Koopman-based lifting techniques for nonlinear systems identification

Mauroy, Alexandre; Goncalves, Jorge

Published in:
IEEE Transactions on Automatic Control

DOI:
10.1109/TAC.2019.2941433

Publication date:
2020

Document Version
Peer reviewed version

Link to publication
Citation for published version (HARVARD):
https://doi.org/10.1109/TAC.2019.2941433
Koopman-based lifting techniques for nonlinear systems identification

A. Mauroy and J. Goncalves

Abstract—We develop a novel lifting technique for nonlinear system identification based on the framework of the Koopman operator. The key idea is to identify the linear (infinite-dimensional) Koopman operator in the lifted space of observables, instead of identifying the nonlinear system in the state space, a process which results in a linear method for nonlinear systems identification. The proposed lifting technique is an indirect method that does not require to compute time derivatives and is therefore well-suited to low-sampling rate datasets.

Considering different finite-dimensional subspaces to approximate and identify the Koopman operator, we propose two numerical schemes: a main method and a dual method. The main method is a parametric identification technique that can accurately reconstruct the vector field of a broad class of systems. The dual method provides estimates of the vector field at the data points and is well-suited to identify high-dimensional systems with small datasets. The present paper describes the two methods, provides theoretical convergence results, and illustrates the lifting techniques with several examples.

I. INTRODUCTION

The problem of identifying the equations of a continuous-time dynamical system from time-series data has attracted considerable interest in many fields such as biology, finance, and engineering. It is also closely related to network inference, which aims at reconstructing the interactions between the different states of a system, a problem of paramount importance in systems biology. In many cases, the identification problem is more challenging due to the nonlinear nature of the systems and must be tackled with black-box methods (e.g. Wiener and Volterra series models [1], nonlinear auto-regressive models [2], neural network models [3], see also [4], [5] for a survey). In line with the classic approach to system identification, these methods typically deal with (long, highly-sampled) time-series and provide a relationship between the system inputs and outputs.

In the different but related context of nonlinear parameter estimation, a wealth of methods have been developed to identify the state dynamics of autonomous systems with a known structure. Typical methods seek the best linear combination of time derivatives of the state over a set of library functions (similar to the basis functions used in black-box models) [6]. Similar approaches have also been proposed recently, partly motivated by the network identification problem (e.g. Bayesian approach [7], SINDy algorithm [8]). The above-mentioned methods are direct methods, and their main advantage is that they rely on static linear regression techniques. However, they assume that time derivatives of the state can be accurately estimated (e.g. by using collocation techniques), a requirement that becomes prohibitive when the sampling time is too low, the measurements too noisy, or the time-series too short. Instead, indirect methods solve an initial value problem and do not require to estimate time derivatives [9], [10]. Hence, they offer a good alternative to direct methods, but at the expense of solving a (nonconvex) nonlinear least squares problem. The goal of this paper is to propose a new indirect method for nonlinear system identification/parameter estimation, a method which not only circumvents the estimation of time derivatives but also relies on linear least squares optimization.

The approach proposed in this paper is based on the framework of the so-called Koopman operator [11], [12]. The Koopman operator is a linear infinite-dimensional operator that describes the evolution of observable-functions along the trajectories of the system. Starting with the seminal work of [13], several studies have investigated the interplay between the spectral properties of the operator and the properties of the associated system, a body of work that has led to new methods for the analysis of nonlinear systems (e.g. global stability analysis [14], global linearization [15], monotone systems [16], delayed systems [17]). While the above-mentioned studies focus on systems described by a known vector field, the Koopman operator approach is also conducive to data analysis and directly connected to numerical schemes such as Dynamic Mode Decomposition (DMD) [18], [19], [20], [21]. This yielded another set of techniques for data-driven analysis and control of nonlinear systems (observer synthesis [22], model predictive control [23], optimal control [24], power systems stability analysis [25], to list a few). In this context, this paper aims at connecting data to vector field, thereby bridging these two sets of methods.

The Koopman operator provides a linear representation of the nonlinear system in a lifted (infinite-dimensional) space of observable-functions. Our key idea is to exploit this lifting approach and identify the linear Koopman operator in the space of observables, instead of identifying the nonlinear system in the state space. Our numerical scheme proceeds in three steps: (1) lifting of the data, (2) identification of the Koopman operator, and (3) iden-
The first step is to define the vector field. In the first step, snapshot data are lifted to the space of observables. In the second step, we derive two distinct methods: (1) a main method which identifies a representation of the Koopman operator in a basis of functions; (2) a dual method which identifies the representation of the operator in the “sample space”. The first two steps are directly related to a component of the Extended Dynamic Mode Decomposition (EDMD) [26] (main method) or inspired from kernel-based EDMD [27] (dual method). In the third step, we connect the vector field to the infinitesimal generator of the identified operator and solve a linear least squares problem to compute the linear combination of the vector field in a basis of library functions. The two methods are complemented with convergence results showing that they identify the vector field exactly in optimal conditions. The main method has been initially proposed in [28] and a similar approach developed in a stochastic framework can also be found in the recent work [29].

The proposed lifting technique has several advantages. First of all, it relies only on linear methods which are easy and efficient to implement. It is also well-suited to data acquired from short time-series with low sampling rates (e.g. data pairs generated from multiple trajectories). Although initially limited to polynomial vector fields, the main method works efficiently with a broad class of behaviors, including unstable and chaotic systems. In addition, the dual method is well-suited to identify large-dimensional systems and to reconstruct network topologies, in particular when the number of sample points is smaller than the unknown system parameters. Finally, lifting techniques can be extended to identify non-polynomial vector fields and open systems (with input or process noise).

The rest of the paper is organized as follows. In Section II, we present the problem and introduce the general lifting technique used for system identification. Section III describes the main method and provides theoretical convergence results, while Section IV discusses some extensions of the methods to non-polynomial vector fields and open systems. In Section V, we propose the dual method to identify high-dimensional systems with small datasets and give convergence proofs. The two methods are illustrated with several examples in Section VI, where the network reconstruction problem is also considered. Concluding remarks and perspectives are given in Section VII.

II. IDENTIFICATION IN THE KOOPMAN OPERATOR FRAMEWORK

A. Problem statement

We address the problem of identifying the vector field of a nonlinear system from time series generated by its dynamics. We consider the system

$$\dot{x} = F(x), \quad x \in \mathbb{R}^n$$  \hspace{1cm} (1)

where the vector field $F(x)$ is of the form

$$F(x) = \sum_{k=1}^{N_F} \mathbf{w}_k h_k(x).$$  \hspace{1cm} (2)

The vectors $\mathbf{w}_k = (w_1^k \cdots w_n^k)^T \in \mathbb{R}^n$ are unknown coefficients (to be identified) and the library functions $h_k$ are assumed to be known. Note that some coefficients might be equal zero. Unless stated otherwise, we will consider that the vector field is polynomial, so that $h_k$ are monomials: $h_k = p_k$ with $p_k(x) \in \{x_1^{s_1} \cdots x_n^{s_n}|(s_1, \ldots, s_n) \in \mathbb{N}^n, s_1 + \cdots + s_n \leq m_F\}$  \hspace{1cm} (3)

where $m_F$ is the total degree of the polynomial vector field. The number of monomials in the sum (2) is given by $N_F = (m_F + n)!/(m_F!n!)$. As shown in Section IV-C, the proposed method can also be generalized to other types of vector fields in a straightforward way.

Our goal is to identify the vector field $F$ (i.e. the $N_F$ coefficients $\mathbf{w}_k$) from snapshot measurements of the system trajectories. We consider $K$ snapshot pairs $(\mathbf{x}_k, \mathbf{y}_k)$ obtained from noisy measurements (proportional to the exact state value): we have

$$\mathbf{x}_k = \bar{x}_k + \epsilon(\mathbf{x}_k) \quad \mathbf{y}_k = \bar{y}_k + \epsilon(\mathbf{y}_k)$$  \hspace{1cm} (4)

where $\epsilon$ is the state-dependent measurement noise, and

$$\bar{y}_k = \varphi^T_{\mathbf{x}_k}(\bar{x}_k)$$  \hspace{1cm} (5)

where $t \mapsto \varphi^t(\mathbf{x}_0)$ is the solution to (1) associated with the initial condition $\mathbf{x}_0$. We assume that the measurement noise is Gaussian and proportional to the state value, i.e. $\epsilon(\mathbf{x}) = \mathbf{x} \odot \mathbf{v}$ where $\odot$ denotes the element-wise product and $\mathbf{v}$ is a Gaussian random variable with zero mean and standard deviation $\sigma_{max}$. We also assume that all pairs $(\mathbf{x}_k, \mathbf{y}_k)$ lie in a compact set $X \subset \mathbb{R}^n$ and are obtained with the same sampling period $T_s$. They can belong to a single trajectory or to multiple trajectories. Stochastic systems with process noise and systems with inputs will also be considered (see Section IV).

Remark 1. For numerical reasons, we will assume in general that the data points lie in a set $X \subset [-1, 1]^n$. If original data do not satisfy this assumption, then they can be rescaled to yield new data pairs $(\mathbf{x}'_k, \mathbf{y}'_k) = (\mathbf{x}_k/\alpha, \mathbf{y}_k/\alpha) \in [-1, 1]^{2n}$. These new pairs enable to identify a vector field $F'(x)$ with coefficients $\mathbf{w}'_k = \alpha^{m_k-1}\mathbf{w}_k$, where $m_k$ is the total degree of the monomial $p_k$. 

B. Koopman operator

System (1) represents the state dynamics in $\mathbb{R}^n$. Alternatively, the system can be described in a lifted space $\mathcal{F}$ of observable functions $f : \mathbb{R}^n \rightarrow \mathbb{R}$. Provided that the observable functions are continuously differentiable, their dynamics in the lifted space are given by

$$\dot{f} = (F \cdot \nabla)f, \quad f \in \mathcal{F},$$  \hspace{1cm} (6)

where $\dot{f}$ denotes $\partial(f \circ \varphi^t)/\partial t$ (with a slight abuse of notation) and $\nabla$ denotes the gradient (see e.g. [30]). In

- \hspace{1cm}
contrast to (1), the dynamics (6) are infinite-dimensional but linear.

While the flow induced by (1) in the state space is given by the nonlinear flow map $\varphi$, the flow induced by (6) in the lifted space is given by the linear semigroup of Koopman operators $U^t: \mathcal{F} \rightarrow \mathcal{F}$, $t \geq 0$. This semigroup governs the evolution of the observables along the trajectories, i.e.

$$U^t f = f \circ \varphi^t.$$  

Under appropriate conditions (see Section III-C), the semigroup of Koopman operators is strongly continuous and given by the nonlinear flow map

$$U^t = e^{Lt}.$$  

The operator $L$ is called the infinitesimal generator of the Koopman operator and we denote its domain by $\mathcal{D}(L)$.

### C. Linear identification in the lifted space

There is a one-to-one correspondence between systems of the form (1) and lifted systems (6), or equivalently between the flow $\varphi^t$ and the semigroup of Koopman operators $U^t$. Exploiting this equivalence, we propose to solve the identification problem in the lifted space instead of the state space. This can be done in three steps (see Figure 1).

1) **Lifting of the data.** Snapshots pairs $(x_k, y_k)$ are lifted to the space of observable by constructing new pairs of the form $(g(x_k), g(y_k))$ for some $g \in \mathcal{F}$. The functions $g$ are assumed to be continuously differentiable and we call them basis functions. It follows from (4) and (5) that

$$g(y_k) = g(\varphi^T(x_k - \epsilon(x_k)) + \epsilon(y_k)) \approx U^T g(x_k) + O(||\epsilon||).$$  

2) **Identification of the Koopman operator.** A finite-dimensional projection of the Koopman operator is obtained through a classic linear identification method that is similar to a component of the Extended Dynamic Mode Decomposition (EDMD) algorithm [26]. This yields (an approximation of) the infinitesimal generator $L$ of the Koopman operator.

3) **Identification of the vector field.** Using (7), we can finally obtain the vector field $F$.

### III. The main lifting method

#### A. Description of the method

This section describes in detail the three steps of our main method. The first step and the first part of the second step are related to a component of the EDMD algorithm (see [26] for more details).

1) **First step - lifting of the data:** The data must be lifted to the infinite-dimensional space $\mathcal{F}$ of observables. However, the method has to be numerically tractable and is developed in a finite-dimensional linear subspace $\mathcal{F}_N \subset \mathcal{F}$ spanned by a basis of $N$ linearly independent functions. The choice of basis functions $\{g_k\}_{k=1}^N$ can be arbitrary (e.g. Fourier basis, radial basis functions), but might affect the method performances. Since the vector field is assumed to be polynomial, we naturally choose the basis of monomials $\{g_k\}_{k=1}^N$ with total degree less or equal to $m$ to facilitate the representation of the Koopman operator. The number of basis functions is equal to $N = (n + m)!/(n! m!)$.

For each snapshot pair $(x_k, y_k) \in \mathbb{R}^{n \times 2}$, $k \in \{1, \ldots, K\}$, we construct a new pair $(p(x_k), p(y_k)) \in \mathbb{R}^{N \times 2}$ where $p(x) = (p_1(x), \ldots, p_N(x))^T$ denotes the vector of basis monomials. In the following, we will also use the $K \times N$ matrices

$$P_x = \begin{pmatrix} p(x_1)^T \\ \vdots \\ p(x_K)^T \end{pmatrix}, \quad P_y = \begin{pmatrix} p(y_1)^T \\ \vdots \\ p(y_K)^T \end{pmatrix}.$$  

2) **Second step - identification of the Koopman operator:** Now we proceed to the identification of the Koopman operator $U^t$, for $t = T_s$. More precisely, we will identify the finite-rank operator $U_N : \mathcal{F}_N \rightarrow \mathcal{F}_N$ of the form $U_N = P_N U^T |_{\mathcal{F}_N}$, where $P_N : \mathcal{F} \rightarrow \mathcal{F}_N$ is a projection operator onto the subspace $\mathcal{F}_N$ and where $U^T |_{\mathcal{F}_N} : \mathcal{F}_N \rightarrow \mathcal{F}_N$ is the restriction of the Koopman operator to $\mathcal{F}_N$. Considering

$$f = a^T p, \quad U_N f = (U_N a)^T p,$$  

we can define a matrix $U_N \in \mathbb{R}^{N \times N}$ such that

$$U_N a = b.$$  

The matrix $U_N$ is a representation of the projected Koopman operator $U_N$. It also provides an approximate finite-dimensional linear description of the nonlinear system. This description is not obtained through global linearization techniques and is valid globally.

It follows from (11) and (12) that

$$U_N f = U_N (a^T p) = (U_N a)^T p$$  

and, since (13) holds for all $a$, we have

$$U_N p^T = p^T U_N.$$
where the operator $U_N$ acts on each component of the vector $p$. By considering each column separately, we obtain $P_N U^{T} p_j = U_N p_j = c_j^T p$, where $c_j$ is the $j$th column of $U_N$. This shows that each column of $U_N$ is related to the projection onto $\mathcal{F}_N$ of the image of a basis function $p_j$ through the Koopman operator $U^{T}$.

There are an infinity of possible projections $P_N$. We consider here a discrete orthogonal projection yielding the least squares fit at the points $x_k$, $k = 1, \ldots, K$, with $K \geq N$:

$$P_N g = \arg \min_{\hat{g} \in \text{span}\{p_1, \ldots, p_N\}} \sum_{k=1}^{K} |\hat{g}(x_k) - g(x_k)|^2. \quad (15)$$

This corresponds to the least squares solution

$$P_N g = p^{T} P_x^T \begin{pmatrix} g(x_1) \\ \vdots \\ g(x_K) \end{pmatrix}$$

where $P_x$ denotes the pseudoinverse of $P$. For $g = U^{T} p_j$, we obtain

$$P_N (U^{T} p_j) = p^{T} P_x \begin{pmatrix} U^{T} p_j(x_1) \\ \vdots \\ U^{T} p_j(x_K) \end{pmatrix} \approx p^{T} P_x \begin{pmatrix} p_j(y_1) \\ \vdots \\ p_j(y_K) \end{pmatrix}$$

where we used (9) evaluated at the states $x_k$ and assumed that measurement noise $||\cdot||$ is small. Equivalently, we have $U_N p^{T} \approx p^{T} P_x P_y$ so that (14) yields

$$\mathcal{U}_N \approx P_x^T P_y. \quad (16)$$

Inspired by (8), we finally compute

$$\mathcal{L}_{\text{data}} = \frac{1}{T_s} \log (P_x^T P_y), \quad (17)$$

where the function log denotes the (principal) matrix logarithm. The matrix $\mathcal{L}_{\text{data}}$ is an approximation of the matrix representation $\mathcal{L}_N$ of $L_N = P_N L_{\mathcal{F}_N}$, where $L_N f = p^{T} \mathcal{L}_N a$ for all $f = p^{T} a$. A rigorous justification is given in Section III-C.

**Remark 2.** Even with no measure noise, $\mathcal{L}_{\text{data}}$ is only an approximation of $\mathcal{L}_N$. Indeed, $\mathcal{L}_{\text{data}}$ is the matrix representation of the finite-rank operator $\frac{1}{T_s} \log (P_N U^{T})_{\mathcal{F}_N} = \frac{1}{T_s} \log (P_N e^{L_{\mathcal{F}_N}})_{\mathcal{F}_N} \neq P_N L_{\mathcal{F}_N}$. The two matrices $\mathcal{L}_{\text{data}}$ and $\mathcal{L}_N$ are identical only in the limit $N \rightarrow \infty$ and under some additional conditions related to the non-uniqueness of the matrix logarithm (see Section III-C).

**3) Third step - identification of the vector field:** We are now in position to identify the coefficients $w_k = (w_k^0, \ldots, w_k^n)$ of the vector field. To do so, we express the matrix representation of $L$ in terms of the coefficients $w_k^n$.

Then, we can find the coefficients by comparing the matrix representation with the matrix $\mathcal{L}_{\text{data}}$ obtained from data.

**Computation of a matrix representation of $L$:** It follows from (2) and (7) that

$$L = \sum_{j=1}^{n} \sum_{k=1}^{N_F} w_k^j L_k^j \quad (18)$$

with the linear operators

$$L_k^j = p_k \frac{\partial}{\partial x_j}, \quad j = 1, \ldots, n, \quad k = 1, \ldots, N_F. \quad (19)$$

Recalling that the vector field is a polynomial of total degree less or equal to $m_F$, we clearly see that the operators $L_k^j$ map polynomials of total degree less or equal to $m_0$ to polynomials of total degree less or equal to $m = m_0 + m_F - 1$. It follows that the operator $L_k^j_{|\mathcal{F}_N_0} : \mathcal{F}_N_0 \rightarrow \mathcal{F}_N$ is finite-rank (note that $P_N L_k^j_{|\mathcal{F}_N_0} = L_k^j_{|\mathcal{F}_N_0}$) and has a matrix representation $\mathcal{L}_k^j \in \mathbb{R}^{N \times N_0}$, with $N_0 = (m_0 + n)!/(m_0!n!) \leq N = (m + n)!/(m!n!)$. Denoting by $p_m^n$ and $p_m^n$ the vectors of monomials of total degree less or equal to $m_0$ and $m$, respectively, we have $L_k^j f = (\mathcal{L}_k^j a)^T p_m^n$ for all $f = a^T p_m^n$. It follows that $L_k^j p_m^n = (\mathcal{L}_k^j)^T p_m^n$, which implies that the $j$th column of $\mathcal{L}_k^j$ corresponds to the expansion of $L_k^j p_1$ in the basis of monomials $p_m^n$.

Next, we define an index function $\Psi(k) = (\psi_1(k), \ldots, \psi_n(k))$ that encodes the order of the monomials in the vector $p$, i.e. $p_k(x) = x_1^{\psi_1(k)} \cdots x_n^{\psi_n(k)}$. Then (19) implies that

$$L_k^j p_l = \psi_j(l)p_{\Psi^{-1}(\Psi(k)+\Psi(l))} - e_j,$$

where $e_j \in \mathbb{R}^n$ is the $j$th unit vector and the entries of $\mathcal{L}_k^j$ are given by

$$\mathcal{L}_k^j_{it} = \begin{cases} \psi_j(l) & \text{if } \Psi(i) = \Psi(k) + \Psi(l) - e_j, \\ 0 & \text{otherwise}. \end{cases} \quad (20)$$

Note that the matrices $\mathcal{L}_k^j$ can also be obtained by multiplying a multiplication matrix and a differentiation matrix (see [14] for more details). Finally, it follows from (18) that the matrix representation of $P_N L_{|\mathcal{F}_N_0} = L_{|\mathcal{F}_N_0}$ is given by

$$\mathcal{L} = \sum_{j=1}^{n} \sum_{k=1}^{N_F} w_k^j \mathcal{L}_k^j. \quad (21)$$

**Computation of the coefficients $w_k^n$:** In the previous section, we have derived a matrix representation $\mathcal{L}$ of $P_N L_{|\mathcal{F}_N_0}$. An approximation of this matrix representation is also given by the $N \times N_0$ matrix $[\mathcal{L}_{\text{data}}]_{N_0}$, constructed with the $N_0$ columns of (17) associated with monomials of total degree less or equal to $m_0$. Note that by disregarding the $N - N_0$ remaining columns in (17), we only consider monomials that are mapped by $L$ onto the span of basis monomials of total degree less or equal to $m$, for which the discrete orthogonal projection —i.e. the identity in this case— is exact (see also the discussion in Section IV-C). Then, it follows from (21) that the equality

$$[\mathcal{L}_{\text{data}}]_{N_0} \approx \sum_{j=1}^{n} \sum_{k=1}^{N_F} w_k^j \mathcal{L}_k^j. \quad (22)$$
yields a linear set of equations, whose solutions \( \hat{w}_j^k \) are the estimates of the \( nNF \) coefficients \( w_j^k \). If \( m = 1 \), we have \( N_0 = n + 1 \) and \( N = NF \). Moreover, \( \Psi_j(l) = 1 \) if \( \Psi(l) = e_j \) and \( \Psi_j(l) = 0 \) otherwise. Using (20) and (22), we obtain in this case\( \hat{w}_j^k = [\text{data}]_{kl} \) with \( l \) such that \( \Psi(l) = e_j \). Indeed, since \( L_N f = p^T\text{data}_N a \) for all \( f = p^T a \), it follows that

\[
P_N (F \cdot \nabla p_l) = p^T \text{data}_N e_l.
\] (23)

Since \( \Psi(l) = e_j \) (i.e. \( p_l \) is a monomial of degree 1), we also have

\[
P_N (F \cdot \nabla p_l) = P_N F_j = F_j,
\] (24)

so that \( F_j = p^T \text{data}_N e_l \approx p^T \text{data}_N e_l \), i.e. the \( l \)th column of \( \text{data}_N \) contains the estimates \( \hat{w}_j^k \).

If \( m > 1 \), the number of equations is greater than \( nNF \) and the set of equations is overdetermined. The estimates \( \hat{w}_j^k \) are obtained in this case by the least squares solution to the system

\[
\begin{pmatrix}
\hat{w}_1 \\
\vdots \\
\hat{w}_{NF}
\end{pmatrix} = \tilde{B}^t \text{vec} \left( \text{data}_N \right)_{N_0}
\] (25)

with

\[
\tilde{B} = \begin{pmatrix}
\text{vec}(\text{data}_1) & \cdots & \text{vec}(\text{data}_{N})
\end{pmatrix}
\]

and where \( \text{vec} \) stands for the vectorization of the matrix.

Some entries of \( \text{data}_N \) are zero for all \( j \) and \( k \), so that the corresponding entries of \( \tilde{B} \) do not depend on the values \( \hat{w}_j^k \) and the related equalities in (25) can be disregarded.

**Remark 3** (Nonlinear least squares problem). The identification problem could also be performed at the level of the Koopman semigroup. However solving the equality \( U_t = e^{T} \) (with a square matrix \( T \)) amounts to solving a (nonconvex) nonlinear least squares problem (as done in [29]). This might also be equivalent to solving the direct identification problem with an exact Taylor discretization of time-derivatives [31].

**Remark 4** (Estimation of the vector field). If needed, the method can directly provide the values \( F(x_k) \) of the vector field. Evaluating (23) and (24) at \( x_k \) for all \( k = 1, \ldots, K \) and considering \( L_N \approx \text{data}_N \), we obtain an approximation \( \hat{F}_j \) of the vector field given by

\[
\begin{pmatrix}
\hat{F}_j(x_1) \\
\vdots \\
\hat{F}_j(x_K)
\end{pmatrix} = P_X \left( \text{data}_N e_l \right)
\] (26)

with \( \Psi(l) = e_j \). This is quite similar to the approach developed with the dual method presented in Section V.

**B. Algorithm**

Our main lifting method for system identification is summarized in Algorithm 1.

---

**Algorithm 1 Main lifting method for nonlinear system identification**

**Input:** Snapshot pairs \( \{(x_k, y_k)\}_{k=1}^{K}, x_k \in \mathbb{R}^n \); sampling period \( T_s \); integers \( m_0 \geq 1 \) and \( m_F \geq 0 \).

**Output:** Estimates \( \hat{w}_j^k \).

1: \( m := m_0 + m_F - 1 \);

2: \( N_0 := (m_0 + n)!/(m_0 n! n!) \); \( N := (m + n)!/(m! n! n!) \); \( NF := (m F + n)!/(m! n! n!) \).

3: while \( N > K \) do

4: Increase \( K \) (add snapshot pairs) or decrease \( m_0 \) or \( m \).

5: end while

6: Construct the \( K \times N \) matrices \( P_X \) and \( P_Y \) defined in (10).

7: Compute the \( N \times N \) matrix \( \text{data}_N \) defined in (17).

8: if \( m_0 = 1 \) then

9: \( \hat{w}_j^k := [\text{data}]_{kl} \), with \( l \) such that \( \Psi_l = e_j \).

10: else

11: Construct the \( N \times N_0 \) matrices \( \tilde{B}_k \) using (20).

12: \( \hat{w}_j^k \) are given by (25).

13: end if

---

**C. Theoretical results**

In this section, we prove the convergence of Algorithm 1 in optimal conditions, i.e. with an infinite number of data points and basis functions, and an arbitrarily high sampling rate.

We consider the space \( F = L^2(X) \) (where \( \| \cdot \| \) is the \( L^2 \) norm) and the subspace \( \mathcal{F}_N \) spanned by the monomials \( \{p_k\}_{k=1}^{N} \). We first have the following lemma.

**Lemma 1.** Assume that the flow induced by (1) is invertible and nonsingular \(^1\), and that \( X \) is forward-invariant (i.e. \( \varphi^t(X) \subseteq X \) for all \( t > 0 \)) or backward-invariant \(^2\) (i.e. \( \varphi^{-t}(X) \subseteq X \) for all \( t > 0 \)). If

\[
\|P_X f - f\| \to 0
\] (27)

for all \( f \in L^2(X) \) as \( N \to \infty \), then the semigroup of Koopman operators \( U^t : L^2(X) \to L^2(X) \) and the infinitesimal generator \( L : \mathcal{D}(L) \to L^2(X) \) satisfy\( \|P_X U^t e^{P_N L} x \| \to 0 \) for all \( f \in L^2(X) \) as \( N \to \infty \).

**Proof.** We first check that the semigroup \( U^t \) is strongly continuous. For continuous functions \( g : X \to \mathbb{R} \), which are dense in \( L^2(X) \), we have \( \lim_{t \to 0} \|g - U^t g\| = 0 \). Moreover, we have

\[
\|U^t f\|^2 = \int_X |U^t f(x)|^2 dx = \int_{\varphi^t(x)} |f(x)|^2 |J_{\varphi^t}(x)| dx \leq \max_{x \in X} |J_{\varphi^t}(x)| \|f\|^2
\]

\(^1\)The flow is nonsingular if \( \mu(A) \neq 0 \) implies \( \mu(\varphi^t(A)) \neq 0 \) for all \( A \in \mathbb{R}^n \) and all \( t > 0 \), where \( \mu \) is the Lebesgue measure. This is a generic condition that is satisfied when the vector field \( F \) is Lipschitz continuous, for instance.

\(^2\)When \( X \) is backward-invariant, we assume that \( f(x) = 0 \) for all \( x \notin X \) and all \( f \in \mathcal{F} \), so that \( U^t \) is a well-defined semigroup.
or equivalently
\[
\frac{\|U'f\|^2}{\|f\|^2} \leq \max_{x \in X} |J_{\varphi'}(x)|^{-1}
\]
where \(|J_{\varphi'}(x)|\) is the determinant of the Jacobian matrix of \(\varphi'(x)\). Since the flow is nonsingular, \(|J_{\varphi'}(x)| \neq 0\) implies that \(U'\) is bounded. It follows that the semigroup of Koopman operators \(U^t\) is strongly continuous (see e.g. [32, Proposition 1.5.3(c)]).

Next, (27) implies that \(\|P_N L_N f - L f\| \to 0\) for all \(f \in \mathcal{F}_N\) as \(N \to \infty\) and one easily verifies that \(P_N L_N\) generates a strongly continuous semigroup of operators. Moreover, the set \(\mathcal{F}_N\) with \(N \to \infty\) is dense in \(D(L)\) for the graph norm \(\|f\|_L = \|f\| + \|Lf\|\) (since polynomial functions are dense in Sobolev spaces, see e.g. [33]). Then, it follows from the Trotter-Kato approximation theorem (see e.g. [32, Theorem 4.8]) that \(\|e^{P_N L_N t} f - U^t f\| \to 0\) for all \(f \in \mathcal{F}_N\) as \(N \to \infty\). Finally, using again (27), we obtain
\[
\|e^{P_N L_N t} f - P_N U^t f\| \leq \|e^{P_N L_N t} f - U^t f\| + \|U^t f - P_N U^t f\| \to 0
\]
for all \(f \in \mathcal{F}\) as \(N \to \infty\).

We are now in position to show that Algorithm 1 yields exact estimates \(\tilde{w}_k^{j}\) of the vector field coefficients in optimal conditions.

**Theorem 1.** Assume that the sample points \(x_k \in X\) are uniformly randomly distributed in a compact forward or backward invariant set \(X \subset [-1,1]^n\), and consider \(y_k = \varphi^T(x_k)\) (no measurement noise) where \(\varphi^T\) is invertible and nonsingular. If the Algorithm 1 is used with the data pairs \(\{x_k,y_k\}_{k=1}^{N}\), then \(\tilde{w}_k^{j} \to w_k^{j}\) for all \(j = 1,\ldots, n\) and \(k = 1,\ldots, N_{F}\) as \(N \to \infty\), \(K \to \infty\), and \(T_s \to 0\).

**Proof.** Recall that \(\overline{\mathcal{L}}_N\) and \(\overline{\mathcal{L}}\) are the matrix representations of \(L_N = P_N L\big|_{\mathcal{F}_N}\) and \(P_N L\big|_{\mathcal{F}_{N_0}}\), respectively. Using (21), we have
\[
\sum_{j=1}^{n} \sum_{k=1}^{N_{F}} (w_k^{j} - \tilde{w}_k^{j}) \overline{\mathcal{L}}_N = \overline{\mathcal{L}} - \sum_{j=1}^{n} \sum_{k=1}^{N_{F}} \tilde{w}_k^{j} \overline{\mathcal{L}}_N
\]
\[
= \left(\overline{\mathcal{L}} - \overline{\mathcal{L}}_{\text{data}}\right)_{N_0} + \left(\overline{\mathcal{L}}_{\text{data}}\right)_{N_0} - \sum_{j=1}^{n} \sum_{k=1}^{N_{F}} \tilde{w}_k^{j} \overline{\mathcal{L}}_N
\]
where \(\overline{\mathcal{L}}_{\text{data}}\) is constructed with \(N_0\) columns of \(\overline{\mathcal{L}}_{\text{data}}\). We first show that
\[
\overline{\mathcal{L}} - \overline{\mathcal{L}}_{\text{data}}\to 0
\]
as \(N \to \infty\). Since \(x_k \in X\), the discrete orthogonal projection (15) is a well-defined projection from \(L^2(X)\) to the subspace \(\mathcal{F}_N \subset L^2(X)\) spanned by the monomials \(\{p_k\}_{k=1}^{N}\). In addition, as \(K \to \infty\) (which is implied by \(N \to \infty\)), the discrete orthogonal projection converges in the strong operator topology to the orthogonal \(L^2\) projection (see e.g. [34] for a proof) and since the monomials form a complete basis of \(L^2(X)\), the orthogonal projection converges in the strong operator topology to the identity operator as \(N \to \infty\). This implies that (27) is satisfied. Supposing that \((P_N U^T_s - P_N e^{L_N T_s}) p_l = a^l_p\) with \(a \in \mathbb{R}^N\) and where \(p_l \in \mathcal{F}_N\) is a monomial, we can apply Lemma 1 and obtain \(\|P_N U^T_s - P_N e^{L_N T_s} p_l\| = \|a^l\| \to 0\) as \(N \to \infty\). Since the monomials form a complete basis of linearly independent functions, we have that \(a \to 0\). It is clear that \(a\) is the lth column of \(\overline{\mathcal{L}}_{N} - e^{L_N T_s}\), so that \(\overline{\mathcal{L}}_{N} - e^{L_N T_s} \to 0\), or equivalently \(\frac{1}{t} \log \overline{\mathcal{L}}_{N} - \frac{1}{t} \log e^{L_N T_s} \to 0\) as \(N \to \infty\). Since there is no measurement noise, (16), and (17) imply that \(\frac{1}{t} \log \overline{\mathcal{L}}_{N} = \overline{\mathcal{L}}_{\text{data}}\). In addition, the spectrum of \(\mathcal{L}_N\) is a subset of \(\mathbb{R} \times [-i \pi / T_s, i \pi / T_s] \to \mathbb{C}\) as \(T_s \to 0\), so that \(\frac{1}{t} \log e^{L_N T_s} = \overline{\mathcal{L}}_{N}\). Finally, this implies that \(\overline{\mathcal{L}}_{\text{data}} - \overline{\mathcal{L}}_{N} \to 0\) and since \([\mathcal{L}_N]_{N_0} = \mathcal{L}\), we obtain (29).

Now, we show that
\[
\overline{\mathcal{L}}_{\text{data}} = \sum_{j=1}^{n} \sum_{k=1}^{N_{F}} \hat{w}_k^{j} \overline{\mathcal{L}}_N \rightarrow 0.
\]
result to hold is that the functions \( \{ (L^j_1 h_1, \ldots, L^j_k h_{N_0}) \}_{j,k} \) should be linearly independent in \( \mathcal{F}^{N_0} \).

According to Theorem 1, Algorithm 1 identifies exactly the vector field, even if the data are collected in a small region of the state space. However, the result requires optimal conditions, i.e., an infinite number of data points \((K \to \infty)\) and an infinite sampling frequency \((T_s \to 0)\). The infinite number of data points is needed to represent the Koopman operator exactly with an infinite basis. Note that the requirement to collect the data points on an invariant set might not always be satisfied in practice. This is however a technical condition that ensures that the requirement to collect the data points on an invariant set might not always be satisfied in practice. This is however a technical condition that ensures that the requirement to collect the data points on an invariant set might not always be satisfied in practice.

### A. Systems with inputs

Consider an open dynamical system of the form

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

with \( x \in \mathbb{R}^n \) and with the input \( u \in U : \mathbb{R}^+ \to \mathbb{R}^p \). We define the associated flow \( \varphi : \mathbb{R}^+ \times \mathbb{R}^n \times \Omega \) so that \( t \mapsto \varphi(t, x, u(\cdot)) \) is a solution of (31) with the initial condition \( x \) and the input \( u(\cdot) \). Following the generalization proposed in [36], [37], we consider observables \( f : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R} \) and define the semigroup of Koopman operators

\[
U^t f(x, u) = f(\varphi^t(x, u(\cdot) = u), u)
\]

where \( u(\cdot) = u \) is a constant input. In this case, \( u \) can be considered as additional state variables and the above operator is the classic Koopman operator for the augmented system \( \dot{x} = F(x, u), u = 0 \). In particular, the infinitesimal generator is still given by (7).

It follows that the method proposed in Sections III-A1 and III-A2 can be used if

\[
\varphi^{T_s}(x, u(\cdot)) \approx \varphi^{T_s}(x, u(\cdot) = u(0))
\]

This condition holds when the input can be considered as constant between two snapshots (zero-order hold assumption), or equivalently if the sampling rate is high enough. The matrix \( \mathbf{U}_N \) is now obtained with snapshot pairs \( (\{x_k, u_k\}, \{y_k, u_k\}) \in \mathbb{R}^{(n+p) \times 2} \) and the rest of the procedure follows on similar lines with the augmented state space \( \mathbb{R}^{n+p} \). The efficiency of the identification method in the case of systems with inputs is shown in Section VI-B.

### B. Process noise

We have considered so far only measurement noise. We show that the proposed method is also robust to process noise. Consider a system described by the stochastic differential equation

\[
dx = F(x)dt + \sigma dw(t)
\]

where \( w(t) \) is the Wiener process. We define the flow \( \varphi : \mathbb{R}^+ \times \mathbb{R}^n \times \Omega \), where \( \Omega \) is the probability space, such that \( t \mapsto \varphi(t, x, \omega) \) is a solution to (32). In this case, the semigroup of Koopman operators is defined by (see e.g. [19])

\[
U^t f(x) = \mathbb{E}[f(\varphi(t, x, \omega))]
\]

and its infinitesimal generator is given by

\[
Lf = F \cdot \nabla f + \frac{\sigma_{\text{proc}}^2}{2} \Delta f
\]

where \( \Delta = \sum_j \partial^2 / \partial x_j^2 \) denotes the Laplacian operator that accounts for diffusion. The infinitesimal generator is related to the so-called Kolmogorov backward equation.

Now we can show that the numerical scheme of the proposed identification method does not need to be adapted to take process noise into account. As explained in [26], the first step of the method (Section III-A1) is still valid for identifying the matrix \( \mathbf{U} \). In the second step (Section III-A2), the procedure is the same, except that one has to consider the Laplacian operator whose representation in the basis of monomials is

\[
\mathbf{D} = \sum_{j=1}^n \mathbf{D}_j \in \mathbb{R}^{N \times N_0}
\]

with

\[
[\mathbf{D}_j]_{ij} = \begin{cases} 
\Psi_j(i)(\Psi_j(l) - 1) & \text{if } \Psi(i) = \Psi(l) - 2e_j \\
0 & \text{otherwise}
\end{cases}
\]

(33)

where \( \Psi \) is the monomial basis.

The equality (22) is then replaced by

\[
\sum_{j=1}^n \sum_{k=1}^{N_0} \hat{w}_k^j \mathbf{D}_k + \frac{\sigma^2}{2} \mathbf{D} = [\mathbf{L}_{\text{data}}]_{N_0}
\]

where \( \sigma \) is an additional unknown. While the operators \( L^j_k \) map monomials of total degree \( m \) to monomials of total

\[
\sum_{j=1}^n \sum_{k=1}^{N_0} \hat{w}_k^j \mathbf{D}_k + \frac{\sigma^2}{2} \mathbf{D} = [\mathbf{L}_{\text{data}}]_{N_0}
\]

where \( \sigma \) is an additional unknown. While the operators \( L^j_k \) map monomials of total degree \( m \) to monomials of total
degree greater or equal to \( m - 1 \), the Laplacian operator maps monomials of total degree \( m \) to monomials of total degree \( m - 2 \). Therefore all nonzero entries of \( \mathbf{D} \) correspond to zero entries of \( \mathbf{D}_K \), so that the addition of the diffusion term only modifies entries of \( \mathbf{D} \) which do not depend on \( \hat{w}_k \). In other words, the diffusion term does not affect the equalities on \( \hat{w}_k \), whose solution is still given by (25). In Section VI-B, an example illustrates the robustness of the method against process noise.

C. Non polynomial vector fields

The method can be adapted to identify non-polynomial vector fields of the form (2), where the library functions \( h_k \) are not monomials. In this case, it is clear that the infinitesimal generator \( L = F \cdot \nabla \), and in particular the operators \( L_k = h_k \partial / \partial x_j \), might not map the monomials \( p_k \) onto polynomials. However, we can still perform the second step of the algorithm by considering the operators \( P_N L_k |_{F_N} : F_N \rightarrow F_N \), where \( F_N \) is a subspace of polynomial functions. In this case, the projection \( P_N : F \rightarrow F_N \) must be effectively computed, which was not necessary when the vector field was polynomial. This projection can be the usual orthogonal projection or the discrete \( L^2 \) projection minimizing the least squares error at the data points.

However, using the projection \( P_N \) adds an additional error to the finite-dimensional approximation of the operator. Instead, we prefer to consider a representation of the operators \( L_k \) from a subspace \( F_{N_0} \) of polynomial functions of total degree less or equal to \( m_0 \) to an “augmented” subspace

\[
F_N = F_{N_0} \times \text{span} \left\{ \left\{ L_k p_k \right| p_k \in F_{N_0}, j = 1, \ldots, n, k = 1, \ldots, N_F \right\}. 
\]

We can therefore compute the matrix representation of the operators \( L_k |_{F_{N_0}} : F_{N_0} \rightarrow F_N \) without using the projection \( P_N \) onto \( F_N \), since \( P_N L_k |_{F_{N_0}} = L_k |_{F_{N_0}} \). In the case \( F_{N_0} = \text{span} \left\{ \{ x_1, \ldots, x_n \} \right\} \) (i.e. \( m_0 = 1 \)), we simply get \( F_N = F_{N_0} \times \text{span} \left\{ \{ h_k, k = 1, \ldots, N_F \} \right\} \).

We finally note that a more straightforward method is to perform a least squares regression on the values of the vector field at the sample points, values which can be obtained according to Remark 4. However, numerical experiments suggest that this method is less efficient than the above-mentioned method.

V. A DUAL LIFTING METHOD FOR LARGE SYSTEMS

A major limitation of the main method presented in Section III (Algorithm 1) is that it might require a large number of data points. Indeed, the number of data points must be larger than the number of basis functions (\( K \geq N \)) to ensure that the discrete orthogonal projection (15) is well-defined. In the case of high-dimensional systems in particular, the number of basis functions is huge and is likely to exceed the number of available data points. Moreover, the algorithm might also be computationally intractable (e.g. computation of the matrix logarithm in (17)). In this section, we circumvent the above limitations by proposing a dual approach, which is developed in a \( K \)-dimensional “sample space” instead of the \( N \)-dimensional functional space. This method can be used when the number of basis functions is larger than the number of data points, i.e. \( N \geq K \).

A. Description of the method

Similarly to the main lifting method, the dual method consists of three steps: lifting of the data, identification of the Koopman operator, and identification of the vector field. In the last step, the algorithm provides the value of the vector field at each data point, so that the dual method can be seen as an indirect method for time derivatives estimation. This is similar in essence to the vector field estimation detailed in Remark 4. The identification is achieved in a distributed way, a feature which makes the algorithm computationally more tractable in the case of high-dimensional systems and well-suited to parallel computing.

1) First step - lifting of the data: This step is similar to the first step of the main method (Section III-A1). But in this case, choosing the basis functions equal to the library functions of the vector field is not more convenient for the next steps. We will therefore consider monomials \( g_k = p_k \), but also Gaussian radial basis functions \( g_k(x) = e^{-\gamma \|x-x_k\|^2} \) with \( k = 1, \ldots, K \) and where \( \gamma > 0 \) is a parameter. We construct the data \( K \times N \) matrices

\[
P_x = \begin{pmatrix} g(x_1)^T \\ \vdots \\ g(x_K)^T \end{pmatrix}, \quad P_y = \begin{pmatrix} g(y_1)^T \\ \vdots \\ g(y_K)^T \end{pmatrix}
\]

(33)

where \( g \) is the vector of basis functions \( g_k \). When using Gaussian radial basis functions, the number of basis functions is equal to the number of samples (i.e. \( N = K \)) and therefore does not depend on the dimension \( n \). This is particularly useful in the case of high-dimensional systems, where the matrices (33) should be of reasonable size. In Section VI, we will only use Gaussian radial basis functions.

2) Second step - identification of the Koopman operator: We use a dual matrix representation of the Koopman operator, which is inspired (but slightly different) from a kernel-based approach developed in [27].

In the main method, we constructed the \( N \times N \) matrix \( \mathbf{U}_N \approx \mathbf{P}_x \mathbf{P}_x^\top \) which represents the operator \( U_N \). Instead, we can consider the \( K \times K \) matrix representation

\[
\tilde{U}_K \approx P_y \mathbf{P}_x^\top = P_x \mathbf{U}_N \mathbf{P}_x^\top, \quad (34)
\]

a construction which is similar to the original formulation of the Dynamic Mode Decomposition (DMD) algorithm\(^3\) [21]. The matrix \( \mathbf{P}_x \) can be interpreted as a change of coor-
radial basis functions is equivalent to constructing the matrices \( P \) of inner products can be approximated by a Gaussian kernel. Following similar lines as in [27], we note that \( \tilde{U}_K \) is given by

\[
\tilde{f}(x) = \sum_{j=1}^{N_{p}} w_j^T \phi_j(x),
\]

where \( w_j \) are the coefficients of the linear combination of the values \( f(x_1), \ldots, f(x_K) \) that is equal to \( U^T f(x) \). The row \( r_i \) is obtained by considering the \( N \) “test” functions \( f_j \).

In this case, each of the \( N \) functions yields a constraint and there are \( K \) unknowns, so that \( K \leq N \) is required.

**Remark 6.** Following similar lines as in [27], we note that we have

\[
\tilde{U}_K \approx P_x P_x^T = P_y P_x^T (P_x P_x^T)^+ \triangleq A G^T
\]

where the entries of \( A \) and \( G \) can be interpreted as the inner products

\[
[A]_{ij} = p(x_j)^T \phi_i(y_i), \quad [G]_{ij} = p(x_j)^T \phi_i(x_i).
\]

(Here, we consider without loss of generality that the matrices \( P_x \) and \( P_y \) are constructed with monomials.) The inner products can be approximated by a Gaussian kernel \( g(x_i, x_j) = g_j(x_i) \), so that

\[
[A]_{ij} = [A]_{ij} = g(x_i, x_j), \quad [G]_{ij} = [A]_{ij} = g(x_j, x_j).
\]

In this context, constructing \( P_x \) and \( P_y \) with Gaussian radial basis functions is equivalent to constructing the inner-product matrices \( A \) and \( G \).

Finally, similarly to (17), we compute the \( K \times K \) matrix

\[
\tilde{L}_{\text{data}} = \frac{1}{L_2} \log(P_y P_x^T).
\]

3) **Third step - identification of the vector field:** Using a similar idea as the one explained in Remark 4, we can directly identify the vector field at the different values \( x_k \) and the coefficients \( w_k \) are then obtained by solving \( N \) separate regression problems.

**Computation of the vector field \( \tilde{F}(x_k) \):** We assume that \( \tilde{L}_{\text{data}} \) is an approximation of the matrix representation of \( L \) in the sample space and we have

\[
\begin{pmatrix}
\tilde{F}(x_1) \cdot \nabla f(x_1) \\
\vdots \\
\tilde{F}(x_K) \cdot \nabla f(x_K)
\end{pmatrix} \approx \tilde{L}_{\text{data}}
\]

Considering the above equality with the identity function \( f(x) = x \), we obtain an approximation \( \tilde{F} \) of the vector field that is given by

\[
\tilde{F}(x_k)^T \approx \tilde{L}_{\text{data}} (x_k^T)
\]

The choice of the functions \( f \) used to obtain (37) is arbitrary. However, considering monomials of degree one is natural and choosing more functions would yield an over-constrained problem which does not necessarily improve the accuracy of the result. Note also that an approach more similar to the main method is to compute an approximation of the matrix representation of \( L = F \cdot V \) in the sampling space and compare it with \( L_{\text{data}} \). However, this does not yield better results.

**Computation of the coefficients \( \tilde{w}_k \):** When the value of the vector field is known at every data points, we can find an estimation \( \tilde{w}_k \) of the coefficients \( w_k \) by solving a regression problem. This problem is decoupled: for each \( j = 1, \ldots, n \), we have to solve

\[
\tilde{F}_j(x_k) = \tilde{w}_j^T h_j(x_k) \quad k = 1, \ldots, K,
\]

which takes the form

\[
\begin{pmatrix}
\tilde{F}_j(x_1) \\
\vdots \\
\tilde{F}_j(x_K)
\end{pmatrix} = H_x
\begin{pmatrix}
\tilde{w}_1 \\
\vdots \\
\tilde{w}_N_x
\end{pmatrix}
\]

with

\[
H_x =
\begin{pmatrix}
h(x_1)^T \\
\vdots \\
h(x_K)^T
\end{pmatrix}
\]

and where \( h \) is the vector of library functions \( h_k \) of the vector field. We do not make any assumption on these library functions, which are not necessarily monomials.

Since we can reasonably assume that most coefficients are zero, we can promote sparsity of the vector of coefficients \( \tilde{w}_k \) by adding a penalty term, which yields the Lasso optimization problem [38]

\[
\min_{w \in \mathbb{R}^{N_x}} \left\| H_x w - \begin{pmatrix}
\tilde{F}_j(x_1) \\
\vdots \\
\tilde{F}_j(x_K)
\end{pmatrix} \right\|_2^2 + \rho \|w\|_1
\]

where \( \rho \) is a positive regularization parameter. Other techniques could also be used to infer \( w \) from the values of the vector field (see e.g. [8], [7]). More generally, machine learning techniques could also be used to solve the regression problem (38).

**B. Algorithm**

The dual method is summarized in Algorithm 2.

**C. Theoretical results**

We now show the convergence of Algorithm 2 in optimal conditions. Let \( X \subset \mathbb{R}^n \) be a compact set that contains all the data points \( x_k \) and assume that \( F = L^2(X) \) (where \( \| \cdot \| \) is the \( L^2 \) norm). We consider a partition \( \{X_k\}_{k=1}^K \) of \( X \) such that \( x_k \in X_k \) and \( x_j \notin X_k \) for all \( j \neq k \) and such that the Lebesgue measure \( \mu(X_k) = \mu(X)/K \). Let also \( \tilde{F}_K \) be the set of functions that
Algorithm 2 Dual lifting method for nonlinear system identification

**Input:** Snapshot pairs \( \{(x_k, y_k)\}_{k=1}^{N} \), \( x_k \in \mathbb{R}^n \); basis functions \( \{g_k\}_{k=1}^{N} \) (with \( N \geq K \)); library functions \( \{h_k\}_{k=1}^{N} \).

**Output:** Estimates \( \hat{f}(x_k) \) and \( \hat{w}_k \).

1. Construct the \( K \times N \) matrices \( \mathbf{F}_x \) and \( \mathbf{P}_x \) defined in (33)
2. Compute the \( K \times K \) matrix \( \mathbf{L}_{\text{data}} \) defined in (36)
3. Obtain \( \hat{F}(x_k) \) with (37)
4. Construct the \( K \times N_\mathcal{F} \) matrices \( \mathbf{H}_x \) defined in (39)
5. For each \( j \), solve the regression problem (38), e.g. solve the Lasso problem (40), to obtain \( \hat{w}_k \).

are piecewise constant on this partition. Then, the matrix \( \mathbf{U}_K \) can be interpreted as the matrix representation of the finite-rank operator \( \tilde{U}_K : \mathcal{F} \to \mathcal{F} \) that minimizes

\[
\sum_{i=1}^{N} \left\| \left( \tilde{P}_K U^T \mathbf{K} - \tilde{U}_K \tilde{P}_K \right) g_i \right\|^2 = \frac{\mu(X)}{K} N \sum_{i=1}^{N} \left\| \begin{pmatrix} g_i(y_1) \\ \vdots \\ g_i(y_K) \end{pmatrix} - \tilde{U}_K \begin{pmatrix} g_i(x_1) \\ \vdots \\ g_i(x_K) \end{pmatrix} \right\|^2 \quad (41)
\]

where \( \tilde{P}_K : \mathcal{F} \to \mathcal{F} \) is the projection operator such that \( \tilde{P}_K f(x) = f(x_k) \) for all \( x \in X_k \). Denoting by \( \mathcal{F}_K \) the span of \( \{g_i\}_{i=1}^{K} \), we will also use the projection \( \Pi_K : \mathcal{F} \to \mathcal{F}_K \) obtained through interpolation with collocation points \( x_k \), i.e. \( \Pi_K f(x_k) = f(x_k) \) for all \( k = 1, \ldots, K \). It is easy to see that, for all \( f \in \mathcal{F} \),

\[
\tilde{P}_K \Pi_K f = \tilde{P}_K f \quad \Pi_K \tilde{P}_K f = \Pi_K f. \quad (42)
\]

We first prove the following lemma.

**Lemma 2.** Consider the semigroup of operators \( U^t : L^2(X) \to L^2(X) \) and its infinitesimal generator \( L : \mathcal{D}(L) \to L^2(X) \). Suppose that \( \{g_i\}_{i=1}^{\infty} \), with \( g_i \in C^1(X) \), is dense in \( \mathcal{D}(L) \) for the graph norm \( \|f\|_L = \|f\| + \|Lf\| \). If there exists \( L_K : \mathcal{F}_K \to \mathcal{F}_K \) that satisfies

\[
\left( \tilde{P}_K L - \tilde{L}_K \tilde{P}_K \right) g_i = 0 \quad (43)
\]

for all \( g_i \in \mathcal{F} \) with \( l \leq K \), then \( \left\| \left( \tilde{P}_K U^t - e^{-\tilde{L}_K t} \tilde{P}_K \right) g_i \right\| \to 0 \) as \( K \to \infty \).

**Proof.** We have

\[
\left\| \left( L - \tilde{L}_K \tilde{P}_K \right) g_i \right\| \leq \left\| L - \tilde{L}_K \right\|_L g_i + \left\| \left( \tilde{P}_K L - \tilde{L}_K \tilde{P}_K \right) g_i \right\| \quad (44)
\]

where we used (43) and the fact that piecewise constant functions are dense in \( L^2(X) \). Moreover, it follows from (42) that

\[
\tilde{L}_K \tilde{P}_K f = \tilde{L}_K \tilde{P}_K \Pi_K f = \tilde{P}_K L \Pi_K f = \tilde{P}_K \Pi_K L \Pi_K \tilde{P}_K f,
\]

so that \( \tilde{L}_K \tilde{P}_K \) generates a strongly continuous (finite-rank) semigroup of operators \( e^{\tilde{L}_K t} \mathbf{K} : \mathcal{F} \to \mathcal{F} \). Using (44) and applying Trotter-Kato approximation theorem (see e.g. [32, Theorem 4.8]), we obtain

\[
\left\| \left( U^t - e^{-\tilde{L}_K t} \tilde{P}_K \right) g_i \right\| = \left\| \left( U^t - e^{-\tilde{L}_K t} \tilde{P}_K \right) g_i \right\| \to 0
\]
as \( K \to \infty \), and finally

\[
\left\| \left( \tilde{P}_K U^t - e^{-\tilde{L}_K t} \tilde{P}_K \right) g_i \right\| \leq \left\| \left( \tilde{P}_K U^t - U^t \right) \right\| + \left\| \left( U^t - e^{-\tilde{L}_K t} \tilde{P}_K \right) g_i \right\| \to 0.
\]

The following results show that Algorithm 2 provides exact estimates \( \hat{F}(x_k) \) of the vector field in optimal conditions.

**Theorem 2.** Assume that the sample points \( x_k \in X \) are uniformly randomly distributed in a compact forward invariant set \( \mathcal{X} \), and consider \( y_k = \varphi_k(x_k) \) (no measurement noise). Suppose that \( \{g_i\}_{i=1}^{\infty} \), with \( g_i \in C^1(X) \), form a linearly independent basis of \( \mathcal{F} \) and that its span is dense in \( \mathcal{D}(L) \) for the graph norm \( \|f\|_L = \|f\| + \|L f\| \). If \( \sum_{i} \|\nabla g_i(x_k)\|_{L^\infty} < \infty \) (case 2), then

\[
\frac{1}{K} \sum_{k=1}^{K} \left\| \tilde{F}(x_k) - \hat{F}(x_k) \right\|^2 \to 0
\]
as \( K \to \infty \), \( N \to \infty \), and \( T_s \to 0 \).

**Proof.** In the following, we denote by \( \tilde{f}(x_k) \) the vectors

\[
\tilde{f}(x_k) \triangleq \begin{pmatrix} f(x_1) \\ \vdots \\ f(x_K) \end{pmatrix}
\]

If \( N = K \) (case 1), the minimization of (41) is not overconstrained and \( \tilde{U}_K \) satisfies

\[
\tilde{g}_i(x_k) - \tilde{U}_K \tilde{g}_i(x_k) = 0 \quad (45)
\]

exactly for \( l \). In the other situation (case 2), we denote \( x_k^* = \arg \min x_k - y_k \) and it follows from (41) that

\[
\sum_{i=1}^{N} \left\| \tilde{g}_i(x_k) - \tilde{U}_K \tilde{g}_i(x_k) \right\|^2 \leq \min_{u_k \in \mathbb{R}^N} \frac{1}{K} \sum_{k=1}^{K} \left\| \tilde{g}_i(x_k) - u_k \tilde{g}_i(x_k) \right\|^2
\]

\[
\leq \frac{1}{K} \sum_{i=1}^{N} \left\| \tilde{g}_i(x_k) - \tilde{g}_i(x_k) \right\|^2
\]

since \( \sum_{i=1}^{N} \|\nabla g_i(x_k)\|_{L^\infty} < \infty \) is bounded (case 2) and \( \|x_k^* - y_k\| \) is arbitrarily small as \( K \to \infty \). (\( X \) is forward invariant). This implies that

\[
\frac{1}{K} \left\| \tilde{g}_i(x_k) - \tilde{U}_K \tilde{g}_i(x_k) \right\| \to 0 \quad (46)
\]
as $K \to \infty$.

Moreover, since $\{g_l\}_{l=1}^N$ is a set of linearly independent functions, so is the set $\{\tilde{P}_K g_l\}_{l=1}^N$ in $\mathcal{F}_K$ for $K$ large enough. It follows that the operator $\tilde{L}_K$ satisfying (43) exists. Then, Lemma 2 implies that

$$\frac{1}{K} \| \tilde{g}_l(y_k) - e^\tilde{L}_K T_k \tilde{g}_l(x_k) \| \to 0$$

where $\tilde{L}_K$ is the matrix representation of $\tilde{L}_K$. Combining with (45) or (46), we obtain

$$\frac{1}{K} \left\| \left( \tilde{U}_K - e^\tilde{L}_K T_k \right) \tilde{g}_l(x_k) \right\| \leq \frac{1}{K} \left\| \tilde{U}_K \tilde{g}_l(x_k) - \tilde{g}_l(y_k) \right\| + \frac{1}{K} \left\| \tilde{g}_l(y_k) - e^\tilde{L}_K T_k \tilde{g}_l(x_k) \right\| \to 0.$$ 

Since the vectors $\tilde{g}_l(x_k)$ form a linearly independent basis in $\mathbb{R}^K$ as $K \to \infty$, we have $\tilde{U}_K - e^\tilde{L}_K T_k \to 0$ or equivalently $\frac{1}{T_k} \log \tilde{U}_K - \frac{1}{T_k} \log e^\tilde{L}_K T_k \to 0$. Since there is no measurement noise, (34) and (36) imply that $\tilde{L}_\text{data} = \frac{1}{T_k} \log \tilde{U}_K$ and, in addition, $\frac{1}{T_k} \log e^\tilde{L}_K T_k = \tilde{L}_K$ as $T_k \to 0$. Hence, we obtain $\tilde{L}_\text{data} = \tilde{L}_K$. Denoting by $x^{(j)}$ the jth component of $x$, we have

$$\frac{1}{K} \sum_{k=1}^K \left( F_j(x_k) - \tilde{F}_j(x_k) \right)^2 = \frac{1}{K} \left\| \begin{pmatrix} F_j(x_1) \\ \vdots \\ F_j(x_K) \end{pmatrix} - \tilde{L}_\text{data} \begin{pmatrix} x^{(1)}_1 \\ \vdots \\ x^{(1)}_K \end{pmatrix} \right\|^2$$

or equivalently

$$\frac{\mu(X)}{K} \sum_{k=1}^K \left( F_j(x_k) - \tilde{F}_j(x_k) \right)^2 = \left\| \tilde{P}_K Lx^{(j)} - \tilde{L}_K \tilde{P}_K x^{(j)} \right\|^2$$

where $\mu(X)$ denotes the Lebesgue measure of $X$. Finally, we have

$$\left\| \left( \tilde{P}_K L - \tilde{L}_K \tilde{P}_K \right) x^{(j)} \right\| \leq \left\| \tilde{P}_K L \left( x^{(j)} - \Pi_K x^{(j)} \right) \right\| + \left\| \tilde{L}_K \tilde{P}_K \left( x^{(j)} - \Pi_K x^{(j)} \right) \right\| + \left\| \left( \tilde{P}_K L - \tilde{L}_K \tilde{P}_K \right) \Pi_K x^{(j)} \right\| \to 0$$

where we used the fact that $\| \Pi_K f - f \|_L \to 0$ (see (42), (43). This concludes the proof. \hfill $\Box$

Theorem 2 proves the convergence of Algorithm 2 when it is used with Gaussian radial basis functions. This basis is dense with respect to the Sobolev norm in $\mathcal{D}(L)$ and so that $\| \Pi_K f - f \|_L \to 0$ (see [39], [40]). Moreover, it is linearly independent and satisfies the additional technical conditions (case 1). Polynomial functions are dense with respect to the Sobolev norm, but the condition $\| \Pi_K f - f \|_L \to 0$ might not be satisfied for general polynomials and without additional conditions on the sampling (collocation) points $x_k$. This motivates the use of Gaussian radial basis functions.

Table I summarizes the main differences between the two frameworks (main and dual methods).

VI. ILLUSTRATIVE EXAMPLES

The goal of this section is to provide several examples to illustrate the two methods, including some extensions of the main method. We do not provide here an extensive study of the performance with respect to the choice of basis functions and parameters, considering that this is out of the scope of the present paper.

We consider simulated data and, unless otherwise stated, we add a Gaussian state-dependent measurement noise with zero mean and standard deviation $\sigma_{\text{meas}} = 0.01$ (see (5)).

A. Main method

We use the lifting method described in Section III, with the parameters $m_0 = 1$ and $m_p = 3$ ($m_1 = m_0 + m_p - 1 = 3$). We consider three systems that exhibit different types of behaviors.

1) Van der Pol oscillator: the dynamics are given by

$$\dot{x}_1 = x_2,$$

$$\dot{x}_2 = (1 - x_1^2)x_2 - x_2$$

and possess a stable limit cycle.

2) Unstable equilibrium: the dynamics are given by

$$\dot{x}_1 = 3x_1 + 0.5x_2 - x_1x_2 + x_1^2 + 2x_1^3,$$

$$\dot{x}_2 = 0.5x_1 + x_2$$

and are characterized by an unstable equilibrium at the origin.

3) Chaotic Lorenz system: the dynamics are given by

$$\dot{x}_1 = 10(x_2 - x_1),$$

$$\dot{x}_2 = x_1(28 - x_3) - x_2,$$

$$\dot{x}_3 = x_1x_2 - 8/3x_3$$

and exhibit a chaotic behavior.

A set of $K$ data pairs is generated by taking snapshots at times $\{0, T_s, \ldots, K/rT_s\}$ from $r$ trajectories of these systems. For the first two systems, we consider a setting that is not well-suited to a direct estimation of the derivatives: the sampling period $T_s$ is (reasonably) large and only two or three data points are taken on each trajectory. The identification of the third system, however, requires a smaller sampling period and a larger number of samples. Parameters used to generate the datasets are summarized in the left part of Table II.

For each model, Algorithm 1 yields the estimates $\hat{w}_k^{j}$ of the coefficients $w_k^{j}$. We compute the root mean square error

$$\text{RMSE} = \sqrt{\frac{1}{nN_F} \sum_{j=1}^n \sum_{k=1}^{N_F} (w_k^{j} - \hat{w}_k^{j})^2}$$
and the normalized root mean square error \( \text{NRMSE} = \frac{\text{RMSE}}{\bar{w}} \), where \( \bar{w} \) is the average value of the nonzero coefficients \( |w_k^i| \). The RMSE and NRMSE values averaged over 50 experiments are small (Table II) and show that the lifting method achieves good performance to identify each system with a fairly low number of samples.

For the three systems described above, we also consider the effect of the sampling period \( T_s \) on the performance of the method (Figure 2). In the noiseless case, the NRMSE decreases (exponentially) as the sampling period decreases. This is in agreement with the fact that the NRMSE tends to zero as \( T_s \to 0 \) (Theorem 1). With measurement noise, this is not the case since the method is biased. In this case, small values of the sampling period make the method more sensitive to noise, so that the minimal (nonzero) value of the NRMSE is obtained with an intermediate value of the sampling period.

Next, the approximation of the vector field obtained with (26) is compared with the approximation obtained directly from data through (central) finite differences, i.e. 
\[
\hat{F}(x_k) = \frac{x_{k+1} - x_{k-1}}{2T_s}.
\]
We consider the three systems and compute the normalized root mean square error on the vector field 
\[
\text{NRMSE}_F = \sqrt{\frac{1}{K-1} \sum_{j=1}^{n} \sum_{k=2}^{K-1} \left\| \hat{F}(x_k) - F(x_k) \right\|^2 / \left( \frac{1}{K-1} \sum_{k=2}^{K} \left\| F(x_k) \right\| \right)}
\]
averaged over 10 experiments, for different values of the sampling period. The results are shown in Figure 3. For each system, we observe that the approximation obtained with the lifting method provides an estimate with an acceptable error (e.g. \( \text{NRMSE}_F < 0.1 \)) for larger values of the sampling period than the direct finite difference method. This approximation is also characterized by a clear transition at a critical value of the sampling period, above which the NRMSE sharply increases (not observed with the unstable system, for which the critical value is beyond the maximal integration time). These results demonstrate the need of considering an indirect method to estimate the vector field (and therefore identify the system) when the sampling period is large.

### B. Extensions

We now illustrate several extensions of the lifting method mentioned in Section IV: systems with inputs, process noise, and non-polynomial vector fields.

1) **Input and process noise:** We consider the forced Duffing system 
\[
\dot{x}_1 = x_2 \quad \dot{x}_2 = x_1 - x_1^3 - 0.2 x_2 + 0.2 x_1^2 \cos(t)
\]
and generate \( K = 250 \) snapshot data pairs from 5 trajectories (50 each), with initial conditions \([-1, 1]^2\]. The lifting method provides a good estimation of the vector field (including the forcing term 0.2 \( x_1^2 \cos(t) \)). The RMSE (see Equation (47)) and NRMSE computed over all coefficients (including those related to the forcing term) are given in Table III for different values of the sampling period. Note that we use again the parameters \( m_n = 1 \) and \( m_F = 3 \).

Now, we replace the forcing term in (49) by the white noise \( \eta(t) \) with different values of the standard deviation \( \sigma_{\text{proc}} \) (note that we still add measurement noise with \( \sigma_{\text{meas}} = 0.01 \)). We generate \( K = 500 \) snapshot data pairs from 10 trajectories computed with the Euler-Maruyama scheme, with initial conditions \([-1, 1]^2\). The sampling period is equal to \( T_s = 0.2 \). As shown in Table IV, the error is small even with strong process noise, suggesting that the method is robust against process noise.
consider a genetic toggle switch (see e.g. [41])

period is $T_m$ functions are the 5 monomials of total degree 0 and 1 (i.e. we use the extension presented in Section IV-C. The basis and we generate trajectories, with initial conditions on $[0, s]$. However, the results are inferred correctly and we obtain a NRMSE equal to 0.008 (averaged over 50 experiments). However, the results are sensitive to noise in this case. With a measurement noise with $\sigma_{\text{meas}} = 0.001$, the NRMSE increases to 0.494. As shown below, the dual method is more robust to noise in this case.

2) Non polynomial vector fields: In this example, we consider a genetic toggle switch (see e.g. [41])

$$
\dot{x}_1 = -x_1 + 2x_2 \\
\dot{x}_2 = -\frac{x_2}{1 + x_3^2} \\
\dot{x}_3 = -2x_3 + x_4 \\
\dot{x}_4 = -2x_4 + \frac{1}{1 + x_1^2}
$$

and we generate $K = 50$ snapshot data pairs from 50 trajectories, with initial conditions on $[0, 1]^4$. The sampling period is $T_s = 0.1$. Since the vector field is not polynomial, we use the extension presented in Section IV-C. The basis functions are the 5 monomials of total degree 0 and 1 (i.e. $m_0 = 1$ and $m_p = 1$), to which we add 12 Hill functions

$$
k = \{1, 2, 3, 4\}, \quad l = \{1, 2, 3\}.
$$

When there is no measurement noise, all coefficients (including those related to non-polynomial terms) are inferred correctly and we obtain a NRMSE equal to 0.008 (averaged over 50 experiments). However, the results are sensitive to noise in this case. With a measurement noise with $\sigma_{\text{meas}} = 0.001$, the NRMSE increases to 0.494. As shown below, the dual method is more robust to noise in this case.

C. Dual method

We illustrate the dual method in the case of a non-polynomial vector field. The main interest of the method, however, is its use with high-dimensional datasets, where the number of basis functions $N$ is (much) larger than the number of sample points $K$. This will be illustrated in the next section.

The dual method requires to solve a regression problem. When $K < N_F$, we solve the (underconstrained) Lasso problem (40) with the MATLAB toolbox “yall1” [42], [43] (L1-L2 problem, with the parameter $\rho = 0.01$). When $K \geq N_F$, we solve the (overconstrained) problem (40) with the MATLAB function “lasso” (with the parameter $\lambda = 1/K$). Note that the value of the regularization parameter might not be optimal in all cases, and we did not extensively study its effect on the performance of the algorithm. In the following, we only use Gaussian radial basis functions with $\gamma = 0.1$ or $\gamma = 0.01$. Numerical simulations performed with monomial bases (not shown here) yield similar results for small dimensions, but are less accurate and more computationally expensive for large dimensions.

We consider the toggle switch system introduced in Section VI-B. Sample points are generated in the same conditions (i.e. $K = 50$, $T_s = 0.1$). We consider Gaussian radial basis functions with $\gamma = 0.1$ and 17 library functions (5 monomials of total degree 0 and 1, and 12 Hill functions (50)). With no noise, the NRMSE (averaged over 50 experiments) is equal to 0.064, which is worse than with the main method (Section VI-B). However, we obtain a NRMSE equal 0.117 with $\sigma_{\text{meas}} = 0.001$ and equal to 0.637 with $\sigma_{\text{meas}} = 0.01$. This shows that, in this case, the dual method is more robust to measurement noise than the main method.

D. Application to network identification

In the context of dynamical systems, each state can be seen as the node of a network. Moreover, a link can be drawn from node $i$ to node $j$ if the dynamics of the state $x_j$ depends on the state $x_i$. Under the assumption that the vector field is of the form (2), there is a link from node $i$ to node $j$ if there is at least one nonzero coefficient $w_{ij}$ such that the corresponding library function $h_k$ depends on $x_i$.

Network reconstruction aims at predicting links between states from data, a goal which is equivalent to finding nonzero coefficients $w_{ij}$ in our setting. We will consider that estimated coefficients $\hat{w}_{ij}$ with a small absolute value

<table>
<thead>
<tr>
<th>Table II</th>
</tr>
</thead>
<tbody>
<tr>
<td>Features of the datasets and (normalized) root mean square error averaged over 50 simulations.</td>
</tr>
<tr>
<td>Sampling period ($T_s$)</td>
</tr>
<tr>
<td>-----------</td>
</tr>
<tr>
<td>1. Van der Pol</td>
</tr>
<tr>
<td>2. Unstable</td>
</tr>
<tr>
<td>3. Lorenz</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table III</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Normalized) root mean square error (averaged over 50 experiments) related to the identification of the forced Duffing system.</td>
</tr>
<tr>
<td>Sampling period ($T_s$)</td>
</tr>
<tr>
<td>-----------</td>
</tr>
<tr>
<td>0.2</td>
</tr>
<tr>
<td>0.4</td>
</tr>
<tr>
<td>0.6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table IV</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Normalized) root mean square error (averaged over 10 experiments) related to the identification of the Duffing system with process noise.</td>
</tr>
<tr>
<td>Noise strength ($\sigma_{\text{proc}}$)</td>
</tr>
<tr>
<td>-----------</td>
</tr>
<tr>
<td>0.2</td>
</tr>
<tr>
<td>0.4</td>
</tr>
<tr>
<td>0.6</td>
</tr>
<tr>
<td>0.8</td>
</tr>
<tr>
<td>1.0</td>
</tr>
</tbody>
</table>
are mainly due to measurement noise and have an exact value \( w_k \) equal to zero. Hence, we decide that a link is present in the network only if the related value \(|w_k|\) is above a given threshold. To evaluate the performance of the method, one can compute the true positive rate (i.e. number of correctly identified links divided by the actual number of links) and the false positive rate (i.e. number of incorrectly identified links divided by the actual number of missing links). Varying the threshold value, we can plot the true positive rate against the false positive rate, which corresponds to the receiver operating characteristic (ROC) curve. If the area under the ROC curve (AUROC) is close to one, the network inference method provides good results (bad result correspond to a value close to 0.5).

1) Kuramoto oscillators: We consider a network of \( n \) Kuramoto phase oscillators

\[
\dot{\theta}_i = \omega_i + C \sum_{j=1}^{n} a_{ij} \sin(\theta_j - \theta_i) \quad i = 1, \ldots, n
\]

with \( \theta_i \in [0, 2\pi) \). The coupling strength is set to \( C = 10 \) and the natural frequencies \( \omega_i \) are uniformly randomly distributed on \([0, 0.1]\). The values \( a_{ij} \) are the entries of the weighted adjacency matrix of a random Erdős-Rényi graph (with a probability \( p_{\text{link}} = 0.3 \) for any two nodes to be connected). The link weights are uniformly randomly distributed on \([0, 1]\).

For two networks (\( n = 20 \) and \( n = 100 \)), we generate \( K \) sample pairs from \( K/5 \) trajectories (5 data pairs on each trajectory), with \( T_s = 0.2 \). Initial conditions are uniformly distributed on \([0, 2\pi]^n\). Note that we do not consider data points on \([0, 2\pi]\) but on the real line \( \mathbb{R} \) (i.e. without the modulo operation) where there is no discontinuity between 0 and 2\( \pi \). We use the dual method with Gaussian radial basis functions (with \( \gamma = 0.1 \)) and with \( N_F = n \) library functions

\[
\{1, \sin(\theta_1 - \theta_i), \ldots, \sin(\theta_{n-1} - \theta_i), \sin(\theta_{i+1} - \theta_i), \ldots, \sin(\theta_n - \theta_i)\}
\]

for the \( i \)th component of the vector field. ROC curves are shown in Figure 4 and the results are summarized in Table V, for different values of \( K \) and \( \sigma_{\text{meas}} \). They show that the dual method achieves good performance to reconstruct the whole network. In particular, with high threshold values, one can infer many true positive links with no false positive links.

2) Network with nonlinear couplings: We consider a network where each state is directly influenced by other states through \( n_{\text{inter}} \) quadratic and cubic nonlinearities. Each nonlinear interaction depends on at most two states. The dynamics of the system are given by

\[
\dot{x}_j = -C_j x_j + \sum_{k=1}^{n_{\text{inter}}} \xi_{j,k} \sigma_1(j,k) \sigma_2(j,k) \quad j = 1, \ldots, n
\]

where the coefficients \( \xi_j \) are chosen according to a uniform distribution on \([0, 1]\) and \( \sigma_{1,k} \) are distributed according to a Gaussian distribution of zero mean and standard deviation equal to one. The map \( \nu = (\nu_1, \nu_2) : \{1, \ldots, n\} \times \{1, \ldots, n_{\text{inter}}\} \to \{1, \ldots, n\}^2 \) randomly selects the subscripts and the map \( \sigma = (\sigma_1, \sigma_2) : \{1, \ldots, n\} \times \{1, \ldots, n_{\text{inter}}\} \to \{0, 1, 2, 3\}^2 \) randomly selects the exponents in such a way that \( \sigma_1(j,k) + \sigma_2(j,k) \in \{2, 3\} \). The first term in (51) is a linear term that ensures local stability of the origin. For several network sizes (\( n \in \{20, 50, 100\} \)), we generate \( K \) samples from \( K/2 \) trajectories (2 data pairs on each trajectory), with \( T_s = 0.5 \). Initial conditions are uniformly randomly distributed on \([-0.5, 0.5]^n\).

Although we could also consider the main method for small networks (typically \( n \leq 20 \)), we use only the dual method with Gaussian radial basis functions (with \( \gamma = 0.01 \)). The library functions are monomials of total degree less or equal to 3. The method provides an accurate estimation of the vector field and a good reconstruction of the network (Table VI). The ROC curves depicted in Figure 5(a) show that most of half of the links can be inferred with no false positive link (with high threshold values). As shown in Figure 5(b-d), the method is also efficient to infer the nature of the interactions (e.g. quadratic, cubic). Taking advantage of sparsity, it uses not more than 1000 sample points to identify up to 17,10^6 coefficients (most of which are zero). We finally note that, for larger networks, the use of monomials as library functions becomes too demanding in terms of memory. In this case, the dual method can still be used to estimate the value of the vector field at the sample points, but should be combined with other (regression) methods to infer the network.

<table>
<thead>
<tr>
<th>Table V</th>
<th>Results obtained with the dual method for the reconstruction of a network of Kuramoto oscillators.</th>
</tr>
</thead>
<tbody>
<tr>
<td>( n )</td>
<td>( K )</td>
</tr>
<tr>
<td>20</td>
<td>250</td>
</tr>
<tr>
<td>100</td>
<td>500</td>
</tr>
<tr>
<td>100</td>
<td>2000</td>
</tr>
</tbody>
</table>

Figure 4. ROC curves obtained with the dual method for the reconstruction of a network of Kuramoto oscillators.

VII. Conclusion

We have proposed a novel method for nonlinear systems identification. This method relies on a lifting technique developed in an operator-theoretic framework: it aims at identifying the linear Koopman operator in the space of observables. Key advantages of the method are that...
inference could be considered as a relevant approach. A careful study of the matrix logarithm used in the lifting method could also help to select the good branch (instead of the principal one), a strategy which might improve the performances when the sampling rate is low. Finally, theoretical results could also be obtained to provide bounds on the estimation error.

ACKNOWLEDGMENTS

The authors acknowledge J. Winkin and F. Lamoline for fruitful discussions and suggestions, and for their help in the proofs presented in the manuscript. They also wish to thank S. Brunton and N. Kutz for suggesting Kuramoto oscillators in the reconstruction problem. The authors acknowledge support from the Luxembourg National Research Fund. This paper presents research results of the Belgian Network DYSKO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme initiated by the Belgian Science Policy Office. This research used resources of the “Plateforme Technologique de Calcul Intensif (PTCI)” located at the University of Namur, Belgium, which is supported by the F.R.S.-FNRS under the convention No.2.5020.11. The PTCI is member of the “Consortium des Équipements de Calcul Intensif (CÉCI)”.

REFERENCES


Table VI

<table>
<thead>
<tr>
<th>N</th>
<th>w_exact</th>
<th>K</th>
<th>AUROC</th>
<th>NRMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>5</td>
<td>200</td>
<td>0.94</td>
<td>0.019</td>
</tr>
<tr>
<td>50</td>
<td>15</td>
<td>600</td>
<td>0.87</td>
<td>0.015</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
<td>1000</td>
<td>0.91</td>
<td>0.004</td>
</tr>
</tbody>
</table>

Figure 5. ROC curves and vector field coefficients obtained with the dual method for the reconstruction of a network with quadratic and cubic interactions. In (d), the inset shows a close-up of some estimated coefficients.

numerical schemes rely only on linear techniques and do not require the estimation of state time derivatives. For these reasons, this is a promising alternative to direct identification methods. As shown with several examples, the method is efficient to recover the vector field of several classes of systems, even from small time series with low sampling rate. Moreover, a dual method is also proposed to identify high-dimensional systems and is successfully applied to network reconstruction. Theoretical results also prove the convergence of the two methods in optimal conditions.

The results presented in this paper open the door to further developments and improvements of lifting techniques for nonlinear systems identification, some of which are related to recent advances in Koopman operator theory. For instance, identification lifting techniques with dictionary learning could be developed [44]. Extensions to general vector fields might also be considered, possibly without using library functions. Toward this end, lifting techniques could be combined with other methods: identify unknown parameters with Kalman filtering [45], consider rational functions in the vector field with alternating directions method [46], apply machine learning regression techniques on time derivatives estimated with the dual method, etc. Moreover, we might improve the method robustness to (measurement) noise and provide numerical schemes that are unbiased and consistent. In this context, Bayesian


**Alexandre Mauroy** is an Assistant Professor at the University of Namur, Belgium (Department of Mathematics and Namur Institute for Complex Systems). He received the M.S. degree in aerospace engineering and the Ph.D. degree in applied mathematics from the University of Liège, Belgium, in 2007 and 2011, respectively. Prior to joining UNamur, he was a postdoctoral researcher at the University of California Santa Barbara from 2013 to 2015, at the University of Liège from 2013 to 2015, and at the University of Luxembourg in 2016. His research interests include synchronization in complex networks, network identification, and applications of operator theoretic methods to dynamical systems and control.

**Jorge Goncalves** is a Professor at the Luxembourg Centre for Systems Biomedicine, University of Luxembourg and a Principal Research Associate at the Department of Engineering, University of Cambridge. He received his Licenciatura (5-year S.B.) degree from the University of Porto, Portugal, and the M.S. and Ph.D. degrees from the MIT, Cambridge, MA, all in Electrical Engineering and Computer Science, in 1993, 1995, and 2000, respectively. He then held two postdoctoral positions, first at the MIT for seven months, and from 2001 to 2004 at the California Institute of Technology (Control and Dynamical Systems Division). At the Information Engineering Division of the Department of Engineering, University of Cambridge, he was a Lecturer from 2004 until 2012, a Reader from 2012 until 2014, and since 2014 he is a Principal Research Associate. From 2005 until 2014 he was a Fellow of Pembroke College, University of Cambridge. From June to December 2010 and January to September 2011 he was a visiting Professor at the University of Luxembourg and California Institute of Technology, respectively.