, 48, 49], the graphical node disjoint path counting problem can be reformulated as flow problem, which can efficiently be solved.

As an initial step of the algorithm, the influence graph \( g \) is transformed to a corresponding flow graph \( G \) by copying each node \( i \) to separate the ingoing and outgoing edges (see Fig. 4). Now, a family \( \Pi \) of node-disjoint paths in the original graph \( g \) corresponds to a family of edge-disjoint paths in \( G \). In a second step, an additional source node \( \sigma \) is connected to each of the input nodes and an additional sink node \( \zeta \) is connected to each of the output nodes. If each edge in the resulting graph \( G \) is assigned a weight of 1, the maximum flow from source \( \sigma \) to sink \( \zeta \) in \( G \) equals the number of edge-disjoint paths from source to sink, and thus the number of node-disjoint paths from \( S \) to \( Z \) in the original graph \( g \). In our implementation we use the Goldberg-Tarjan algorithm [50] to efficiently compute the maximum flow. Several alternative algorithms exist in the combinatorial optimisation literature, see e.g. [49].

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 \( g \). As before, we denote the number of input nodes by \( M \) and the number of sensor nodes by \( P \). 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 \( G \) we use the Goldberg-Tarjan algorithm with running time scaling like \( O(n^2 \sqrt{e}) \) to compute the maximum flow. All together we find that the structurally invertibility algorithm has a running time of \( O(N^3) \). Please note that there are even more efficient optimised versions of the Goldberg-Tarjan algorithm with better running time [49], but for all our purposes the standard version was sufficient.

Our implementation is based on the python networkx package. On a single node of an Intel Xeon Processor E5-2690 v2 we need an average time of 0.15 \( \pm \) 0.03 seconds for a network with \( N = 10^3 \) nodes and 2.4 \( \pm \) 0.4 seconds for \( N = 10^4 \) nodes to decide structural invertibility.

G. Practical invertibility and robustness

Invertibility (or structural invertibility) is a necessary condition for the unique reconstruction of unknown inputs \( \mathbf{u}(t) \) from outputs \( \mathbf{y}(t) \). If the input-output map \( \Phi \) defined by the general system in Eq. (1) is invertible, the operator equation

\[
y = \Phi \mathbf{w}
\]

for given \( y = y(t) \) has a unique solution \( \mathbf{w}(t) \) and the inverse operator \( \Phi^{-1} \) exists.

In reality, we have to reconstruct the unknown input \( \mathbf{w} \) from measured output data \( y^\text{data} \), which will always be subject to measurement errors and noise. Therefore, the data based estimate \( \hat{\mathbf{w}} = \Phi^{-1} y^\text{data} \) will differ from the true input \( \mathbf{w} = \Phi^{-1} y \). For a discontinuous inverse operator \( \Phi^{-1} \), the difference between \( \mathbf{w} \) and \( \hat{\mathbf{w}} \) can be drastic. Unfortunately, the inverse operator of the general nonlinear system Eq. (1) is not continuous. Thus, estimating the unknown input from real data remains an ill posed problem [51, 52], even if the system is invertible and \( \Phi^{-1} \) exists.

The underlying reason for the discontinuity of the inverse input-output operator \( \Phi^{-1} \) of the linear system (Eq. (19)) is a well known theorem from the theory of inverse problems (see e.g. [51, 52]): Linear compact operators with an infinite dimensional range cannot have a continuous inverse. The linear input-output operator Eq. (20) corresponding to the linear system in Eq. (19) is an integral operator and thus it is compact and has infinite range, as shown in the Appendix (Theorem 1). For nonlinear operators, a similar theorem states that completely-continuous operators with infinite range cannot have a continuous inverse. This indicates, that the inverse of the nonlinear system Eq. (1) cannot be continuous.

The degree of discontinuity of the inverse to the linear compact operator in Eq. (20) can be quantified by means of the singular value decomposition (SVD). Since \( \Phi \) has an infinite dimensional range, its SVD is an infinite series. The infinite series \( (\sigma_k) \), \( k = 1, 2, \ldots \) of singular values is usually ordered by decreasing magnitude \( (\sigma_k \geq \sigma_{k+1}) \). The smaller singular values determine the response of \( \Phi^{-1} \) to high frequency components of \( y^\text{data} \) and are thus responsible for the discontinuity of the inverse. Thus, it is in principle possible to quantify the degree of discontinuity by the speed of decay of the sequence \( (\sigma_k) \). The problem in Eq. (28) is considered to be mildly ill-posed, if the sequence of singular value \( (\sigma_k) \) decays at most with polynomial speed, whereas it is called severely ill-posed, if \( (\sigma_k) \) decays faster than any polynomial [51]. However, this approach is not straightforward to implement, because it requires computing the spectrum of the gramian operator \( \Phi^* \Phi \), where \( \Phi^* \) is the adjoint of Eq. (20). In addition, it is not yet clear
whether the SVD can also be useful for the inverse nonlinear input-output operator corresponding to nonlinear system Eq. (1), possibly after a suitable linearisation. This is certainly beyond the scope of this text and we leave this as an interesting direction for further research.

Please note, that the situation for invertibility is different from that of controllability or observability [24, 27–33], where the corresponding gramian matrices correspond to operators with finite dimensional range. Consequently, there is a smallest singular value which can be used as the condition number characterising the degree of controllability or observability, respectively.

Regularisation of Unknown Inputs

There are several algorithms for estimating the unknown input \( \mathbf{w} \) from measurement data, ranging from feedback controllers via modifications of the nonlinear Kalman filter to moving horizon estimation [9–15]. We can not discuss all these approaches here, but it is instructive to briefly discuss a simple version of the optimisation based approach, where an error functional \( J[\mathbf{w}] \) is minimised with respect to \( \mathbf{w}(t) \). This leads to the following optimal control problem

\[
\begin{align*}
\text{minimize } J[\mathbf{w}] &= d[\mathbf{y}^{\text{data}}, \mathbf{y}] + R[\mathbf{w}] \\
\text{subject to } & \quad \dot{x}(t) = f(x(t)) + \mathbf{w}(t) \\
& \quad \mathbf{y}(t) = c(x(t)).
\end{align*}
\]

(29)

Here, \( d \) quantifies the fit of the corrected model output \( \mathbf{y} \) to the data \( \mathbf{y}^{\text{data}} \). A typical choice is the squared error

\[
d[\mathbf{y}^{\text{data}}, \mathbf{y}] = \frac{1}{T} \sum_{i=1}^{P} \int_{0}^{T} (y_i^{\text{data}}(t) - y_i(t))^2 dt.
\]

(30)

which penalises the squared 2-norm of the unknown input \( \mathbf{w} \) or

\[
R[\mathbf{w}] = \frac{1}{T} \sum_{i=1}^{P} \int_{0}^{T} \mathbf{w}_i^2(t) dt,
\]

(31)

Even for invertible systems the regularisation is necessary, because the unregularised least squares fit (\( \alpha = 0 \)) will be very sensitive to measurement errors, again a symptom of the discontinuity of the inverse input-output operator \( \Phi^{-1} \) corresponding to Eq. (1). Typical choices for the regularisation function are

\[
R[\mathbf{w}] = \frac{1}{T} \sum_{i=1}^{P} \int_{0}^{T} \mathbf{w}_i^2(t) dt,
\]

(32)

which penalises the norm of the first derivative \( \dot{\mathbf{w}} \). These two examples are known as Tikhonov regularisation in function spaces [52]. For linear operators like the Eq. (20), the effect of the regularisation is to suppress the effect of small singular values. There are many more possible choices for regularisation terms, e.g. sparsity promoting regularisation [12]. Often, regularisation terms are also chosen to render the optimisation problem convex, which avoids problems with local minima [53].

Example for an unknown input estimate

In our computational experiments we found the regularisation of the derivative as in Eq. (30) to yield good results for the case of additive measurement noise, where \( \mathbf{y}^{\text{data}} - \mathbf{y} \) is a zero mean stochastic disturbance of the output measurement. This is illustrated in Fig. 5 for the model of a protein cascade [46]

\[
\begin{align*}
\dot{x}_1(t) &= -0.2x_1(t) + w_1(t) \\
\dot{x}_2(t) &= \frac{x_1(t)}{1 + x_2(t)} - x_2(t) \\
\dot{x}_3(t) &= x_2(t) - x_3(t) + w_2(t) \\
\dot{x}_4(t) &= x_3(t) - x_4(t),
\end{align*}
\]

(33)

with known initial value

\[
x_0 = (1, 0.5, 0.5, 0.1)^T.
\]

(34)

We generated synthetic data by solving the system of ODEs for a given input function \( \mathbf{w}(t) \) acting on the input nodes \( S = \{x_1, x_3\} \) and adding gaussian noise to the output, see Fig. 5(a,b). We then tried to recover the input from the noisy output data by solving the regularised optimal control problem in Eqs. (29) and (30). If the measurement nodes are \( Z = \{x_3, x_4\} \), the measured output data can accurately be fitted (Fig. 5(b)), but the recovered inputs \( \delta(c) \) do not correspond to the true inputs. The reason is that the system with the output nodes \( Z = \{x_3, x_4\} \) is not structurally invertible. For the sensor node set \( Z = \{x_2, x_4\} \), the system is structurally invertible and the solution of the optimal control
problem (Eqs. (29) and (30)) provides a reconstruction of the inputs from the noisy measurements, see Fig. 5(d-f). In particular, the magnitude of the estimate for $w_1$ is very small and differs from the true unknown input $w_1 = 0$ only due to transient effects of the numerics and measurement noise.

This simple example illustrates again the importance of structural invertibility as a prerequisite for systems inversion. As discussed above, the accuracy of the estimates can vary with the degree of continuity of the inverse input-output operators, which itself depends on the functional form and on the specific parameters of the ODE. However, structural invertibility is a necessary requirement to estimate unknown inputs or model errors. Therefore, we will focus on structural aspects of invertibility in the remainder of this text.

IV. STRUCTURAL INVERTIBILITY OF COMPLEX NETWORKS

It is natural to ask whether certain network properties affect structural invertibility. It has been shown previously, that important systems properties including controllability [18], observability [19] or target controllability [20] are related to network structure, see [21, 22] for reviews.

In this section we will explore the invertibility of large simulated and real networks using the very efficient structural invertibility algorithm from Subsec. III F. To mimic scenario SC I, where we have no influence on the choice of the input and output nodes, we will first use a uniform random sampling scheme for the selection of both node sets. Later, we will investigate the effect of hubs on invertibility and study the preferential selection of hubs as input or output nodes.

A. Invertibility of random and scale free networks under uniform sampling (scenario SC I)

Intuitively, a densely connected network allows to find many node-disjoint paths connecting the input node set $S$ to the output node set $Z$. Thus, for a set of randomly selected input nodes $S$ and a disjoint set of randomly selected output nodes $Z$, the chance for invertibility should increase with the average degree $\bar{d}$ of the network $g$.

To test this hypothesis, we simulated graphs with $N = 10^3$ nodes using either Erdős-Rényi random graphs [54] or scale-free networks [55, 56] with varying average degree $\bar{d}$. Throughout these simulations, we used $M = P$, i.e. the same number of input and output nodes. For a given graph $g$, we first sampled a set $S$ of $M$ input nodes uniformly at random and then randomly sampled the set $Z$ of $P$ output nodes from the remaining nodes, such that the input and output node sets are disjoint: $S \cap Z = \emptyset$. We will refer to this sampling scheme for inputs and outputs as uniform random sampling, which simulates scenario SC I. Then, we used the structural invertibility algorithm to check whether the resulting network represents the influence graph of an invertible or a non-invertible system (Eq. (20)). To estimate the probability $\rho = \rho(\bar{d}, N, M)$, that a graph with $N$ nodes, $M = P$ inputs and outputs and average degree $\bar{d}$ is invertible under this random scheme, we sampled 100 triples of $(S, g, Z)$ of different graphs and input/output node sets and counted the relative frequency of structurally invertible systems represented by these graphs.

As can be seen in Fig. 6(a), the probability of invertibility for Erdős-Rényi random graphs increases indeed monotonously with the average degree $\bar{d}$. For small $\bar{d}$, almost no graph is invertible, whereas for large $\bar{d}$ almost all graphs are invertible. These two regimes are separated by a transition zone, where some networks are invertible and others are not. In this transition zone, invertibility depends on the specific characteristics of the random graph and the average degree is not sufficient to decide about invertibility. For more inputs and outputs (increasing $M = P$), the transition zone moves towards higher $\bar{d}$. This is plausible, because a family of $M = P$ node disjoint paths $\Pi$ connecting the input and the output sets $S$ and $Z$ cannot be found in sparse networks with a small overall number of paths. We found empirically, that for $M = P$ the function $\rho = \rho(\bar{d}, N, M)$ attains an asymptotic limit $\rho(\bar{d}, M)$ for large networks with a given average degree ($N \to \infty$ and $\bar{d}$ fixed).

Scale-free networks offer another network topology induced by a power law degree distribution $P(k) \propto k^{-\gamma}$ that has been observed to be the underlying structure of many real networks [55, 56]. Scale-free networks have a tendency for a few highly connected hubs and many weakly connected satellites. The effect of this heterogeneity is not immediately obvious: On the one hand, the hubs act as bottlenecks, that shrink the chance of finding node-disjoint paths. On the other hand, the diameter of scale-free networks is much smaller than the diameter of Erdős-Rényi random graphs [57]; hence paths are shorter and might possibly find their way to the output set $Z$, before they intersect.

The python networkx package was used to do the simulations. For the Erdős-Rényi graphs $g = g(N, p)$ we used fast_gnp_random_graph. We implemented the static model from [56] to generate scale-free graphs $g = g(\bar{d}, N, \gamma)$. Here, $N$ is the node number, $p$ the probability, that an edge in the Erdős-Rényi graph exists, $\bar{d}$ is the average degree and $\gamma$ the exponent in the power law degree distribution. As before we drew 100 graphs, distributed $M = P$ input and output nodes uniformly over each graph, and took $\rho$ as the fraction of structural invertible graphs. For a given number of inputs and outputs, the transition zone for scale free networks (see Fig. 6(b)) is broader in comparison to Erdős-Rényi systems. In scale free networks, increasing the number of inputs and outputs has a more drastic effect on diminishing the chance for invertibility, as can be seen from the larger gaps between the different curves in Fig. 6(b) com-
Figure 5. Estimation of unknown inputs to a simple protein cascade. (a) The influence graph of the system in Eq. (31) with input nodes $S = \{x_1, x_3\}$ and output nodes $y_1 = x_3$ and $y_2 = x_4$. (b) The model output for the system without unknown inputs (blue) can not reproduce the observed data (dots with error bars for measurement noise). Solving the regularised optimisation problem (see Eqs. (29) and (30)) provides accurate fits to the data (red line). (c) However, the estimates of the unknown inputs (red lines) are incorrect (dashed lines indicate the true inputs), because the system is not structurally invertible. (d) The system is structurally invertible, if $y_1 = x_2$ and $y_2 = x_4$ are measured. (e,f) The fit to the measurements in (e) is now sufficient to estimate the unknown inputs in (f). Please note also the different scale of the plots for in (c) and (f). All values are understood in arbitrary units.

B. The role of the degree distribution

To explore the effect of the degree distribution on invertibility, we compared the scale free E.coli metabolic network [58] to an ensemble of simulated scale free networks. The E.coli metabolic network has an estimated power law exponent of $\gamma = 2.61$ and an average degree of $\bar{d} = 11.17$. We used the static model and the same parameters to simulate the ensemble of 100 scale free
The structural invertibility of (a) Erdős-Rényi random graphs and (b) scale free networks (power law exponent $\gamma = 2.4$) depends on the average node degree $\bar{d}$. For each data point, an ensemble of 100 networks with $N = 10^3$ nodes was simulated and disjoint sets of $M = P$ input and output nodes were chosen by uniform random sampling. Each network was tested for structural invertibility and the fraction $\rho$ of invertible networks in the ensemble was recorded. For large networks with many nodes $N \to \infty$ and fixed average degree $\bar{d}$, the function $\rho(\bar{d}, N, M)$ asymptotically approaches $\rho(\bar{d}, M)$. We found empirically that $N = 10^3$ is a good approximation for this asymptotic regime.

We selected 100 input and output node sets for the E.coli network using uniform random sampling (scenario SC I) and computed the fraction $\rho$ of invertible systems as a function of the number $M = P$ of inputs and outputs. The uniform random sampling scheme was also applied to each of the 100 simulated scale free networks. Intriguingly, we found that the probability for invertibility $\rho$ is higher in the simulated networks than in the E.coli metabolic network, see Fig. 7(a). In addition, we performed a degree-preserving randomization (rand-Degree) [59] to all networks (E.coli and simulated) and found that this doesn’t change $\rho$, up to small sampling deviations (see next Subsec. IV C). In this degree-preserving randomization, the in-degree $d_{in}$ (number of incoming edges) and the out-degree $d_{out}$ (number of outgoing edges) of each node is preserved, but the nodes which link to each other are randomly selected.

In Fig. 7(b,c) we have plotted $d_{out}$ versus $d_{in}$ for the E.coli metabolic network (b) and a typical simulated scale free network (c). It can be seen, that the E.coli network has many more high degree nodes with a large difference $d_{out} - d_{in}$ between out- and in-degree. This asymmetry is by construction much smaller in the simulated networks. These results indicate, that the joint distribution of in- and out-degrees $P(d_{in},d_{out})$ largely determines the probability of finding an invertible system.

![Figure 6](image-url)  
Figure 6. The structural invertibility of (a) Erdős-Rényi random graphs and (b) scale free networks (power law exponent $\gamma = 2.4$) depends on the average node degree $\bar{d}$. For each data point, an ensemble of 100 networks with $N = 10^3$ nodes was simulated and disjoint sets of $M = P$ input and output nodes were chosen by uniform random sampling. Each network was tested for structural invertibility and the fraction $\rho$ of invertible networks in the ensemble was recorded. For large networks with many nodes $N \to \infty$ and fixed average degree $\bar{d}$, the function $\rho(\bar{d}, N, M)$ asymptotically approaches $\rho(\bar{d}, M)$. We found empirically that $N = 10^3$ is a good approximation for this asymptotic regime.

![Figure 7](image-url)  
Figure 7. Effect of the degree distribution on invertibility under the uniform random scheme. (a) The fraction of structural invertible input-output configurations for the E.coli metabolic network (black dots) is compared to an ensemble of scale free random networks with the same power law exponent $\gamma = 2.61$ and the same average degree $\bar{d} = 11.17$. (b,c) The out-degree versus the in-degree for the E.coli metabolic network (b) and a typical scale free network (c).
under the uniform input-output sampling scheme (scenario SC I).

To further explore the role of hubs in networks with a more symmetric assignment of in- and output nodes, we modified the uniform random scheme. Instead of uniform sampling (see IV A), we now ranked all the state nodes according to their degree and selected the M nodes with the highest degree as input nodes and the P = 10 output nodes are sampled uniformly from the remaining N − M = 490 nodes. Conversely, for the orange symbols, the P = 10 nodes with highest degree were chosen as outputs and the inputs were uniformly sampled from the remaining N − P = 490 nodes. The black dots were obtained for the same uniform random scheme as in Fig. 6, were both inputs and outputs were sampled uniformly (scenario SC I).

V. SENSOR NODE PLACEMENT FOR INVERTIBILITY

Whilst the uniform random scheme (scenario SC I) provides some insights into the effects of network properties on invertibility, it is not a very efficient strategy to randomly place the sensor nodes over the network.

A second, more realistic scenario (SC II) is the following: Assume, we have observed a systematic discrepancy between the output measurements and the model and we want to infer the unknown inputs (or model errors). Assume further, that the input node set S is given, either from domain knowledge about the respective system or from educated guessing about possible positions for input signals or model errors. However, the system might not be invertible with the current output node set. Typically, we know which states could in principle be measured and we can define a maximum set Z₀ of potential sensor nodes. If the resulting system with the maximum output set Z₀ is invertible, one can start the acquisition of time series data and feed them into one of the algorithms [9–15], to infer the input. This approach, though straightforward, would potentially be wasteful or even impractical. In domains like biology or economics, measurements might in principle be possible, but costly or take a great deal of time. Thus, a more feasible approach is to reduce this excessive effort by selecting a minimum set of sensor nodes from the maximum set Z₀.

A similar sensor node placement problem for state observability [19, 44] has been investigated before. In this

![Figure 8. Preferential sampling of hubs as input nodes (blue) or output nodes (orange) for an ensemble of 300 scale free networks (N = 500, M = P = 10, γ = 2.4). The blue triangles show the fraction of invertible networks, when the M = 10 nodes with highest degree are chosen as input nodes and the P = 10 output nodes are sampled uniformly from the remaining N − M = 490 nodes. Conversely, for the orange symbols, the P = 10 nodes with highest degree were chosen as outputs and the inputs were uniformly sampled from the remaining N − P = 490 nodes. The black dots were obtained for the same uniform random scenario as in Fig. 6, were both inputs and outputs were sampled uniformly (scenario SC I).]
Figure 9. Structural invertibility of real networks. (a) The probability of invertibility under the random scheme. For each network in Table I, the disjoint sets of $M = P$ in- and output nodes were obtained by uniform random sampling (scenario SC I). This was repeated 100 times for each $M$ and the fraction $\rho$ of structural invertible graphs was estimated. (b) Comparison of $\rho$ from (a) to the probability of invertibility after a degree-preserving randomization ($\rho_{\text{rand}}$). Note that some of the symbols are exactly masking each other. In this randomisation, the in- and out-degree of each node is preserved and the nodes linked to each other are randomly selected [59].

| Name            | $N$  | $|E|$ | $\bar{d}$ | Scale free | $\gamma$ | Brief Description                               | Database        |
|-----------------|------|------|-----------|------------|----------|------------------------------------------------|----------------|
| Regulatory      |      |      |           |            |          |                                                 |                |
| TRN-Yeast-2     | 688  | 1079 | 3.14      | True       | 2.29     | Transcriptional regulatory network of S.cerevisiae | Uri Alon Lab [60] |
| Ownership-USCorp| 7253 | 6726 | 1.85      | True       | 2.45     | Ownership network of US corporations            | Pajek [6]       |
| Trust           |      |      |           |            |          |                                                 |                |
| WikiVote        | 7115 | 103689 | 29.13    | False      |          | Who-vote-whom network of Wikipedia users        |                 |
| Epinions        | 75888 | 508837 | 13.41    | True       | 1.73     | Who-trust-whom network of Epinions.com users     |                 |
| Food Web        |      |      |           |            |          |                                                 |                |
| Little Rock     | 183  | 2494 | 27.26     | False      |          | Food Web in Little Rock lake                    | Mount Sinai [65] |
| Metabolic       |      |      |           |            |          |                                                 |                |
| E.coli          | 1039 | 5802 | 11.17     | True       | 2.61     | Network of the metabolic reactions of the E. coli bacteria | BiGG [66]       |
| Neuronal        |      |      |           |            |          |                                                 |                |
| C.elegans       | 297  | 2345 | 15.79     | True       | 1.89     | Neural network of C.elegans                     | Network Repository [5] |
| Citation        |      |      |           |            |          |                                                 |                |
| ArXiv-HepTh     | 27770 | 352807 | 25.41    | False      |          | Citation networks in HEP-TH category of Arxiv   | snap Stanford [8] |
| ArXiv-HepPh     | 34364 | 421578 | 24.41    | False      |          | Citation networks in HEP-PH category of Arxiv   | snap Stanford [8] |
| WWW             |      |      |           |            |          |                                                 |                |
| Political blogs | 1224 | 19025 | 31.09     | True       | 1.04     | Hyperlinks between weblogs on US politics       | Moreno [70]     |
| Internet        |      |      |           |            |          |                                                 |                |
| p2p-1           | 10876 | 39794 | 7.36      | False      |          | Gnutella peer-to-peer file sharing network      | snap Stanford [8] |
| p2p-2           | 8846  | 31839 | 7.2       | False      |          | Same as above (at different time)              | snap Stanford [8] |
| p2p-3           | 8717  | 31525 | 7.2       | False      |          | Same as above (at different time)              | snap Stanford [8] |
| Social Communication | |      |           |            |          |                                                 |                |
| UCIonline       | 1899 | 20296 | 21.38     | True       | 1.33     | Online message network of students at UC, Irvine | Opsahl [73]     |
| EmailKiel       | 57194 | 103731 | 3.63     | True       | 1.77     | Email network of traffic data collected at University of Kiel, Germany Barabasi [19] |
| Intraorganizational | |      |           |            |          |                                                 |                |
| Manufacturing   | 77   | 2228 | 3.14      | False      |          | Social network from a manufacturing company     | Opsahl [73]     |
| Consulting      | 46   | 879  | 38.22     | False      |          | Social network from a consulting company        | Opsahl [73]     |

Table I. A compilation of networks from various fields, also examined by other authors ([19]). Here $N$ is the number of nodes and $|E|$ the number of edges. The column “Scale free” indicates whether the degree distribution shows a power law and if so, the power law exponent $\gamma$ was computed.

In this section, we present a very simple but efficient greedy algorithm to select a minimum set of sensor nodes for invertibility for a given fixed set of input nodes (scenario SC II). This algorithm can drastically improve the chances for invertibility, as we will demonstrate by comparing to uniform random sampling for scenario SC I. Finally, we will also investigate a third scenario SC III, where the input nodes can also be selected.
A. Sensor node placement algorithm

Let us formalise the scenario SC II motivated above: The influence graph \( g \) for dynamic system (19) including a set \( S \) of \( M \) potential input nodes is assumed to be given. In addition, we have an initial maximum set

\[
Z_0 = \{z_1, \ldots, z_{P_0}\} \subseteq \{x_1, \ldots, x_N\}
\]

(33)
of \( P_0 \) potential output or sensor nodes. Thus, \( Z_0 \) incorporates all systems states which could potentially be monitored. If the system with \( S \) as given input set and \( Z_0 \) as maximum output set is not invertible, then there is no way to reconstruct the inputs from the outputs. However, if invertibility is ensured for \( Z_0 \), then we want to reduce this maximum set to a smaller, potentially minimal subset \( Z^* \subseteq Z_0 \) with \( P^* \) outputs, which is still invertible, given the inputs \( S \).

From the structural invertibility theorem (see Subsec. III E) we know that the smallest output set has at least as many nodes as the input set. Thus, we will always have \( P^* \geq M \). For small sets \( Z_0 \), it might be possible to try all \( \binom{P_0}{M} \) possible subsets of \( M \) nodes from the maximum set \( Z_0 \). However, this brute force strategy becomes quickly infeasible, if \( P_0 \) is large. Reducing \( P_0 \) from the beginning is usually also not an option, since the maximum sensor node set \( Z_0 \) needs to provide an invertible system, which might not be the case for small sets.

A practical solution is given by a simple greedy algorithm, which assumes that the triple \( (S, g, Z_0) \) containing the maximum node set \( Z_0 \) is invertible. To initialise the algorithm, we assume that the nodes are in some random order in \( Z_0 \). In the first iteration, we select the first node \( z \) from \( Z_0 \) and try to delete it, but only if \( (S, g, Z_0 \setminus z) \) with \( P_0 - 1 \) sensor nodes is still invertible. If not, we keep \( z \) in the node set \( Z_1 := Z_0 \), i.e. we reject the deletion of this node. Otherwise we delete the node by setting \( Z_1 := Z_0 \setminus z \). In any case, we continue and try to delete a different node, say the next node in \( Z_1 \). We proceed in this way until we have a sensor node set \( Z_k \) with \( P_k = M \) output nodes and set \( Z^* = Z_k \). This algorithm takes at most \( P_0 - M \) steps. Note, that the greedy algorithm will always find a minimum node set with the minimum number \( P^* = M \) of sensor nodes, provided \( (S, g, Z_0) \) with the initial node set \( Z_0 \) is invertible.

In Fig. 10 we present an example for a network with \( N = 1000 \) nodes and \( M = 100 \) uniformly sampled input nodes. All other nodes were included in the maximum output set \( Z_0 \), i.e. \( P_0 = 900 \). The algorithm takes 598 iterations to find a minimum node set \( Z^* \) with \( P^* = 100 \). The number of iterations can be reduced by replacing the random removal of output nodes by a more selective satellite-deletion strategy. By ranking the nodes in \( Z_0 \) according to their degree and selectively removing nodes with low degree shrinks the (invertible) output set much faster than random deletion. Reversing the order of the ranking, i.e. trying to selectively delete hubs from the set of sensor nodes results in more rejections and thus more iterations (Fig. 10). This is consistent with the results from (Fig. 8), were we found that preferential selection of high degree output nodes improves the chance for invertibility. Please note, that the greedy algorithm is usually fast enough for most purposes, even without ranking the nodes. This analyses merely serves to better understand the role of hubs as inputs or outputs.

B. Application to real networks

The sensor node placement algorithm can only be successful, if the maximum sensor node set \( Z_0 \) (together with the given input node set) yields an invertible system. Larger \( Z_0 \), i.e. a larger number \( P_0 \) of potentially measurable outputs, will obviously increase the chances for invertibility and thus also the chance to find a minimum sensor node set \( Z^* \) of cardinality \( P^* = M \). Apart from directly measuring all possible nodes, the largest possible nontrivial sensor node set \( Z_0 \) is given by the \( N - M \) nodes which are not input nodes. We used this \( Z_0 \) with \( P_0 = N - M \) to check the compendium of real networks (Table I) for invertibility.

For the results in Fig. 11(a), we sampled \( M \) input nodes \( S \) uniformly from all nodes and then selected \( Z_0 \) as the remaining \( N - M \) nodes (scenario SC I with the largest possible \( Z_0 \)). We repeated this over 100 randomly sampled input sets \( S \) and estimated the fraction \( \rho \) of invertible systems \((S, g, Z_0)\). Thus, \( \rho \) provides an estimate for the probability to obtain a structurally invertible system (for a given graph \( g \)) under scenario SC II, where the \( M \) input nodes \( S \) are given and cannot be chosen, but all the other nodes can in principle be measured. This is

![Figure 10. The greedy algorithm for optimal sensor node selection for invertibility applied to a network with \( N = 1000 \) state nodes, and \( M = 100 \) randomly distributed inputs. From the remaining 900 nodes, the maximum output set \( Z_0 \) with \( P_0 = 700 \) potential sensor nodes was randomly chosen. The plot compares the number of sensor nodes \( P \) versus the number of iterations of the greedy algorithm for random-, satellite-, and hub-deletion strategies.](image-url)
then identical to the fraction of systems, where the sensor node algorithm can reduce this initial sensor node set $Z_0$ to a minimum set $Z^*$ with $P^* = M$ outputs. By comparing Fig. 11(a) with Fig. 9(a) we can observe, that this strategy improves (in some cases drastically) the chances to find an invertible system. For better visibility see also Fig. 11(b), where we have plotted the difference $\Delta \rho$ between optimal sensor node placement in Fig. 11(a) and uniform output sampling Fig. 9(a).

C. Input node selection

So far we have assumed the input node set $S$ as given. To mimic this frequent situation that the input nodes cannot deliberately be selected, we performed uniform random sampling of the input nodes. However, there might be situations where we can influence the selection of inputs (scenario SC III). For the design of communication networks, the ability to uniquely distinguish different input signals is clearly a requirement and input node sets are often deliberately chosen [76]. Another example is given by the modular approach to model building, where one aims to describe a subsystem (or module) by a system of ODEs [1, 2]. This module will by definition receive inputs from the environment (see Fig. 1), which might not be directly measurable. Then, invertibility to infer these inputs from outputs is clearly an important requirement, which might influence decisions about the right state variables to include in the subsystem.

Based on the results of Fig. 8 we hypothesised that hubs with a high node degree are good candidates for inputs promoting invertibility. To test this hypothesis, we used again the networks listed in Table I. For each network $g$, we selected the $M$ nodes with highest out-degree as input node set $S$. As before, we used the remaining $N - M$ nodes as maximum sensor node set $Z_0$. If $(S, g, Z_0)$ is invertible, the sensor node selection algorithm can always reduce this to $(S, g, Z^*)$ with a minimum node set $Z^*$ having only $P^* = M$ outputs. Starting from $M = 1$ we increased the number of input nodes for each network one by one as long as the corresponding system $(S, g, Z_0)$ was invertible. The maximum number $M_{\text{max}}$ of inputs for each network is shown in Fig. 12, together with the probability of invertibility for the other scenarios using the uniform random scheme (data from Fig. 9(a)) and the sensor node placement algorithm (data from Fig. 11(a)). Please note, that for this hub input selection strategy all systems with $M < M_{\text{max}}$ inputs are invertible. Clearly, the hub input selection strategy provides a way to increase the number of input signals which can still be reconstructed in these networks.

VI. SUMMARY AND OPEN QUESTIONS

A. Summary and significance of the results

Reconstructing unknown inputs from outputs of open systems is useful in many settings. For modellers, the inputs provide important information about model errors and cues for model improvement or extension [12, 13, 34]. In biomedical systems, the unknown inputs can represent unmodelled environmental or physiological inputs, which might be interesting for the design of devices or measurement strategies. In electrical or secure networks, the unknown inputs could be attack signals, which need to be reconstructed and then mitigated. Unknown inputs can also be useful for improved state estimation [12–14] and data assimilation [53, 77]. Thus, from the viewpoint of modellers and engineers, invertibility is a desirable prop-
Figure 12. A heatmap comparing the probability of invertibility $\rho$ for the networks in Table I with three different node selection schemes as a function of the number of inputs. Scenario SC I, where neither inputs nor outputs can be selected, was simulated by uniform random sampling of input and output nodes (same data as in Fig. 9(a)). In scenario SC II we are able to select the output nodes and the results for the optimal sensor node placement from Fig. 11(a) are included for comparison. Under scenario SC III, we can select both, input and output nodes. We used hub input selection (selection of nodes with highest out-degree) in combination with optimal sensor node placement and found that $\rho$ is either one or zero. The numbers on the right indicate the maximum number $M_{\text{max}}$ of inputs which still provide an invertible system.

Although invertibility is desirable from an applied and analytic perspective, our results for an uniform random input selection scheme indicate that invertibility cannot be taken for granted, especially not in networks with low average degree, many inputs and with a scale free degree distribution. Thus, under the scenario SC I, we were neither inputs nor outputs can deliberately be chosen, many real networks have a disposition to mask differences between different input signals. It is well known that for example some dynamic biological systems respond often identically or similarly to a variety of different stimuli. Thus, living dynamic systems often distinguish different patterns rather than small differences in the inputs. In addition, real systems need to have a certain robustness against small perturbations and noise. Thus, invertibility is possibly not always a desirable property for the specific tasks to be performed by the network. Further research is needed to investigate tradeoffs between invertibility and other network traits, like e.g. controllability or robustness. Nevertheless, non-invertibility poses a challenge for experimentalists and modellers to reconstruct structural model errors and inputs from the environment.

We approached this problem by deriving an efficient sensor node placement algorithm, which extracts a minimum set of measurement nodes required for invertibility of a given network with a given input set (scenario SC II). In this scenario, the sensor node placement algorithm facilitates optimal experimental design for the reconstruction of inputs from outputs. As such, it can be used in conjunction with input reconstruction algorithms [9–13] and input observers [14, 15]. Structural invertibility pro-
vides a necessary condition for these algorithms to work.

In a third scenario SC III we assumed that both inputs and outputs can be selected. We found that selecting nodes with a high out-degree as input nodes, in combination with optimal output selection using the sensor node placement algorithm, drastically increases the number of inputs which can still be reconstructed from output measurements. Intuitively, these input hubs distribute the input signal widely over the network, therefore increasing the likelihood for finding node disjoint paths linking these inputs to the outputs. Although scenario SC III, where input nodes can deliberately be selected, might not always be realistic, it can be useful for the design of dynamic mathematical models or for the design of synthetic systems. For example, a key goal of Synthetic Biology is to engineer new biological systems for desired functionalities. In general, these systems are embedded in larger systems and will receive inputs from their environment, which should be inferable from measurements. In this case, optimal input selection in conjunction with optimal sensor node placement can provide important benefits. Another example is modular modelling, were an interest in subsystem embedded in a larger system is modelled in detail [1, 2]. To detect both potential model error or genuine inputs from the environment to the model, invertibility is essential. To achieve invertibility, it might be useful to include additional states which otherwise would not be deemed to be essential to understand the modular subsystem.

B. Limitations and open questions

Purely structural approaches to controllability and observability have been criticised to sometimes provide suboptimal conclusions for real systems. Depending on the quantitative properties of the interactions between the state nodes, a system might be practically uncontrollable or unobservable, even if the structural criteria are fulfilled [24–33]. Non-binary indices quantifying the degree of controllability and observability have been devised. These indices require at least the algebraic structure of the coupling functions between the state nodes [32] or even the full functional form and the parameters of the network [29].

A similar caveat applies to structural invertibility, which is only a condition for the existence of the inverse input-output map. As in any inverse problem [52], this might not be sufficient to actually implement this inverse map for reconstructing unknown inputs (including model errors) from outputs. Some unknown input signals might be hard to detect by noisy sensors with limited sensitivity. As discussed in Subsec. III G, these issues are related to the mathematical fact that the inverse of the compact input-output map is not continuous. Devising an index or condition number for the degree of invertibility (or continuity of the inverse) is therefore an important question for future research. Such an invertibility index might be used to rank sensor nodes, which could readily be utilised in a straightforward modification of our sensor node placement algorithm (Sec. V). The desired invertibility index might also be useful for improving or designing unknown input reconstruction algorithms [9–15], which adapt their regularisation automatically to the degree of discontinuity of the inverse input-output map.

A further potential objection against structural invertibility is that the influence graph has to be completely known. Indeed, if the aim is to assess invertibility of the true but unknown systems structure, we might obtain erroneous results if we use an incorrect influence graph with missing or spurious edges. However, the aim of systems inversion is to detect unknown inputs including systematic model errors. Missing or spurious interactions in the incorrect model graph are therefore causes of unknown inputs. To reconstruct the model errors, it is important that our potentially incorrect or incomplete model is invertible.

Currently, our invertibility results are limited to deterministic systems described by ODEs. However, the definition of invertibility doesn’t exclude stochastic unknown input functions. Nonlinear extensions of the Kalman-Filter [9–11] or fully probabilistic approaches [13] to reconstruct the moments or even the full probability distribution of the unknown input will only work for invertible systems. Their actual utility to simultaneously overcome the discontinuity of the inverse input-output map and to estimate probabilistic features of unknown inputs should systematically be explored.

Invertibility of systems with intrinsic process noise, which are often described by stochastic differential equations (SDEs), is a largely unexplored field. First, a modified probabilistic definition of invertibility is required for systems governed by SDEs. Second, the role of the different sources of noise for the invertibility needs to be investigated. Recent results for stochastic synchronisation [78, 79] indicate, that noise can have both, detrimental and beneficial effects. It would be exciting to investigate, whether similar effects are possible for unknown input reconstruction.

To conclude, invertibility (or the lack of it) has important implications for modelling frameworks and strategies to deal with incomplete and uncertain systems. Our analysis and algorithm for optimal experimental design are only a first step towards more sophisticated methods specifically tailored to handle systematic model errors and open systems. We believe that these approaches will increase our ability to better understand and manipulate complex systems, even if our knowledge will not be complete.

ACKNOWLEDGMENTS

This paper is part of the SEEDS project, which is funded by the Deutsche Forschungsgemeinschaft (DFG project number 354645666).
Appendix: Derivation of the Algebraic Criterion

For the sake of completeness let us explicitly write down the general mathematical setting for the linear case in our notation and deduce the algebraic criterion. In the general linear case, the initial state is given by a vector \( x_0 \in \mathbb{R}^N \), the dynamic of the system is given by \( A \in \mathbb{R}^{N \times N} \), and \( C \in \mathbb{R}^{P \times N} \) maps the system state to the output. There might be a known input \( u : [0, T] \to \mathbb{R}^M \) distributed by \( B \in \mathbb{R}^{N \times M'} \) over the state variables. Finally \( Dw \) models the structural model errors, so that we get the linear dynamic system,

\[
\begin{align*}
\dot{x}(t) &= Ax(t) + Bu(t) + Dw(t) & (A.1a) \\
x(0) &= x_0 & (A.1b) \\
y(t) &= Cx(t). & (A.1c)
\end{align*}
\]

Let \( \Phi : w \to y \) denote the solution operator, that maps the input \( w \) to the output \( y \) according to the dynamic system (A.1). For the dynamic system (A.1) we introduce the homogeneous system

\[
\begin{align*}
\dot{x}(t) &= Ax(t) + Dw(t) & (A.2a) \\
x(0) &= 0 & (A.2b) \\
y(t) &= Cx(t). & (A.2c)
\end{align*}
\]

with the solution operator \( \Phi^{\text{hom}} : w \to y \). Recall that a system is called invertible if for given data \( y^{\text{obs}} : [0, T] \to \mathbb{R}^P \) any solution of

\[
\Phi(w)(t) = y^{\text{obs}}(t) \quad \forall t \in [0, T] \quad (A.4)
\]

is unique. We will make use of the Volterra-operator

\[
V(w)(t) := \int_0^t w(s) \, ds. \quad (A.5)
\]

The Volterra-operator has the property

\[
V^n(w)(t) = \int_0^t \frac{(t-s)^{n-1}}{(n-1)!} w(s) \, ds \quad (A.6)
\]

and hence

\[
\sum_{n=0}^{\infty} CA^nDV^{n+1}(w)(t) = \int_0^t C \exp((t-s)A)Dw(s) \, ds, \quad (A.7)
\]

where the integration is understood component-wise.

Lemma 1. Let \( \Phi \) and \( \Phi^{\text{hom}} \) be the solution operators of a dynamic system as defined above. Then

1. \( \Phi^{\text{hom}} \) is linear, continuous, and compact,

2. \( \Phi(w + v) = \Phi(w) \) if and only if \( \Phi^{\text{hom}}(v) = 0 \).

Proof. 1. The homogeneous solution operator takes the form

\[
\Phi^{\text{hom}} = \sum_{n=0}^{\infty} (CA^nD)V^{n+1}. \quad (A.8)
\]

To see this, let

\[
x(t) = \sum_{n=0}^{\infty} A^nDV^{n+1}(w)(t) \quad (A.9)
\]

then

\[
\frac{d}{dt}x(t) = Dw(t) + A \sum_{n=1}^{\infty} A^{n-1}DV^n(w)(t) = Dw(t) + Ax(t) \quad (A.11)
\]

which shows that \( x \) solves the dynamic equations as well as \( x(0) = 0 \). Multiplication with \( C \) yields the solution of the homogeneous system

\[
y(t) = \Phi^{\text{hom}}(w)(t). \quad (A.12)
\]

Since \( V^n \) is linear, so is \( \Phi^{\text{hom}} \). As we make the restriction \( w_i \in L^2([0, T]) \) we get that \( \Phi^{\text{hom}} \) is Hilbert-Schmidt thus continuous and compact.

2. The inhomogeneous part of \( \Phi \) is given by

\[
\phi(t) := \sum_{n=0}^{\infty} CA^nDV^{n+1}(u)(t), \quad (A.13)
\]

such that the full solution can be written as

\[
\Phi(w)(t) = \phi(t) + \Phi^{\text{hom}}(w)(t) \quad (A.14)
\]

which shows that

\[
\Phi(w + v) = \Phi(w) + \Phi^{\text{hom}}(v) \quad (A.15)
\]

hence if and only if \( \Phi^{\text{hom}}(v) = 0 \) then \( v \) leaves the solution of the inhomogeneous system invariant.

As a result of the above lemma, we can set \( x_0 \) as well as \( u \) to zero. Furthermore this shows that \( \Phi \) given in Sec. III is indeed the relevant solution operator that has to be one-to-one. We now follow the proof of Sain and Massey [37]. After Laplace-transformation the dynamic equation becomes

\[
sL[x](s) = AC[x](s) + DL[w](s) \quad (A.16)
\]

with a complex variable \( s \in \mathbb{C} \). Using the transfer function

\[
T(s) := C(\mathbb{I}s - A)^{-1}D, \quad (A.17)
\]
where \( I \) is the identity in \( \mathbb{R}^N \), we get
\[
T(s) \mathcal{L}[w](s) = \mathcal{L}[y](s)
\]
thus \( T \) is the Laplace-transform of the solution operator \( \Phi \), and since \( \mathcal{L}[w] \) is the zero function if and only if \( w \) is the zero-function (almost everywhere), we know that \( \Phi \) is one-to-one if and only if \( T \) has rank \( T(s) = M \) for \( s \in \mathbb{C} \) (almost everywhere), i.e. if from \( \mathcal{L}[y] = 0 \) it follows, that \( \mathcal{L}[w] = 0 \). We make the assumption, that \( u \) is smooth, which is equivalent to the assumption, that it can be written as a Laurent-series (comprising only the principal part)
\[
\mathcal{L}[w](s) = \frac{1}{s} \sum_{k=0}^{\infty} \frac{1}{s^k} \xi_k
\]
in Laplace-space, where \( (\xi_k)_k \) is a sequence of \( \mathbb{R}^M \) vectors. Using the Neumann-series yields
\[
C(\|s - A\|^{-1} D = \frac{1}{s} \sum_{l=0}^{\infty} C A^l D.
\]
If \( \mathcal{L}[y] = 0 \), then
\[
\frac{1}{s^2} \sum_{k,l=0}^{\infty} \frac{1}{s^{k+l}} C A^l D \xi_k = 0
\]
and since \( \{1, s^{-1}, s^{-2}, \ldots\} \) are linearly independent in function space, by equating coefficients for each \( n \in \mathbb{N}_0 \) we find
\[
\sum_{k=0}^{\infty} C A^k D \xi_{n-k} = 0.
\]
It is now convenient to define
\[
R_n := \begin{bmatrix} CD & CAD & \ldots & CA^l D \end{bmatrix}
\]
as known from the Kalman controllability matrix, as well as
\[
\Xi_n := \begin{bmatrix} \xi_n \\ \vdots \\ \xi_0 \end{bmatrix}
\]
to finally get
\[
R_n \Xi_n = 0 \forall n \in \mathbb{N}_0.
\]
Thus, the dynamic system is invertible if and only if we find a sequence \( (\xi_k)_k \) such that (A.25) holds. If we combine \( R_0, R_1, \ldots, R_l \) to one matrix
\[
Q_l := \begin{bmatrix} CD & CAD & \ldots & CA^l D \\ 0 & CD & \ldots & CA^{l-1} D \\ \vdots & \ddots & \vdots \\ 0 & \ldots & CD \end{bmatrix}
\]
we find, that (A.25) is equivalent to
\[
\text{rank } Q_{N-1} - \text{rank } Q_{N-2} = M,
\]
the criterion stated by Sain and Massey. From (A.25) we directly see, that \( \Xi_l = 0 \) and \( \Xi_{l+1} = \xi_{l+1} \Xi_l^T \), thus
\[
\Xi_{l+1} \in \ker R_{l+1} \cap (\mathbb{R}^M \times \ker R_l).
\]
If we now exclude the trivial solution \( \xi_l = 0 \) for all \( l \), this motivates
\[
K_0 := \ker R_0 \setminus \{0\}
\]
and
\[
K_l := \ker R_{l+1} \cap (\mathbb{R}^M \times K_l)
\]
As we iterate though \( K_0, K_1, \ldots \), as long as \( K_l \neq \emptyset \) there is a non-trivial solution of
\[
R_l \Xi_k = 0 \quad \text{for } k \leq l.
\]
Hence, if and only if we find a \( l \in \mathbb{N}_0 \), such that \( R_l = \emptyset \), then the dynamic system is invertible. From (A.27) one can see, that is suffices to check only \( l \leq N - 1 \). In addition to that we find the following theorem.

**Theorem 1.** The solution space of
\[
\Phi(w) = y^{\text{obs}}
\]
is either zero- or infinite-dimensional.

**Proof.** From the considerations above it is clear, that we have to show that the solution space of (A.25) is either zero- or infinite-dimensional, i.e. if there is a non-trivial sequence \( (\xi_k)_k \), then there is an infinite-dimensional vector space of such sequences.

First, assume \( (\xi_k)_k \) is a sequence, that solves (A.25) and \( \xi_0 = 0 \). We define another sequence \( (\chi_k)_k \) by \( \chi_k := \xi_{k+1} \), i.e. \( (\chi_k)_k \) is the left shift of \( (\xi_k)_k \). Let \( X_l \) the vector \( [\chi_l, \ldots, \chi_0]^T \) analogous to \( \Xi_l \). Then
\[
0 = R_l \Xi_l = R_{l-1} X_{l-1}
\]
for all \( l \). Hence \( (\chi_k)_k \) is also a solution. This shows, that if \( (\xi_k)_k \) solves (A.25), then we can without loss of generality assume \( \xi_0 \neq 0 \).

Now let \( (\xi_k)_k \) a solution and define \( \chi_0 := 0 \) and \( \chi_k = \xi_{k-1} \), i.e. \( (\chi_k)_k \) is the right shift of \( (\xi_k)_k \). Then
\[
R_0 X_0 = 0
\]
is clear, and for \( l \geq 1 \)
\[
R_l X_l = R_{l-1} \Xi_{l-1} = 0
\]
hence the sequence \( (\chi_k)_k \) solves (A.25). Let henceforth \( \xi \rightarrow \) denote the right shift.
Since matrix multiplication is a linear operation it is clear, that if \((\xi_k)_k\) and \((\chi_k)_k\) solve (A.25), so does \((\xi_k + \chi_k)_k\) as well as \((a\xi_k)_k\) for an arbitrary real number \(a\). Therefore the space of sequences that solve (A.25) is a real vector space, denoted \(\mathcal{K}\). Let \((\xi_k)_k \in \mathcal{K}\) with \(\xi_0 \neq 0\). Then \(\mathcal{G}_1^2(\xi_k)_k \in \mathcal{K}\) and

\[
\{ (\xi_k)_k, \mathcal{G}_1(\xi_k)_k, \mathcal{G}_1^2(\xi_k)_k, \ldots \} \tag{A.36}
\]

is a set of infinitely many linearly independent vectors in \(\mathcal{K}\), hence

\[
\dim \mathcal{K} = \infty. \tag{A.37}
\]

[1] 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 Dynamic Modeling in Systems Biology, PLOS ONE 8, 1 (2013).
[2] M. Almgard and A. Korngreen, Is realistic neuronal modeling realistic?, Journal of Neurophysiology 116, 2180 - 2196 (2016).
[3] 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 (2017).
[4] A. Tsigkinopoulou, S. M. Baker, and R. Breitling, Respectful Modeling: Addressing Uncertainty in Dynamic System Models for Molecular Biology, Trends in Biotechnology 35, 518 (2017).
[5] 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 (AAAI Press, 2015) pp. 4292–4293.
[6] V. Batagelj and A. Mrvar, Pajek datasets, http://vlado.fmf.uni-lj.si/pub/networks/data/, access date 2018-12-16.
[7] J. Kunegis, KONECT - The Koblenz Network Collection, in Proc. Int. Web Observatory Workshop (2013) pp. 1343–1350.
[8] J. Leskovec and A. Krevl, SNAP Datasets: Stanford Large Network Dataset Collection, http://snap.stanford.edu/data, access date 2018-12-16.
[9] P. Kühn, M. Diehl, T. Kraus, J. P. Schöder, and H. G. Bock, A real-time algorithm for moving horizon state and parameter estimation, Computers & Chemical Engineering 35, 71 (2011).
[10] A. Raue, C. Kreutz, M. Schelker, and J. Timmer, Comprehensive estimation of input signals and dynamics in biochemical reaction networks, Bioinformatics 28, i529 (2012).
[11] 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 10th UKACC International Conference on Control (IEE, Loughborough, United Kingdom, 2014) pp. 19–24.
[12] B. Engelhardt, H. Fröhlich, and M. Kschischang, Learning (from) the errors of a systems biology model, Scientific Reports 6, 20772 (2016).
[13] B. Engelhardt, M. Kschischang, 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 (2017).
[14] 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 (2017).
[15] 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 (2018).
[16] A. Elphrime, J. A. Lerman, J. Lu, P. Miller, S. Federowicz, Z. A. King, A. Drger, B. O. Palsson, and N. E. Lewis, BiGG Models: A platform for integrating, standardizing and sharing genome-scale models, Nucleic Acids Research 44, D515 (2015).
[17] C.-T. Lin, Structural controllability, IEEE Transactions on Automatic Control 19, 201 (1974).
[18] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, Controllability of complex networks, Nature 473, 167 (2011).
[19] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, Observability of complex systems, in Proceedings of the National Academy of Sciences, Vol. 110 (2013) pp. 2460–2465.
[20] J. Gao, Y.-Y. Liu, R. M. D’Souza, and A.-L. Barabási, Target control of complex networks, Nature Communications 5, 5415 (2014).
[21] A. E. Motter, Networkcontrology, Chaos: An Interdisciplinary Journal of Nonlinear Science 25, 097621 (2015).
[22] Y.-Y. Liu and A.-L. Barabási, Control principles of complex systems, Rev. Mod. Phys. 88, 035006 (2016).
[23] A. Sharma, C. Cinti, and E. Capobianco, Multitype Network-Guided Target Controllability in Phenotypically Characterized Osteosarcoma: Role of Tumor Microenvironment, Frontiers in Immunology 8, 918 (2017).
[24] A. J. Krener and I. Kayo, Measures of unobservability, IEEE Conf. Decis. Control (2009).
[25] F. Lo Indice, F. Garofalo, and F. Sorrentino, Structural permeability of complex networks to control signals, Nature Communications 6, 8349 (2015).
[26] I. Klickstein, A. Shirin, and F. Sorrentino, Energy scaling of targeted optimal control of complex networks, Nature Communications 8, 15145 (2017).
[27] S. P. Cornelius, W. L. Kath, and A. E. Motter, Controlling Complex Networks with Compensatory Perturbations, arXiv:1105.3726 (2011).
[28] J. Sun and A. E. Motter, Controllability Transition and Nonlocality in Network Control, Phys. Rev. Lett. 110, 208701 (2013).
[29] S. P. Cornelius, W. L. Kath, and A. E. Motter, Realistic control of network dynamics, Nature Communications 4, 1942 (2013).
[30] 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 (2015).

[31] 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 (2016).

[32] L. A. Aguirre, L. L. Leonardo, and C. Letellier, *Structural, dynamical and symbolic observability: From dynamical systems to networks*, PLoS ONE **13**, 1 (2018).

[33] A. Haber, F. Molnar, and A. E. Motter, *State Observation and Sensor Selection for Nonlinear Networks*, IEEE Transactions on Control of Network Systems **5**, 694 (2018).

[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 (1988).

[35] R. W. Brocket and M. D. Mesarovici, *The reproducibility of multivariable systems*, Journal of Mathematical Analysis and Applications **11**, 548 (1965).

[36] L. Silverman, *Inversion of multivariable linear systems*, IEEE Transactions on Automatic Control **14**, 270 (1969).

[37] M. Sain and J. Massey, *Invertibility of linear time-invariant dynamical systems*, IEEE Transactions on Automatic Control **14**, 141 (1969).

[38] R. M. Hirschorn, *Invertibility of Nonlinear Control Systems*, SIAM Journal on Control and Optimization **17**, 289 (1979).

[39] H. Nijmeijer, *Invertibility of affine nonlinear control systems: A geometric approach*, Systems & Control Letters **2**, 163 (1982).

[40] H. Nijmeijer, *Right-invertibility for a class of nonlinear control systems: A geometric approach*, Systems & Control Letters **7**, 125 (1986).

[41] M. Fliess, *A note on the invertibility of nonlinear input-output differential systems*, Systems & Control Letters **8**, 147 (1986).

[42] T. Wey, *Rank and Regular Invertibility of Nonlinear Systems: A Graph Theoretic Approach*, IFAC Proceedings Volumes **31**, 257 (1998).

[43] J.-M. Dion, C. Commault, and J. van der Woude, *Generic properties and control of linear structured systems: a survey*, Automatica **39**, 1125 (2003).

[44] T. Boukhobza, F. Hamelin, and S. Martinez-Martinez, *State and input observability for structured linear systems: A graph-theoretic approach*, Automatica **43**, 1204 (2007).

[45] 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 (2014).

[46] 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 (2002).

[47] A. Martinelli, *Nonlinear Unknown Input Observability: The General Analytic Solution*, IEEE Trans. Automat. Contr. **64**, 222 (2019), arXiv: 1704.03252.

[48] K. Menger, *Zur allgemeinen Kurventheorie*, Fund. Math. **10**, 96 (1927).

[49] B. Korte and J. Vygen, *Combinatorial optimization: theory and algorithms*, sixth edition ed., Algorithms and combinatorics No. 21 (Springer, Berlin, Germany, 2018).

[50] R. T. A. Goldberg, *A new approach to the maximum flow problem*, Journal of the ACM. **35**, 921 (1988).

[51] H. Engl, M. Hanke, and A. Neubauer, *Regularization of Inverse Problems* (Springer Netherlands, Kluwer Academic Publishers, Dordrecht, 2000).

[52] G. Nakamura and R. Potthast, *An introduction to the theory and methods of inverse problems and data assimilation*, IOP Expanding Physics (Institute of Physics Publishing, 2015).

[53] H. D. I. Abarbanel, *Predicting the future: completing models of observed complex systems*, Understanding complex systems (Springer, New York, NY, 2013).

[54] P. Erdős and A. Rényi, *On the evolution of random graphs*, Publ. Math. Inst. Hung. Acad. Sci. **5**, 17 (1960).

[55] A.-L. Barabási and R. Albert, *Emergence of scaling in random networks*, Science **286**, 509 (1999).

[56] K.-I. Goh, B. Kahng, and D. Kim, *Universal Behavior of Load Distribution in Scale-Free Networks*, Phys. Rev. Lett. **87**, 278701 (2001).

[57] R. Cohen and S. Havlin, *Scale-Free Networks Are Ultrasmall*, Physical Review Letters **90**, 058701 (2003).

[58] J. Schellenberger, J. O. Park, T. M. Conrad, and B. O. Palsson, *BiGG: A Biochemical Genetic and Genomic knowledgebase of large scale metabolic reconstructions*, BMC Bioinformatics **11**, 213 (2010).

[59] S. Maskov and K. Sneppen, *Specificity and Stability in Topology of Protein Networks*, Science **296**, 910 (2002).

[60] U. Alon, *Collection of complex networks*, http://www.weizmann.ac.il/mcb/UriAlon/download/collection-complex-networks, access date 2018-12-16.

[61] 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), Seoul Korea (2002) pp. 27–129.

[62] 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 (2009).

[63] M. Richardson, R. Agrawal, and P. Domingos, *Trust Management for the Semantic Web*, in *The Semantic Web - ISWC 2003*, edited by D. Fensel, K. Sycara, and J. Mylopoulos (Springer Berlin Heidelberg, Berlin, Heidelberg, 2003) pp. 351–368.

[64] N. D. Martinez, *Artifacts or Attributes? Effects of Resolution on the Little Rock Lake Food Web*, Ecological Monographs **61**, 367 (1991).

[65] M. S. Ma’ayan Laboratory Department of Pharmacology and S. Therapeutics, http://research.mssm.edu/maayan/datasets/qualitative_networks.shtml, access date 2018-12-16.

[66] BiGG Models, http://bigg.ucsd.edu, access date 2018-12-16.

[67] D. J. Watts and S. H. Strogatz, *Collective dynamics of ‘small-world’ networks*, Nature **393**, 440 (1998).

[68] 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* (ACM Press, 2005) p. 177.

[69] L. A. Adamic and N. Glance, *The political blogosphere and the 2004 U.S. election: divided they blog*, in Proceed-
ings of the 3rd international workshop on Link discovery - LinkKDD ’05 (ACM Press, 2005) pp. 36–43.

[70] Datasets, http://moreno.ss.uci.edu/data.html#blogs, access date 2018-12-16 ().

[71] J. Leskovec, J. Kleinberg, and C. Faloutsos, Graph evolution: Densification and shrinking diameters, ACM Transactions on Knowledge Discovery from Data 1, 2 (2007).

[72] T. Opsahl and P. Panzarasa, Clustering in weighted networks, Social Networks 31, 155 (2009).

[73] Datasets, https://toreopsahl.com/datasets/, access date 2018-12-16 ().

[74] H. Ebel, L.-I. Mielsch, and S. Bornholdt, Scale-free topology of e-mail networks, Physical Review E 66, 10.1103/PhysRevE.66.035103 (2002).

[75] 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, Mass, 2004).

[76] C. Larson, Design of modern communication networks: methods and applications, 1st ed. (Academic Press/Elsevier, Amsterdam, 2014) oCLC: 907438989.

[77] S. Reich and C. Cotter, Probabilistic forecasting and Bayesian data assimilation (Cambridge University Press, Cambridge, 2015).

[78] G. Russo and R. Shorten, On common noise-induced synchronization in complex networks with state-dependent noise diffusion processes, Physica D Nonlinear Phenomena 369, 47 (2018).

[79] D. A. Burbano-L., G. Russo, and M. D. Bernardo, Pinning Controllability of Complex Network Systems With Noise, IEEE Transactions on Control of Network Systems 6, 874 (2019).