Data-Driven State Observation for Nonlinear Systems based on Online Learning

Wentao Tang[1]

[1]Wentao Tang, Department of Chemical and Biomolecular Engineering, North Carolina State University

Correspondence

Wentao Tang, Campus Box 7905, North Carolina State University, Raleigh, NC 27695 U.S.A.

Email: wentao_tang@ncsu.edu

Funding information

This work is supported by the author's startup fund from NC State University.

This paper considers the problem of state observation for nonlinear dynamics. While model-based observer synthesis is difficult due to the need of solving partial differential equations, this work proposes an efficient model-free, data-driven approach based on online learning. Specifically, by considering the observer dynamics as a Chen-Fliess series, the estimation of its coefficients has a least squares formulation. Since the series converges only locally, the coefficients are recursively updated, resulting in an online optimization scheme driven by instantaneous gradients. When the state trajectories are available, the online least squares guarantees an ultimate upper bound of average observation error proportional to the average variation of states. In the realistic situations where the states cannot be measured, the immersed linear dynamics based on the Kazantzis-Kravaris/Luenberger structure is assigned, followed by online kernel principal component analysis for dimensionality reduction. The proposed approach is demonstrated by a limit cycle dynamics and

a chaotic system.

KEYWORDS

State observer, nonlinear systems, online optimization, dimensionality reduction

1 | INTRODUCTION

Nonlinearity is a common characteristic of the dynamics of chemical processes, associated with complex phenomena such as bifurcations, limit cycles, and chaos.[1,2,3] In nonlinear process control methods such as nonlinear model predictive control (MPC)[4] and differential geometry-based input–output linearization,[5] *state-space representations* of the process dynamics are typically used. In the recent years, the development of data-driven modeling and control strategies, such as those based on neural network models,[6] reinforcement learning,[7] Koopman operators,[8,9] Gaussian processes,[10,11] sparse system identification,[12,13] and dissipative dynamic behaviors,[14,15] has become an important direction for process control. Most of these strategies focus on analyzing the dynamics on the state space and/or designing a state-feedback controller. In a realistic process control setting, however, one should have data access to only manipulated input and measurable output variables, instead of all the states. It is therefore necessary to infer the state values from the measured variables through a *state observer*.[16]

The classical form of state observer, known as Luenberger observer,[17] refers to a linear dynamics where the state estimates are driven by the signal of innovations, namely the differences between measured outputs and their predicted values. For linear systems, a probabilistic approach can be used to derive the optimal Luenberger observer, namely the well-known (Stratonovich-)Kalman(-Bucy) filter, which can be extended to nonlinear systems but only locally.[18] To obtain a global Luenberger-type observer for nonlinear systems, the seminar work of Kazantzis and Kravaris[19] first proposed to find a static mapping on the state space, under which the transformed states compose a linear dynamics driven by the measured outputs; thus, this static mapping can be inverted to obtain desirable state estimates. The existence of such a *Kazantzis-Kravaris/Luenberger (KKL) observer*, while originally proved under restrictive assumptions, was re-established under milder conditions by Andrieu and Praly[20] and extended from autonomous to controlled systems by Bernard and

Andrieu.[21] The linear dynamics of the transformed states, as long as being controllable and having a sufficient order, can be assigned almost arbitrarily, since the prohibited choices of poles form only a set of zero Lebesgue measure.[22]

The main challenge of synthesizing KKL observers lies in the associated numerical solution. In order to find a mapping (immersion) to transform the states so as to conform to a linear dynamics, partial differential equations (PDEs) that involve the process model need to be solved. To avoid PDE solution, Ramos et al.[23] proposed to train a neural network that represents the inverse transformation from immersed states to the actual states. Buisson-Fenet et al.[24] further considered the tuning of poles in the embedded linear dynamics along with the neural network training. A more sophisticated approach by Niazi et al.[25] adopted the idea of physics-informed neural networks (PINNs) and used two neural networks – one for the immersion and one for its inverse; the loss metric for their training includes a state reconstruction error and a prediction error of the embedding. In a different vein, Miao and Gatsis[26] regarded the training problem as a dynamic optimization problem to minimize accumulated squared state observation error, whose optimality condition (i.e., Pontryagin's minimum principle) leads to neural ordinary differential equations (neural ODEs). The main limitation of the above-mentioned works is the neural architecture, whose training inevitably causes nonconvex optimization problems that can be expensive to compute or trapped in local optima.

In this work, a new method for nonlinear state observation is proposed. This method is based on *Chen-Fliess series*[27,28] as an input–output representation of the observer dynamics. In the Chen-Fliess series, the observer outputs (i.e., state estimates) are expressed as linear combinations of unknown coefficients (arising from Lie derivatives) multiplied by the corresponding nonlinear bases (recursive integrals on the trajectories of observer inputs, namely the process outputs). This results in a *least squares formulation* for state observation, with the following complications.

1. Since the convergence of Chen-Fliess series is guaranteed only within a finite time window (starting from the instant at which the expansion is performed), the coefficients need to be updated recursively online. Therefore, a gradient dynamics can be assigned to specify the flow of Chen-Fliess coefficients in continuous time, which leads to an *online optimization* scheme. This paper proves that under appropriate assumptions, the online optimization guarantees a finite average state observation error that is proportional to the average rate of variation in the true states.

2. If the state measurements are unavailable (which is the realistic case), a direct least-squares formulation to minimize state observation errors may not be possible. In this situation, the transformed states are first obtained according to their KKL linear dynamics driven by the process outputs. Since the transformed states are of a higher dimension than the actual state space, a dimensionality reduction scheme – kernel principle component analysis (KPCA)[29] – is adopted to obtain the principal coordinates, which are essentially diffeomorphisms of, and thus considered as equivalents to, the underlying state variables. The KPCA is performed via an online algorithm, with coefficients of all principal components updated recursively.

The proposed approach of online optimization of Chen-Fliess series, with its simple mathematical structure, involves only computationally efficient routines with a polynomial complexity in the output dimensions where the degree is related to the series truncation. The auxiliary KPCA approach for dimensionality reduction is also based on a lightweighted recursive update of inverse kernel matrix and a gradient flow for coefficient updates. The online KPCA has a guaranteed finite and tunable complexity that depends on the state space. Such efficiency in computation and sampling, along with the provable bound on the state observation error, makes the proposed approach promising to be integrated with nonlinear control, especially data-driven control frameworks.

The remainder of this paper is organized as follows. In Section 2, the preliminaries of KKL observers, Chen-Fliess series, and online KPCA are introduced. The proposed online optimization approach is provided and the performance guarantee is established in Section 3. The case studies on two representative nonlinear dynamics – an oscillator and a Lorentz chaotic system, both of which can be considered as reaction kinetics – are demonstrated in Section 4. Conclusions are made in Section 5.

2 | KKL OBSERVER AND CHEN-FLIESS SERIES

Consider a nonlinear, autonomous dynamical system

$$\dot{x}(t) = F(x(t)), \quad y(t) = H(x(t)) \tag{1}$$

where $x(t) \in \mathcal{X} \subseteq \mathbb{R}^n$ and $y(t) \in \mathbb{R}^m$ are the vector of states and outputs, respectively. F and H are assumed to be smooth vector fields on $\mathcal{X}$, but their expressions may not be

known. Let us assume for simplicity that $m = 1$.

2.1 | KKL Observer and Its Existence

For the nonlinear system (1), a KKL observer refers to an auxiliary dynamical system in the following form:

$$\dot{z} = Az + By, \quad \hat{x} = T^{\dagger}(z). \tag{2}$$

Here the observer states $z \in \mathbb{R}^{n_z}$ is driven by the process outputs through a linear dynamics, (A, B). It is required that (i) (A, B) should be controllable (i.e., the controllability matrix $[B, AB, \ldots, A^{n_z-1}B]$ has rank n_z), (ii) A should be Hurwitz (with all eigenvalues residing on the open left half complex plane), and in addition, (iii) $T^{\dagger}$ is a continuous mapping from $\mathbb{R}^{n_z} \to \mathbb{R}^n$. To obtain a KKL observer, it suffices to find an *immersion* $T : \bar{\mathcal{X}} \to \mathbb{R}^{n_z}$, namely a differential mapping that is injective over the closure of $\mathcal{X}$, that satisfies the following PDE system:

$$\frac{\partial T}{\partial x}(x)F(x) = AT(x) + BH(x), \quad \forall x \in \mathcal{X}. \tag{3}$$

Then, by letting $T^{\dagger}$ to be the left-inverse of the mapping T, namely the one such that $T^{\dagger}(T(x)) = x, \forall x \in \mathcal{X}$ (in other words, $T^{\dagger} \circ T = \mathrm{id}$), the KKL observer is found.

Now we introduce the conditions for the existence of such an immersion to solve the above PDE system. To this end, the conclusion of Brivadis et al.[22] is provided. Use the following notations.

- The solution to the ODEs $\dot{x}(t) = F(x(t))$ with initial conditions $x(0) = x_0$ at time t is denoted as $X(x_0, t)$.
- Given an open set $\mathcal{O} \in \mathbb{R}^n$, denote $\varsigma^-_{\mathcal{O}}(x_0) < 0$ as the infimum of such t that $X(x_0, t)$ exists and is confined in $\mathcal{O}$.
- Write $\mathcal{O} + \epsilon := \{x + \tilde{x} | x \in \mathcal{O}, |\tilde{x}| < \epsilon\}$.

Definition 1 *The system* (1) *is said to be* $(\mathcal{O}, \epsilon)$-backward distinguishable *if for any* x_a *and* x_b *in* $\mathcal{X}$, *unless* $x_a = x_b$, *there must be a time* $0 > t > \max(\varsigma^-_{\mathcal{O}+\epsilon}(x_a), \varsigma^-_{\mathcal{O}+\epsilon}(x_b))$ *such that the backward solution at this time results in different outputs, i.e.,* $H(X(x_a, t)) \neq H(X(x_b, t))$.

The following theorem states that the two matrices A and B for the immersed linear dynamics can be almost arbitrarily chosen, as long as the spectrum of A is bounded on the left side of a left-translated imaginary axis $(-\rho + i\mathbb{R})$.

Fact 1 *Suppose that for some open bounded set $O \subseteq \bar{X}$ and $\epsilon > 0$, the system* (1) *is (O, ϵ)-backward distinguishable. Then there exists a constant $\rho > 0$ such that almost for all $(A, B) \in \mathbb{R}^{(2n+1)\times(2n+1)} \times \mathbb{R}^{2n+1}$ with $A + \rho I$ Hurwitz, there is an immersion $T : O \to \mathbb{R}^{2n+1}$ that solves the PDE system* (3).

In the above theorem, the wording "almost" refers to the exclusion of a zero Lebesgue measure set. Although the numerical solution to find this mapping T can be difficult, it is established that $n_z = 2n + 1$ is a sufficient order of the state observer. A slightly more restrictive construction[20] is to assign $n + 1$ (complex) poles $\{\lambda_i\}_{i=1}^{n+1}$ and let A be a diagonal matrix comprising of blocks $[\mathrm{Re}(\lambda_i), -\mathrm{Im}(\lambda_i); \mathrm{Im}(\lambda_i), \mathrm{Re}(\lambda_i)]$ and B be a repetition of $[1, 0]^\top$ for $n+1$ times. This leads to an observer dynamics of order $n_z = 2n+2$. Nevertheless, if A is allowed to be complex, it suffices to let A be a diagonal matrix of order $n + 1$. According to Andrieu and Praly,[20] when $n_z = n + 1$ and $A \in \mathbb{C}^{(n+1)\times(n+1)}$, still, the excluded choices form only a zero-measure set. Thus, one should usually be able to choose $n + 1$ real poles, and let $A = \mathrm{diag}(\lambda_1, \ldots, \lambda_{n+1})$ and $B = [1, 1, \ldots, 1]^\top$. In this paper, we will use $n_z = n+1$ instead of $2n + 1$.

The determination of inverse immersion mapping $T^\dagger$, however, is indeed challenging. A simplifying routine was recently proposed by Brivadis et al.[22] based on reducing the vector-valued immersion $T : O \to \mathbb{R}^{n_z}$ to a parameterized but scalar-valued $T_0 : \Lambda \times O \to \mathbb{R}$, where $\Lambda \subseteq (-\infty, 0)$ can be interpreted as a range of admissible poles. Specifically, if for any $\lambda \in \Lambda$, the PDE:

$$\frac{\partial T_0}{\partial x}(\lambda, x) F(x) = \lambda T_0(\lambda, x) + H(x) \tag{4}$$

admits an analytic and bounded solution, then the vector-valued T mapping can be obtained by repetitively differentiating $T_0(\lambda, x)$ over λ. Yet solving this parametric PDE is still difficult and is further complicated by the differentiation. The neural network-based approaches proposed in existing works,[23,24,25] although built on the sound basis that neural networks act as universal approximators, theoretically are not exempt from the nonconvexity of training problems and local solutions. In query for a more desirable problem structure, we focus our attention on *Chen-Fliess series*, which provides a linear parameterization of the nonlinear observer dynamics.

2.2 | Chen-Fliess Series and Its Convergence

Suppose that the nonlinear dynamics of the state observer is affine in y (which is satisfied by the KKL observers):

$$\dot{z} = g_0(z) + \sum_{i=1}^{m} g_i(z)y, \quad \hat{x} = h(z). \tag{5}$$

For each component of the state estimate $\hat{x}_j = h_j(z), j = 1, \ldots, n$, define the following Lie derivatives:

$$L_{g_i}h_j(z) = \frac{\partial h_j}{\partial z}g_i(z), \quad i = 0, 1, \ldots, m \tag{6}$$

and recursively one can define high-order Lie derivatives:

$$\begin{aligned} L_{g_{i_k}} \cdots L_{g_{i_2}} L_{g_{i_1}} h_j(z) =& \frac{\partial}{\partial z}\left(\ldots \frac{\partial}{\partial z}\left(\frac{\partial h_j}{\partial z} g_{i_1} \right) g_{i_2} \ldots \right) g_{i_k}(z), \\ & i_1, \ldots, i_k = 0, 1, \ldots, m, \quad j = 1, \ldots, n. \end{aligned} \tag{7}$$

Let us call a tuple of the form $(i_1 i_2 \ldots i_k)$ as a *multi-index* of length k and denote the set of multi-indices of length k, namely $\{0, 1, \ldots, m\}^k$ as $\mathbb{I}_m^k$. The above recursive Lie derivative is thus simply written as $L_\mu h_j(z)$, with $\mu = (i_1 i_2 \ldots i_k) \in \mathbb{I}_m^k$. The set of multi-indices with lengths not exceeding K is denoted as $\mathbb{I}_m^{\leq K} = \cup_{k=0}^{K} \mathbb{I}_m^k$. In particular, the multi-index of zero length is written as $\varnothing$, corresponding to a Lie derivative of the function itself: $L_\varnothing h_j(z) = h_j(z)$. Also define the integrals

$$E_i(t_0, t_1) = \int_{t_0}^{t_1} y_i(\tau) d\tau, \quad i = 0, 1, \ldots, m, \quad t_0, t_1 \in \mathbb{R}, \quad t_0 \leq t_1. \tag{8}$$

where the convention $y_0(t) \equiv 1$ is adopted. Recursively, for multi-indices, one can define

$$E_{i_1 i_2 \ldots i_k}(t_0, t_1) = \int_{t_0}^{t_1} E_{i_1 i_2 \ldots i_{k-1}}(t_0, \tau) y_{i_k}(\tau) d\tau, \quad k \geq 2. \tag{9}$$

For $k = 0$, it is natural to let $E_\varnothing(t_0, t_1) = 1$.

Fact 2 *Suppose that $g_0, g_1, \ldots, g_m$ and h_j are analytic vector fields, then for any z there exist $M_1 \geq 0$ and $M_2 \geq 0$ such that for all $\mu \in \mathbb{I}_m^k$, $|L_\mu h(z)| \leq M_1 M_2^k k!$. If we assume further that each input signal for the state observer $y_j(\cdot)$ is bounded within $[-M_0, M_0]$ for some $M_0 \geq 0$,*

then there exists a horizon $\bar{\Delta} = 1/(m+1)M_0M_2$ *such that for any* $t \in \mathbb{R}$ *and* $\Delta \in [0, \bar{\Delta})$,

$$\hat{x}_j(t+\Delta) = \sum_{k=0}^{\infty} \sum_{\mu \in \mathbb{I}_m^k} L_\mu h_j(z(t)) E_\mu(t, t+\Delta). \tag{10}$$

The above theorem (cf. Isidori[27]) shows that the value of $\hat{x}_j$ within a time window Δ can be expressed as a infinite linear combination of the Lie derivatives (including the original h_j) at the beginning of this window multiplied by the recursive integrals of the input signals. This infinite series is called the Chen-Fliess series. In fact, if we truncate the Fliess series to neglect the terms corresponding to multi-indices longer than K digits, then the truncation error is bounded by

$$\left| \hat{x}_j(t+\Delta) - \sum_{\mu \in \mathbb{I}_m^{\leq K}} L_\mu h_j(z(t)) E_\mu(t, t+\Delta) \right| \leq \sum_{k=K+1}^{\infty} M_1 [(m+1)M_2\Delta]^k = M_1 \frac{[(m+1)M_2\Delta]^{K+1}}{1-(m+1)M_2\Delta}. \tag{11}$$

Hence, if the value of $x_j(t+\delta)$ at all $0 \leq \delta \leq \Delta$ is assumed to be known, then by obtaining the recursive integral values $E_\mu(t, t+\delta)$, the Lie derivatives at t, i.e., $L_\mu h_j(z(t))$ can be approximated by a least squares procedure, which is capable of predicting the future values of $\hat{x}_j$ for at least $\bar{\Delta} - \Delta$ time ahead with a bounded error. Of course, such extrapolation of the Chen-Fliess series will result in exponentially growing errors as the time proceeds. Therefore, the coefficients of the Chen-Fliess series, namely the Lie derivative values, must be updated continuously in accordance with the evolution of states $z(t)$. Such an online optimization scheme will be discussed in the next section. To this end, let us first denote the column vectors formed by vertically stacking the Lie derivatives and integrals as

$$\theta_j(t) = \left[L_\mu h_j(z(t)) \right]_{\mu \in \mathbb{I}_m^{\leq K}}, \quad \phi(t, \delta) = \left[E_\mu(t, t+\delta) \right]_{\mu \in \mathbb{I}_m^{\leq K}}, \tag{12}$$

respectively. Then we can write, for all $\delta \in [0, \Delta]$,

$$\left| \hat{x}_j(t+\delta) - \theta_j(t)^\top \phi(t, \delta) \right| \leq r(K, \delta) := M_1 \frac{[(m+1)M_2\delta]^{K+1}}{1-(m+1)M_2\delta}. \tag{13}$$

For a KKL observer, based on its governing PDE (3) and the fact that $A + \rho I$ is Hurwitz, it is guaranteed that $\|T(x(t)) - T(\hat{x}(t))\|$ decays at a rate of $e^{-\rho t}$. Since T is an immersion and thus $T^\dagger$ is a continuous mapping, if $\bar{\mathcal{X}}$ is bounded, then given enough amount of time, $\|x(t) - \hat{x}(t)\|$ becomes sufficiently small, e.g., below $\varepsilon r(K, \delta)$ for some $\varepsilon > 0$. We can thus

have

$$\left|x_j(t+\delta) - \theta_j(t)^\top \phi(t,\delta)\right| \leq r(K,\delta), \tag{14}$$

where $r(K,\delta)$ is actually multiplied by $(1+\varepsilon)$ without changing notation for simplicity. This allows us to use online (supervised) learning to track the trajectory of state measurements. However, in realistic settings, the states can not be assumed to be measurable. To this end, we need to construct such state trajectories through some unsupervised learning approach and then track these constructed trajectories, which makes the overall state observation problem a semi-supervised one.

2.3 | Online Kernel Principle Component Analysis (KPCA)

Note that the states' information is actually encoded in the transformed variables $z(t) \in \mathbb{R}^{n_z}$, and the computation of $z(t)$ is feasible by simulating the linear dynamics $\dot{z} = Az + By$ which is a priori known. Thus, this work proposes to use a nonlinear dimensionality reduction approach – KPCA – to reduce the trajectory of $z(t)$ onto a n-dimensional manifold.

In KPCA,[30] a kernel function $\kappa(z,z')$ is defined to capture the similarity between the (finite or infinite) features of any two points z and z', i.e., $\kappa(z,z') = \varphi(z)^\top \varphi(z')$ (but the feature map φ does not need to be defined explicitly). A common choice is the Gaussian kernel:

$$\kappa(z,z') = \exp\left(-\frac{1}{2\sigma^2}\|z-z'\|^2\right). \tag{15}$$

Suppose that one has collected an offline sample of N points $z^{(1)}, z^{(2)}, \ldots, z^{(N)}$, then by calculating the symmetric kernel matrix $G \in \mathbb{R}^{N\times N}$, whose (j_1,j_2)-th entry is $G_{j_1 j_2} = \kappa\left(z^{(j_1)}, z^{(j_2)}\right)$, and determining its n eigenvectors $\{\alpha_i\}_{i=1}^n$ associated with the n largest eigenvalues $\{\varrho_i\}_{i=1}^n$ ($\varrho_1 \geq \varrho_2 \geq \cdots \geq \varrho_n \geq 0$), one can consider the following mappings:

$$\pi_i : \mathbb{R}^{n_z} \to \mathbb{R}, \quad \pi_i(z) = \sum_{j=1}^{N} \alpha_{ij}\kappa(z,z_j) =: \alpha_i^\top \kappa(z), \quad i = 1,\ldots,n \tag{16}$$

as the n principal components of any z. Here $\kappa(z)$ stands for the vector of kernel evaluations with all sample points. The eigenvectors $\alpha_1,\ldots,\alpha_n$ are mutually orthogonal. The principle components $P(z) = [\pi_1(z),\ldots,\pi_n(z)]$ in fact have specified n local coordinates

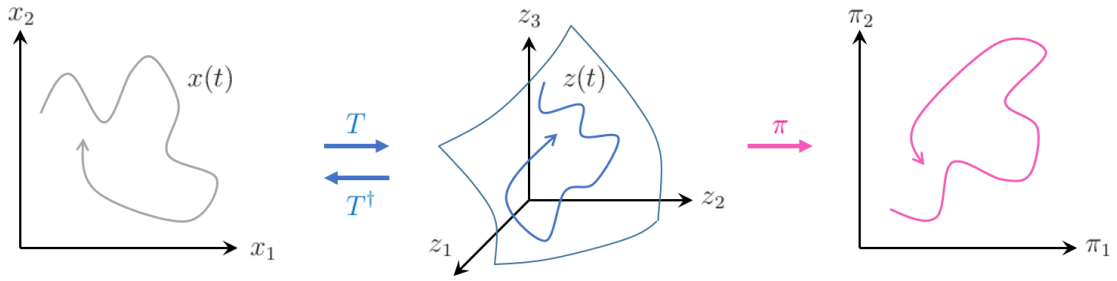


FIGURE 1 Dimensionality reduction for constructing unmeasurable states.

for any $z \in \mathbb{R}^{n_z}$. To see this, it suffices that the Jacobian $d\pi/dz$ is nonsingular. Since

$$\frac{dP}{dz}(z) = \alpha^\top \frac{d\kappa}{dz}(z) \tag{17}$$

where α is the matrix $[\alpha_1, \ldots, \alpha_n]$ horizontally stacking the eigenvectors, and

$$\frac{d\kappa}{dz}(z) = -\frac{1}{\sigma^2}\begin{bmatrix} \kappa(z, z^{(1)})(z - z^{(1)})^\top \\ \vdots \\ \kappa(z, z^{(N)})(z - z^{(N)})^\top \end{bmatrix}, \tag{18}$$

the Jacobian can be rank-deficient only when $[z - z^{(1)}, \ldots, z - z^{(N)}]$ has a rank less than n, which is possible only when $\{z^{(j)}\}_{j=1}^N$ spans a subspace with dimension less than n. As such, $x \mapsto z = T(x) \mapsto \pi = P(z)$ generally forms a differentiable bijection, i.e., $P \circ T$ is a diffeomorphism. By dimensionality reduction, we have constructed π as the "equivalents" of the unmeasurable states x. An illustration of this idea is given in Fig. 1.

However, in a dynamical system, the z data is available in an online data stream. Suppose that the data is passed from the linear dynamics simulator to a KPCA machine at discretized sampling times $k = 0, 1, 2, \ldots$, with sampling interval τ^* so that $t = k\tau^*$. We need to add newly sampled points $z(t)$ into the dataset as time proceeds. On the other hand, we should control the complexity of KPCA, N, to avoid endless accumulation of sample points. Therefore, the approach of Honeine[29] is adopted here. At each sampling time k, the following steps are executed.

- First, check whether the kernel values evaluated based on the existing data points can well approximate the new kernel value. That is, calculate

$$\beta = G^{-1}\kappa(z(t)), \quad \varepsilon^2 = \kappa(z(t), z(t)) - \kappa(z(t))^\top \beta. \tag{19}$$

- If ε^2 is larger than a threshold value v, then
  - the dataset is augmented with the current $z(t)$;
  - every eigenvector α_i is appended with a zero element to passively reflect the addition of a new point;
  - the inverse kernel matrix G^{-1} gets one more row and one more column, and adds a rank-1 matrix:

$$G^{-1} \leftarrow \begin{bmatrix} G^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \frac{1}{\varepsilon^2} \begin{bmatrix} -\beta \\ 1 \end{bmatrix} \begin{bmatrix} -\beta^\top & 1 \end{bmatrix}; \tag{20}$$

  - the β is substituted by $[0, \ldots, 0, 1]^\top$.
- Update the coefficients using a gradient flow dynamics:

$$\alpha = \alpha + \lambda(t) \left[\beta \pi^\top - \alpha U(\pi) \right], \tag{21}$$

where $\pi = \alpha^\top \kappa(z(t))$ and $U(\pi)$ is the matrix $\pi\pi^\top$ with lower triangle ground to zero. For a detailed justification of each step, the reader is referred to Hyvärinen and Oja[31] or Kim, Franz, and Schölkopf.[32] The step size $\lambda(t)$, as suggested in Honeine,[29] should be time-varying in the form of

$$\lambda(t) = \frac{\lambda_0}{1 + t/\tau_\lambda}. \tag{22}$$

It was proved[29] that the online KPCA algorithm introduced above is capable of curbing the growth of the order N by guaranteeing the spacing between the points.

Fact 3 *For any two points, z and z', added during online KPCA, $\|z - z'\|^2 \geq d_{\min}^2 := -2\sigma^2 \ln(1 - v/2)$.*

If $\bar{\mathcal{X}}$ is compact and the sampling is performed only after a certain time, then the range from which z can be sampled is also bounded, say, within a set $\mathcal{Z}$. Suppose that $\mathcal{Z}$ has a diameter d_z. It then follows that the number of points added to the online KPCA, N, is bounded roughly by

$$N \leq \lceil d_z / d_{\min} \rceil^{n_z}. \tag{23}$$

Remark 1 *In fact, if the system* (1) *has an attractor $\mathcal{A}$ and its basin of attraction contains the entire $\bar{\mathcal{X}}$, then there exists a certain time instant T_0, after which the states are restricted within a small distance from the attractor, and thus the $z(t)$ is confined near a corresponding forward*

invariant set $\mathcal{A}_z$. If we do not start dimensionality reduction until after T_0, then the order of KPCA, N, is more tightly bounded according to $\mathcal{A}_z \subseteq \mathcal{Z}$. To quantify the coverage of $\mathcal{A}_z$ by balls of diameter $d_{\min}$, the concept of Hausdorff measure[33] can be used.

Remark 2 *Kernel methods in machine learning have been used extensively in the literature for systems identification, such as for Wiener-Hammerstein processes[34] and Volterra series[35] (also see Pillonetto[36] for a survey). Among these works, the kernel canonical variate analysis (KCCA) approach for nonlinear state-space system identification proposed by Verdult et al.[37] involves the reconstruction of states from input and output measurements, and is related to the KPCA approach in this work. The difference is that since we do not consider the existence of inputs in the observed system, the computable signals $z(t)$ can be directly reduced to states.*

3 | ONLINE OPTIMIZATION AND ITS PERFORMANCE

3.1 | Online Least-Squares for Chen-Fliess Series

With the discussions on the convergence property of Chen-Fliess series in the above section, now let us consider the problem of learning such a series online. That is, for any time $t \in \mathbb{R}$ and state index $j = 1, \ldots, n$, given the trajectory $y(\cdot)$ and thus the resulting recursive integrals $E_\mu(t, \cdot)$ on $[t, t + \Delta]$, we are interested in *the online estimation of coefficient vector* $\theta_j(t)$. Let us first consider the hypothetical situation where the true states' trajectories are available. If the truncation length of multi-indices, K, is sufficiently large, we can approximately regard this problem as a least squares problem of minimizing the integrated squared errors during the time interval. That is,

$$\min_{\theta_j} \quad J(\theta_j, t) := \frac{1}{2} \int_0^\Delta \left(\theta_j^\top \phi(t, \delta) - x_j(t + \delta) \right)^2 d\delta. \tag{24}$$

The optimal solution exists uniquely if this quadratic form of θ_j is strictly positive definite. Therefore, we make the following natural assumption, and in this subsection we discuss the properties of online learning under it.

Assumption 1 *The trajectory of y is such that for any $t \in \mathbb{R}$ and $\Delta > 0$, there exists a corresponding $\alpha_\Delta > 0$, such that the following matrix inequality is satisfied:*

$$\Phi(t) := \int_0^\Delta \phi(t, \delta) \phi^\top(t, \delta) d\delta \succeq \alpha_\Delta I. \tag{25}$$

By $A \succeq B$ for two real symmetric matrices A and B of the same shape, it is meant that $A - B$ is a positive semidefinite matrix. I refers to the unit matrix of proper order.

This assumption shall be referred to as that of *persistent excitation*, and under this assumption, the outputs y are said to be *persistently excited*. Hence, the optimal solution at time $t + \Delta$ can be found as

$$\theta_j^*(t) = \Phi^{-1}(t)\psi_j(t), \text{ where } \psi_j(t) = \int_0^\Delta x_j(t+\delta)\phi(t,\delta)d\delta, \quad j = 1, \ldots, n. \tag{26}$$

However, it is not realistic to obtain the optimal solution $\theta_j^*(t)$ in continuous time, since the linear algebraic computation involved in a matrix inversion may not be conducted in real time and $\Phi(t)$ may have a large condition number. Thus, let us consider using *online optimization*,[38] i.e., letting the solution of θ_j evolve according to some governing dynamics. Clearly, if we directly differentiate $\theta_j^*(t)$ over t, the resulting derivative involves $\dot{\psi}_j(t)$, which in turn requires the knowledge of $\dot{x}_j(t+\delta)$ for $\delta \in [0, \Delta]$. Since directly differentiating a measurement signal is improper, it is impossible to perfectly realize the true dynamics of $\theta_j^*(t)$. Instead, it appears reasonable to drive the evolution of $\theta_j(t)$ by the gradient of the objective function $J(\theta_j, t)$ with respect to θ_j, with ignorance of how the target trajectory x_j varies. Thus, we let

$$\dot{\theta}_j(t) = -\eta \nabla J(\theta_j(t), t) = -\eta \left(\Phi(t)\theta_j(t) - \psi_j(t)\right) \tag{27}$$

This approach is known as *online gradient descent*.[39] The rate $\eta > 0$, which is set to be time-invariant, is a tunable hyper-parameter.

3.2 | Performance under the Persistent Excitation Assumption

A typical performance metric for assessing an online optimization algorithm is *dynamic regret*, which is defined as the total error between the online objective and the truly optimal value or the integrated squared distances between the online solution and the true optimum (if the optimum was reached at all times). Similar to the conclusion of Mokhtari et al.[40] in a discrete-time setting, a bound on the dynamic regret, can be established. This will imply a provable bound on the average squared error of state observation. To this end, the following propositions are presented.

Proposition 1 *Under Assumption 1, for any $t \geq 0$, it is guaranteed that*

$$\int_0^t \|\theta_j(\tau) - \theta_j^*(\tau)\|^2 d\tau \leq C_1 + C_2^2 \int_0^t \|\dot{\theta}_j^*(\tau)\|^2 d\tau \tag{28}$$

for some constants $C_1, C_2 > 0$. In particular, $C_1 = 2\|\theta_j(0) - \theta_j^*(0)\|^2/\eta\alpha_\Delta$ and $C_2 = 2/\eta\alpha_\Delta$ can be chosen.

Proof Consider the difference between $\theta_j(t)$ and $\theta_j^*(t)$. We have

$$\frac{d}{dt}\frac{1}{2}\|\theta_j(t) - \theta_j^*(t)\|^2 = \left(\theta_j(t) - \theta_j^*(t)\right)^\top \left(-\eta\nabla J(\theta_j(t), t) - \dot{\theta}_j^*(t)\right). \tag{29}$$

Due to the persistent excitation assumption, J is α_Δ-strongly convex in its θ argument, which implies that

$$\left(\theta_j(t) - \theta_j^*(t)\right)^\top \nabla J(\theta_j(t), t) \geq \frac{\alpha_\Delta}{2}\|\theta_j(t) - \theta_j^*(t)\|^2. \tag{30}$$

Hence,

$$\begin{aligned}\frac{d}{dt}\frac{1}{2}\|\theta_j(t) - \theta_j^*(t)\|^2 &\leq -\frac{\eta\alpha_\Delta}{2}\|\theta_j(t) - \theta_j^*(t)\|^2 + \left(\theta_j(t) - \theta_j^*(t)\right)^\top \dot{\theta}_j^*(t) \\ &\leq -\frac{\eta\alpha_\Delta}{4}\|\theta_j(t) - \theta_j^*(t)\|^2 + \frac{1}{\eta\alpha_\Delta}\|\dot{\theta}_j^*(t)\|^2.\end{aligned} \tag{31}$$

Rearranging the above inequality and integrating on interval $[0, t]$, we obtain

$$\int_0^t \|\theta_j(\tau) - \theta_j^*(\tau)\|^2 d\tau \leq \frac{2}{\eta\alpha_\Delta}\left(\|\theta_j(0) - \theta_j^*(0)\|^2 - \|\theta_j(t) - \theta_j^*(t)\|^2\right) + \frac{4}{\eta^2\alpha_\Delta^2}\int_0^t \|\dot{\theta}_j^*(\tau)\|^2 d\tau. \tag{32}$$

The subtracted term in the parenthesis must be nonnegative. The proposition is proved.

We note that the error is found to be proportional to the rate of variation of the true solution to the least squares problem, $\dot{\theta}_j^*(t)$. Since this true solution is not actually obtained, it is desirable to interpret its variation based on that of the states. For this, we first bound the rate of change of $\dot{\theta}_j$ by that of the observer states z, and then bound the variation rate of observer states by that of the process states x.

Proposition 2 *Suppose that $\bar{\mathcal{X}}$ is compact (implying that all components of $y(t)$ are bounded in $[-M_0, M_0]$ for some $M_0 > 0$), $\Delta < \bar{\Delta}/2 = 1/2(m+1)M_0M_2$ (where M_2 is the growth rate of*

Lie derivatives with multi-index length). Then there exist constants $C_3, C_4 \geq 0$ such that

$$\|\dot{\theta}_j^*(t)\| \leq C_3\|\dot{z}(t)\| + C_4. \tag{33}$$

Proof For our later proof, let us investigate the derivative of recursive integrals $E_\mu(t, t+\delta)$ for $\mu = (i_1 i_2 \dots i_k)$ and $\delta \in [0, \Delta]$. By definition,

$$\frac{d}{dt}E_\mu(t, t+\delta) = y_{i_k}(t+\delta)E_{i_1\dots i_{k-1}}(t, t+\delta) - y_{i_k}(t)E_{i_1\dots i_{k-1}}(t, t) + \int_t^{t+\delta} y_{i_k}(\tau)\frac{\partial E_{i_1\dots i_{k-1}}(s,\tau)}{\partial s}\Big|_{s=t} d\tau. \tag{34}$$

The second term is clearly zero. When $|y_{i_k}| \leq M_0$, $|E_{i_1\dots i_{k-1}}(t, t+\delta)| \leq (M_0\delta)^{k-1}/(k-1)!$, and thus

$$\left|\frac{d}{dt}E_\mu(t, t+\delta)\right| \leq M_0\frac{(M_0\delta)^{k-1}}{(k-1)!} + M_0\delta \max_{\tau\in[t,t+\delta]}\left|\frac{\partial E_{i_1\dots i_{k-1}}(s,\tau)}{\partial s}\Big|_{s=t}\right|. \tag{35}$$

For expressions in the form of $\partial E_\mu(s,t)/\partial s$, by definition,

$$\frac{\partial E_{i_1\dots i_{k-1}i_k}(s,t)}{\partial s} = -y_{i_k}(s)E_{i_1\dots i_{k-1}}(s,t) + \int_s^t y_{i_k}(\tau)\frac{\partial E_{i_1\dots i_{k-1}}(s,\tau)}{\partial s} d\tau. \tag{36}$$

Thus,

$$\max_{\tau\in[t,t+\delta]}\left|\frac{\partial E_{i_1\dots i_{k-1}i_k}(s,\tau)}{\partial s}\Big|_{s=t}\right| \leq M_0\frac{(M_0\delta)^{k-1}}{(k-1)!} + M_0\int_t^\tau \max_{\tau\in[t,\tau]}\left|\frac{\partial E_{i_1\dots i_{k-1}}(s,\tau)}{\partial s}\Big|_{s=t}\right| d\tau. \tag{37}$$

Denote the left-hand side above as $\bar{E}_k(\delta)$. Then

$$\bar{E}_k(\delta) \leq M_0\frac{(M_0\delta)^{k-1}}{(k-1)!} + M_0\int_0^\delta \bar{E}_{k-1}(\delta')d\delta'. \tag{38}$$

It is not hard to verify that $\bar{E}_k(\delta) = G_k M_0^k \delta^{k-1}$ for some G_k dependent only on k. Then G_k should satisfy $G_k \leq 1/(k-1)! + G_{k-1}/(k-1)$. By induction, $G_k \leq k/(k-1)!$ for all $k \geq 1$. Hence,

$$\max_{\tau\in[t,t+\delta]}\left|\frac{\partial E_{i_1\dots i_{k-1}i_k}(s,\tau)}{\partial s}\Big|_{s=t}\right| \leq \frac{k}{(k-1)!}M_0^k\delta^{k-1}. \tag{39}$$

Substituting the above bound into (35), we have

$$\left|\frac{d}{dt}E_\mu(t, t+\delta)\right| \leq \frac{k+1}{(k-1)!}M_0(M_0\delta)^{k-1}. \tag{40}$$

Now we consider the variation of $\theta_j^*(t)$. Since $\theta_j^*(t) = \Phi^{-1}(t)\psi_j(t)$, with the definition of $\Phi(t)$ and $\psi_j(t)$, we have

$$\theta_j^*(t) = \left[L_\mu h_j(z(t))\right]_{\mu\in\mathbb{I}_m^{\le K}} + \int_0^\Delta \phi(t,\delta)\left[\sum_{\mu\in\mathbb{I}_m^{>K}} L_\mu h_j(z(t))E_\mu(t,t+\delta)\right]d\delta, \tag{41}$$

in which $\mathbb{I}_m^{>K} := \cup_{k>K}\mathbb{I}_m^k$. Differentiating over t, we should have

$$\begin{aligned}\dot\theta_j^*(t) =& \left[\frac{\partial L_\mu h_j}{\partial z}(z(t))\dot z(t)\right]_{\mu\in\mathbb{I}_m^{\le K}} + \Phi^{-1}(t)\int_0^\Delta \dot\phi(t,\delta)\left[\sum_{\mu\in\mathbb{I}_m^{>K}} L_\mu h_j(z(t))E_\mu(t,t+\delta)\right]d\delta \\ &+ \Phi^{-1}(t)\int_0^\Delta \phi(t,\delta)\sum_{\mu\in\mathbb{I}_m^{>K}}\left[\frac{\partial L_\mu h_j}{\partial z}(z(t))\dot z(t)E_\mu(t,t+\delta) + L_\mu h_j(z(t))\frac{dE_\mu(t,t+\delta)}{dt}\right]d\delta \\ =:& \omega_1+\omega_2+\omega_3.\end{aligned} \tag{42}$$

Among the three terms, the first term ω_1, due to the compactness of $\bar X$, is bounded in norm by $\Omega_1\|\dot z(t)\|$ for some constant $\Omega_1 > 0$. For the second term ω_2, since the brackets in the integrand, as the truncation reminder of the Chen-Fliess series, is bounded in absolute value by $r(K,\Delta)$, is bounded by

$$\|\omega_2\| \le \frac{r(K,\Delta)}{\alpha_\Delta}\int_0^\Delta \|\dot\phi(t,\delta)\|d\delta. \tag{43}$$

This bound can be further relaxed by the norm of each component of $\dot\phi(t,\delta)$, namely $(d/dt)E_\mu(t,t+\delta)$:

$$\begin{aligned}\|\omega_2\| &\le \frac{r(K,\Delta)}{\alpha_\Delta}\sum_{\mu\in\mathbb{I}_m^{\le K}}\int_0^\Delta \left|\frac{d}{dt}E_\mu(t,t+\delta)\right|d\delta \le \frac{r(K,\Delta)}{\alpha_\Delta}\sum_{k=0}^K (m+1)^k\int_0^\Delta \frac{k+1}{k!}M^k\delta^{k-1}d\delta \\ &\le \frac{r(K,\Delta)}{\alpha_\Delta}\sum_{k=0}^K \frac{k+1}{k!}(M_0\Delta(m+1))^k\end{aligned} \tag{44}$$

Given that $\Delta < \bar\Delta = 1/(m+1)M_0M_2$ and $M_2 \ge 1$,

$$\|\omega_2\| \le \frac{r(K,\Delta)}{\alpha_\Delta}\sum_{k=0}^\infty \frac{k+1}{k!} = 2e\frac{r(K,\Delta)}{\alpha_\Delta} = \text{const.} \tag{45}$$

For ω_3, the bounding of the term in brackets is similar to the procedure for ω_2, leading to a linear term in $\|\dot{z}(t)\|$ and a constant term. The derivation is redundant and omitted here. It then follows from the notation of $\Phi^{-1}(t)$ and Cauchy-Schwarz inequality that $\|\omega_3\|$ can be obtained by multiplying $\alpha_\Delta^{-1/2}$. Here we require that $\Delta < \bar{\Delta}/2$. This factor 2 originates from recursively bounding the gradient of the Lie derivatives $\partial L_\mu h_j / \partial z$, where the Leibniz rule results in two terms. Combining the three parts: ω_1, ω_2, and ω_3, we reach at the conclusion of the proposition.

Proposition 3 *If the state equation in the observer dynamics is a linear one (as in KKL observers): $\dot{z} = Az + By$, with (A, B) controllable and $A + \rho I$ Hurwitz for a scalar $\rho > 0$, then there exists a constant $C_5 > 0$ such that for any $t > 0$,*

$$\int_0^t \|\dot{z}(\tau)\|^2 d\tau \leq \int_0^t C_5 \|\dot{y}(\tau)\|^2 d\tau. \tag{46}$$

The proof of this proposition follows from linear systems theory. The constant in the statement is simply the H_∞-norm of the dynamical system mapping from y to z. Following this proposition, since the plant output y is given by a smooth mapping of the states, $y = H(x)$, when $\bar{\mathcal{X}}$ is compact, $\|\dot{y}\|$ is bound by $C_6\|\dot{x}\|$ with some $C_6 > 0$. Finally, we note that the bound derived is for the integrated squared error of θ_j instead of for x_j. However, the state observation error is simply the error in the first element of θ_j, which implies $\|\hat{x}_j - \hat{x}_j\| \leq \|\theta_j - \theta_j^*\|$. Therefore, summarizing the above discussions, we reach at the following main theorem.

Theorem 1 *Suppose that $\bar{\mathcal{X}}$ is compact, contained in a open set $O \subseteq \mathbb{R}^n$ such that the system* (1) *is (O, ϵ)-backward distinguishable for some $\epsilon > 0$. Also suppose that the output trajectory satisfies the persistent excitation condition (Assumption 1) using the Chen-Fliess basis functions with multi-indices truncated to length K and a sufficiently small time window length $\Delta < \bar{\Delta}/2 = 1/2(m+1)M_0M_2$. Then there exists positive constants C, C', such that for any $t \geq 0$,*

$$\int_0^t \|\hat{x}(\tau) - x(\tau)\|^2 d\tau \leq C \int_0^t \|\dot{x}(\tau)\|^2 d\tau + C'. \tag{47}$$

3.3 | Performance under the Boundedness Assumption

The premise for the afore-proved performance guarantee of the online learning of Chen-Fliess series is the persistent excitation assumption on the output signal y (Theorem 1).

However, the assumption is difficult to verify or be well satisfied (i.e., if the matrix $\Phi(t)$ is not well-conditioned, then the α_Δ can be too small to yield a tight bound). Hence, it is desirable to establish the performance guarantee without the persistent excitation condition.

The following conclusions are derived in a similar manner to the Theorem 1 of Zhang, Lu, and Zhou.[41] Let us denote by $\theta_j^\circ(t)$ the corresponding true values of the Lie derivatives, $[L_\mu h_j(z(t))]_{\mu\in\mathbb{I}_m^{\le K}}$. Hence, at any time t, accoring to the convergence property of the Chen-Fliess series, $|\hat{x}_j(t+\delta) - \theta_j^\circ(t)\phi(t,\delta)| \le r(K,\delta)$. Thus, $J(\theta_j^\circ(t),t) \le \frac{1}{2}\int_0^\Delta r(K,\delta)^2 d\delta =: \bar{r}(K,\Delta)$. It is thus only needed to examine the gap between $J(\theta_j(t),t)$ and $J(\theta_j^\circ(t),t)$.

Proposition 4 *Suppose that $\bar{X}$ is bounded, which implies that $\theta_j^\circ(t)$ is bounded. Further assume that the trajectory of the estimate $\theta_j(t)$ is bounded. Then by letting $D > 0$ be an upper bound on $\|\theta_j(t)\|$, $\|\theta_j^\circ(t)\|$, and $\|\theta_j(0) - \theta_j^\circ(0)\|$, we have for any $t > 0$,*

$$\int_0^t \left[J(\theta_j(\tau),\tau) - J(\theta_j^\circ(\tau),\tau)\right] d\tau \le \frac{7D^2}{4\eta} + \frac{D}{\eta}\int_0^t \|\dot{\theta}_j^\circ(\tau)\| d\tau, \tag{48}$$

Proof Since $\dot{\theta}_j(t) = -\eta\nabla J(\theta_j(t),t)$, due to the convexity of J in the first argument,

$$\begin{aligned} J(\theta_j(t),t) - J(\theta_j^\circ(t),t) &\le \nabla J(\theta_j(t),t)^\top(\theta_j(t) - \theta_j^\circ(t)) = -\frac{1}{\eta}\dot{\theta}_j(t)^\top(\theta_j(t) - \theta_j^\circ(t)) \\ &= -\frac{1}{2\eta}\frac{d}{dt}\|\theta_j(t)\|^2 + \frac{1}{\eta}\frac{d}{dt}(\theta_j(t)^\top\theta_j^\circ(t)) - \frac{1}{\eta}\theta_j(t)^\top\dot{\theta}_j^\circ(t). \end{aligned} \tag{49}$$

Integrating on $[0,t]$, we have

$$\begin{aligned} \int_0^t \left[J(\theta_j(\tau),\tau) - J(\theta_j^\circ(\tau),\tau)\right] d\tau \le& \frac{1}{2\eta}\left(\|\theta_j(0)\|^2 - \|\theta_j(t)\|^2\right) \\ &+ \frac{1}{\eta}\left(\theta_j(t)^\top\theta_j^\circ(t) - \theta_j(0)^\top\theta_j^\circ(0)\right) - \frac{1}{\eta}\int_0^t \theta_j(\tau)\dot{\theta}_j^\circ(\tau) d\tau. \end{aligned} \tag{50}$$

Since $\|\theta_j(t)\|, \|\theta_j^\circ(t)\| \le D$, $\|\theta_j(0)\|^2 \le D^2$, $\theta_j(t)^\top\theta_j^\circ(t) \le D^2$. Also, $\theta_j(0)^\top\theta_j^\circ(0) \le \|\theta_j(0) - \theta_j^\circ(0)\|/4 = D^2/4$. Hence,

$$\int_0^t \left[J(\theta_j(\tau),\tau) - J(\theta_j^\circ(\tau),\tau)\right] d\tau \le \frac{7D^2}{4\eta} - \frac{1}{\eta}\int_0^t \theta_j(\tau)\dot{\theta}_j^\circ(\tau) d\tau. \tag{51}$$

The theorem is thus proved by further relaxing the last term above.

Due to the strong convexity of J in the first component of the θ_j argument (namely

$x_j(\tau)$), the integrand on the left-hand side upper bounds the difference between $x_j(\tau)$ and $\hat{x}_j^\circ(\tau)$ (multiplied by a constant). Hence, following the above proposition, given a sufficiently long time t, the average observation error $\frac{1}{t}\|x_j(\tau) - \hat{x}_j(\tau)\|d\tau$ is affinely bounded by the total variation of the Lie derivatives: $\frac{1}{t}\int_0^t \|\dot{\theta}_j(\tau)\|d\tau$. Due to the boundedness of $\bar{\mathcal{X}}$, this is further linearly bounded by the average variation of the states. Thus, we have the following theorem

Theorem 2 *Suppose that $\bar{\mathcal{X}}$ is compact, contained in a open set $O \subseteq \mathbb{R}^n$ such that the system* (1) *is (O, ϵ)-backward distinguishable for some $\epsilon > 0$. Using the Chen-Fliess basis functions with multi-indices truncated to length K and a sufficiently small time window length $\Delta < \bar{\Delta}/2 = 1/2(m+1)M_0M_2$, there exist constants $C, C' > 0$ such that for sufficiently large $t \geq 0$,*

$$\int_0^t \|\hat{x}(\tau) - x(\tau)\|d\tau \leq C\int_0^t \|\dot{x}(\tau)\|d\tau + C'. \tag{52}$$

Remark 3 *The conclusions of the two theorems are similar, with the difference in formality that Theorem 1 bounds the average squared observation error by the average squared variation of states:*

$$\frac{1}{t}\int_0^t \|\hat{x}(\tau) - x(\tau)\|^2 d\tau \sim O\left(\frac{1}{t}\int_0^t \|\dot{x}(\tau)\|^2 d\tau\right) \tag{53}$$

while Theorem 2 bounds the average observation error by the average variation of states:

$$\frac{1}{t}\int_0^t \|\hat{x}(\tau) - x(\tau)\|d\tau \sim O\left(\frac{1}{t}\int_0^t \|\dot{x}(\tau)\|d\tau\right). \tag{54}$$

Remark 4 *The form of the state observer:*

$$\dot{\theta}_j(t) = -\eta\left(\Phi(t)\theta_j(t) - \psi_j(t)\right). \tag{55}$$

is a dynamical system involving the regressor $\Phi(t)$ and the vector $\psi_j(t)$, both defined as an integral within a time window $[t, t+\Delta]$. The integrals can be approximated numerically through collocation:

$$\Phi(t) \approx \Delta\sum_{p=1}^{P} w_p\phi(t,\delta_p)\phi(t,\delta_p)^\top, \quad \psi_j(t) \approx \Delta\sum_{p=1}^{P} w_p\phi(t,\delta_p)x_j(t+\delta_p), \tag{56}$$

for some $\delta_p \in [0, \Delta]$, $p = 1, \ldots, P$ and $\sum_{p=1}^{P} w_p = 1$. For the computation of the basis $\phi(t, \delta_p)$,

whose components are defined as recursive integrals $E_\mu(t, t+\delta_p)$ for $\mu \in \mathbb{I}_m^{\leq K}$, a second layer of collocation need to be used.

3.4 | Performance of Dimensionality Reduction

Finally when the states are not measured and KPCA is needed to reduce the dimensionality from z to principal components $\pi_1, \ldots, \pi_n$, the performance of this KPCA must be warranted. This is enabled by (i) bounding the error of online KPCA with respect to the offline KPCA, and (ii) bounding the error of offline KPCA with respect to the true subspaces underlying the data. For the first point, we have the following fact from Honeine.[29]

Fact 4 *Using online KPCA, when m points have been included in the dataset, for each principal component (as a function of z) $\pi_i(\cdot)$, $i = 1, \ldots, n$:*

$$\|\pi_i(\cdot) - \pi_i^*(\cdot)\|_{\mathcal{H}}^2 \leq \nu/\varrho_i. \tag{57}$$

In the above inequality, ν is the hyperparameter used in the online KPCA algorithm controlling the entry of new data. λ_i is the i-th largest eigenvalue of the kernel matrix G divided by m. Here $\pi_i = \sum_{j=1}^N \alpha_{ij}\kappa(\cdot, z^{(j)})$ is based on the normalization $\sum_{j=1}^N \alpha_{ij}^2 = 1/N\varrho_i$. π_i^* is the principal component that would have been obtained if these N points are treated with offline KPCA. The norm is that of the reproducing kernel Hilbert space $\mathcal{H}$, i.e., the linear space spanned by the Gaussian kernel functions $\{\kappa(\cdot, z^{(j)}) | j = 1, \ldots, N\}$ endowed with the inner product

$$\langle \kappa\left(\cdot, z^{(j_1)}\right), \kappa\left(\cdot, z^{(j_2)}\right)\rangle_{\mathcal{H}} := \kappa\left(z^{(j_1)}, z^{(j_2)}\right), \quad j_1, j_2 = 1, \ldots, N. \tag{58}$$

Denote P as the mapping from any data point z to its principal components under online KPCA, and P^* the corresponding mapping if offline KPCA has been used. The above conclusion implies that $\|P - P^*\| \leq \nu \sum_{i=1}^n \varrho_i^{-1}$. For simplification, we may assume that the eigenvalues can be bounded by two geometric sequences.

Assumption 2 *The n largest eigenvalues $\varrho_1, \ldots, \varrho_n$ of the kernel matrix G satisfy the inequalities:*

$$\underline{q} i^{-\gamma} \leq \varrho_i \leq \overline{q} i^{-\gamma}, \quad i = 1, \ldots, n \tag{59}$$

for some $0 < \underline{q} \leq \overline{q}$ and $\gamma > 1$.

Under this assumption, we further obtain $\|P - P^*\| \leq \nu n^{\gamma+1}/\underline{q}(\gamma+1)$. Then, let $\mathcal{S}$ be the sub-

space specified by the online KPCA and $\mathcal{S}^*$ be its offline counterpart. Define the distance between subspaces:

$$d(\mathcal{S}, \mathcal{S}^*) := \mathbb{E}_z \left\| (\mathrm{proj}_{\mathcal{S}} - \mathrm{proj}_{\mathcal{S}^*}) \varphi(z) \right\|^2 \tag{60}$$

where $\varphi(z)$ is the feature vector of z that correspond to the kernel (i.e., $\kappa(z, z') = \varphi(z)^\top \varphi(z')$), and the expectation $\mathbb{E}_z$ over z is evaluated under the true distribution of the sample. Under this definition, we have

$$d(\mathcal{S}, \mathcal{S}^*) \le \nu n^{\gamma+1} / \underline{q}(\gamma + 1) =: \xi_1(\gamma)\nu. \tag{61}$$

For the characterization of the distance between the offline KCPA and the actual manifold, the complexity theory of subspace learning gives the following conclusion.[42]

Fact 5 *Under Assumption 1, there is a constant $\xi_2 = \xi_2(\gamma) > 0$ such that if the sample points are independently drawn from the underlying distribution, then with probability at least $1 - \vartheta$,*

$$d(\mathcal{S}^*, \mathcal{S}^\circ) \le \xi_2(\gamma) \left(\frac{\ln(N/\vartheta)}{N} \right)^{(\gamma-1)/2\gamma}. \tag{62}$$

Combining (61) and (62), we have

$$d(\mathcal{S}, \mathcal{S}^\circ) \le \xi_1(\gamma)\nu + \xi_2(\gamma) \left(\frac{\ln(N/\vartheta)}{N} \right)^{(\gamma-1)/2\gamma}. \tag{63}$$

Thus, when ν is small and N is large, the manifold recovered by online KPCA is close to the true manifold where the samples $z(\cdot)$ reside.

Remark 5 *The number of sample points N, under the online KPCA, in fact depends on the kernel bandwidth σ and the threshold on new data entry ν. As we discussed in Subsection 2.3, N is inversely proportional to the minimum distance between the sample points, which is given by $-2\sigma^2 \ln(1-\nu/2)$. Thus, smaller σ and ν result in generalization error. On the other hand, as σ and ν become small, (i) it takes more time for online KPCA to obtain sufficient data points for accurate projections, (ii) the principal components π_i, $i = 1, \ldots, n$ as functions of z have steeper gradients and thus more sensitive to potentially existing uncertainties, and (iii) the principal components have larger variations and thus are harder for the online least squares to track. Therefore, these two hyperparameters (especially σ) should be well tuned.*

4 | CASE STUDIES

4.1 | Oscillator Dynamics and Online Chen-Fliess Series Estimation

To demonstrate the proposed approach, let us consider a planar dynamical system with limit cycle behavior:

$$\dot{x}_1 = 1 + x_1^2 x_2 - 4x_1, \quad \dot{x}_2 = 3x_1 - x_1^2 x_2 \tag{64}$$

This is the "Brusselator" model by Glansdorff and Prigogine that describes the kinetics of an auto-catalytic reaction system.[43] Suppose that a single output $y = x_1 + x_2$ is measured. The initial condition is set as $x(0) = [1, 1]^\top$. For all the ODE simulations, a sampling interval of 0.02 is chosen, and simple Euler forward difference is used. For the state observation under the online least-squares estimation, we use a window Δ that is 10 times the sampling interval. The Chen-Fliess series is truncated after multi-indices of length 3. All the recursive integrals involved are approximated by the trapezoid formula. The state trajectories, along with the online state estimates, are shown in Fig. 2. It is seen from here that the estimates $\hat{x}_1$, $\hat{x}_2$ track the actual states x_1, x_2 with reasonable accuracy. The discrepancies are more significant at the times when the states undergo rapid changes (e.g., at the sharp peaks of x_1 and sharp valleys of x_2). This is in accordance with the online optimization theory that the performance is relate to the variation of states.

Here, the observer update rate, or the online learning rate, η, is determined through tuning, so that the online optimization is adaptive enough while not so radical as to result in instability (which occurs when $\eta \geq 10$ approximately). To empirically determine η, the effect on the resulting average observation error $\overline{\|e\|} := \sqrt{\int_0^{t_{\max}} \|\hat{x} - x\|^2 dt / t_{\max}}$ is plotted (see Fig. 3). Here $t_{\max}$ stands for the simulation duration. We choose $\eta = 1$ for this system, when the average state observation error has become small and further reduction is slow, and fix this value in the following discussion in the absence of actual state measurements.

4.2 | Oscillator Dynamics and Dimensionality Reduction

Next, let us examine the case when the states are not measured and KPCA is adopted. To immerse the states into $\mathbb{R}^3$, we assign the KKL linear dynamics with poles 0.5, 1.0, and 2.0. For online KPCA, we let the update rate be $\lambda(t) = \lambda_0/(1 + t/\tau_\lambda) = 1/(1 + t)$ based on the intuition that the update should match the inherent dynamics of the process. By varying ν between 0.01 and 0.1, the generated results are found to be similar. Therefore, we fix

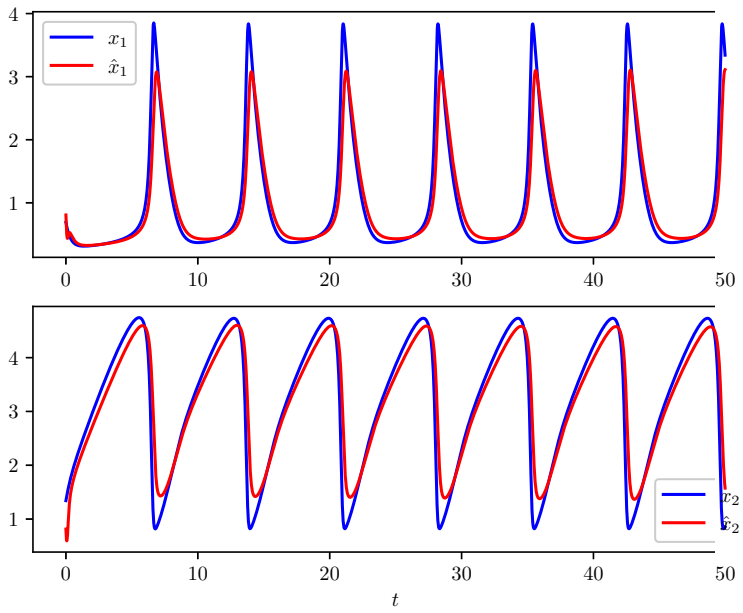


FIGURE 2 Online least squares for the Chen-Fliess series for state observation in the oscillator system ($\eta = 1$).

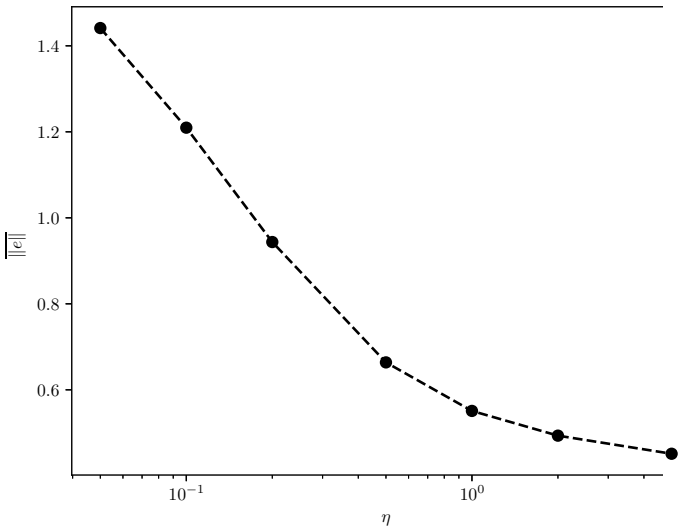
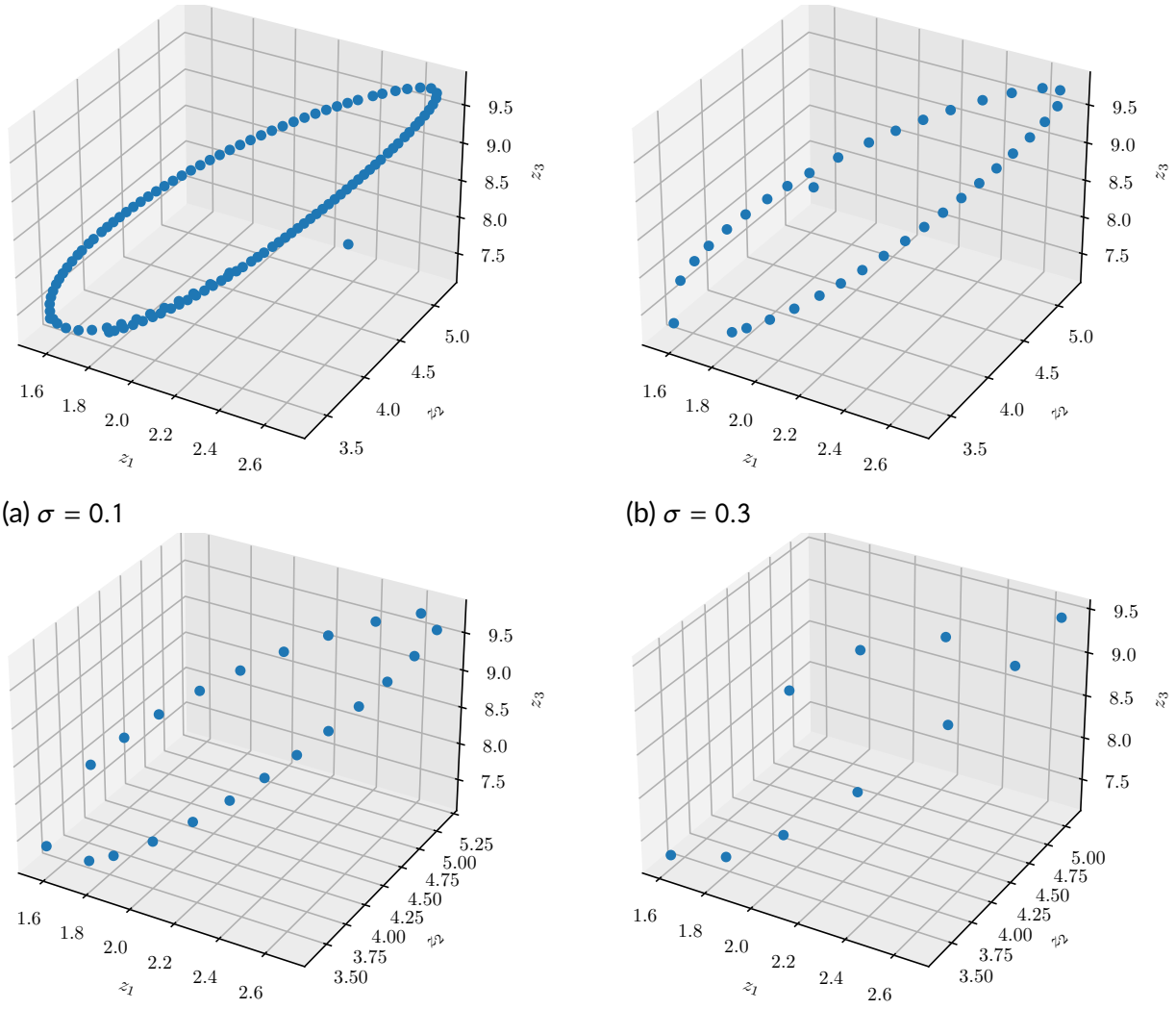


FIGURE 3 Dependence of average observation error on the observer rate for the oscillator system.

$\nu = 0.05$ and tune the kernel bandwidth σ as the key hyperparameter in online KPCA. Four different choices of σ are considered – 0.1, 0.3, 0.5, and 1. The data points obtained during the execution of online KPCA are shown in Fig. 4.

Due to the oscillatory dynamics, most of the data points in the 3-dimensional z-space



(a) $\sigma = 0.1$

(b) $\sigma = 0.3$

(c) $\sigma = 0.5$

(d) $\sigma = 1.0$

FIGURE 4 Datasets gathered during online KPCA under different kernel bandwidths for the oscillator system ($\lambda_0 = 1$, $\nu = 0.05$).

collected during online KPCA lie around a corresponding periodic orbit. We initialize the online KPCA not at $t = 0$ but at $t = 10$, when the dynamics has settled on the limit cycle. At the initialization of the online KPCA, we randomly choose $n = 2$ points on the trajectory during $t = 5$ and $t = 10$ as the initial data points, and assign the principal vectors as $\alpha = I_n$. This is intended to avoid initializing α with singular directions that would hinder the update (e.g., if only one data point is used, then $\alpha_2 = [0, 0]$ and the second principal component will remain 0 without any update; if two very close data points are used, then the two principal components are difficult to be orthogonalized). This random initialization explains the outliers points on the plots in 4. Based on these plots, $\sigma = 0.5$ is selected.

Under the online KPCA, the transformed states $z(t) \in \mathbb{R}^3$ are reduced to a 2-dimensional

trajectory of principal components $\pi(t)$. To see that the $\pi(t)$ can be viewed as equivalents of the underlying states through a diffeomorphism, the phase portraits of $x(t)$, $z(t)$, and $\pi(t)$ are shown in Fig. 5. It can be seen that the trajectories of $\pi(t)$ is topologically similar to that of $x(t)$, with the main difference in the scaling of the two coordinates. On the other hand, the bottom left part of the limit cycle, approximately the curve segment from $(0.5, 3.5)$ to $(3.5, 1.4)$, are mapped to the plot of principal components with a distortion into a protruding part. That is to say, although $x \to \pi$ is still diffeomorphic, the corresponding mapping appears be much steeper in this region. This corresponds to the stage when x_1 rapidly increases and x_2 rapidly decreases, which occurs for a time duration of only about 0.2 within every period of 7.2 time units. With these, we claim that the principal components can well reconstruct the unmeasured states in the majority of times when the states vary smoothly.

Finally, an online least-squares approach is used to track the principal components with a recursively updated Chen-Fliess series. The resulting trajectories are shown in Fig. 6, indicating a good performance.

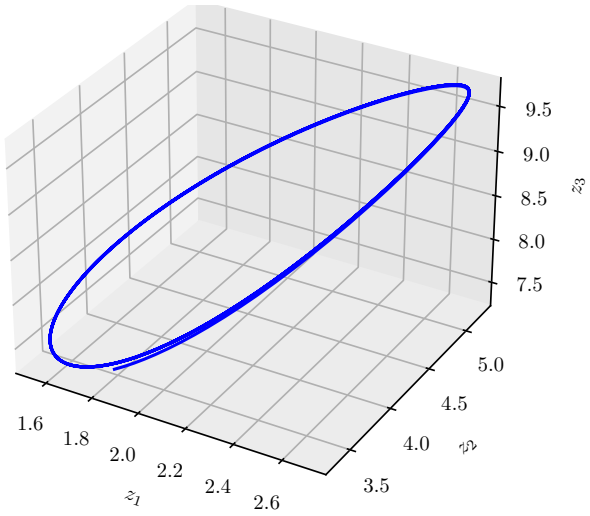
4.3 | Chaotic Dynamics

As a more complex example, we further examine a chaotic Lorenz dynamics, which can also be viewed as the kinetics of a reactive system:[44]

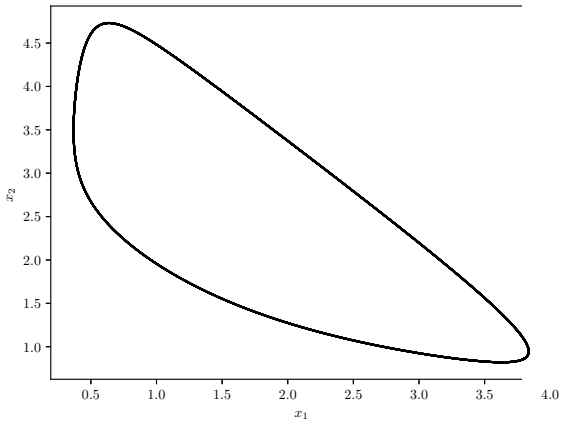
$$\dot{x}_1 = 10(x_2 - x_1), \quad \dot{x}_2 = x_1(28 - 10x_3) - x_2, \quad \dot{x}_3 = 10x_1x_2 - (8/3)x_3. \tag{65}$$

The measured output is $y = x_2$. The initial condition is set as $x(0) = [1, 1, 1]^\top$. The sampling time for simulation is 0.01. For the case when states are measurable, we still first consider the effect of observer update rate on the average observation error $\overline{\|e\|}$. By varying η from 0.01 to 10, it is found that $\overline{\|e\|}$ declines as η increases (see Fig. 7). Empirically we set $\eta = 5$. Under this tuning, the tracking of states by the online least squares is illustrated in Fig. 8.

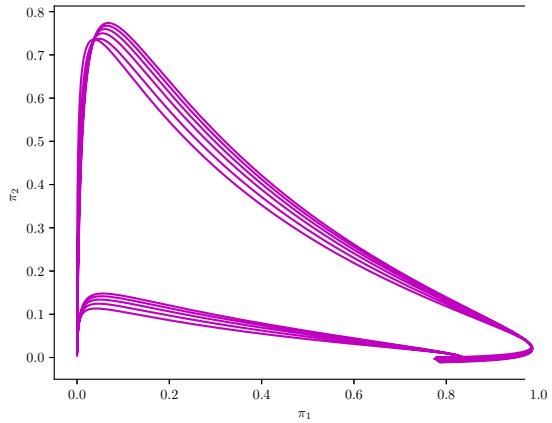
In the absence of state measurements, the selection of the kernel bandwidth σ has a significant impact on the resulting online KPCA. When choosing $\sigma = 0.3$, there exists a long time duration in which the principal components appear to be very close to zero, as newly encountered data in the chaotic dynamics have almost zero kernel function values evaluated at the existing data when the bandwidth is too small. On the other hand, when $\sigma = 0.75$, the a posteriori principal components π_i^* (evaluated based on the offline KPCA)



(a) Transformed states



(b) States



(c) Principal components

FIGURE 5 Phase portraits for the oscillator system ($\nu = 0.05$, $\sigma = 0.5$, $\lambda_0 = 1$).

have large discrepancies with their online counterparts, as the data update is too conservative online. The plotting for the results under these two values are omitted for brevity. A middle ground is chosen to be $\sigma = 0.5$. The plot of a posteriori principal components, online principal components, and online least-squares tracked ones are shown in Fig. 9. At this bandwidth, the ultimately obtained dataset by offline KPCA is shown in Fig. 10.

5 | CONCLUSIONS

In this paper, a data-driven approach has been proposed for state observation in nonlinear systems. While generally the KKL observer is difficult to compute, this work utilizes the affine structure underlying the KKL dynamics and considers the state observation prob-

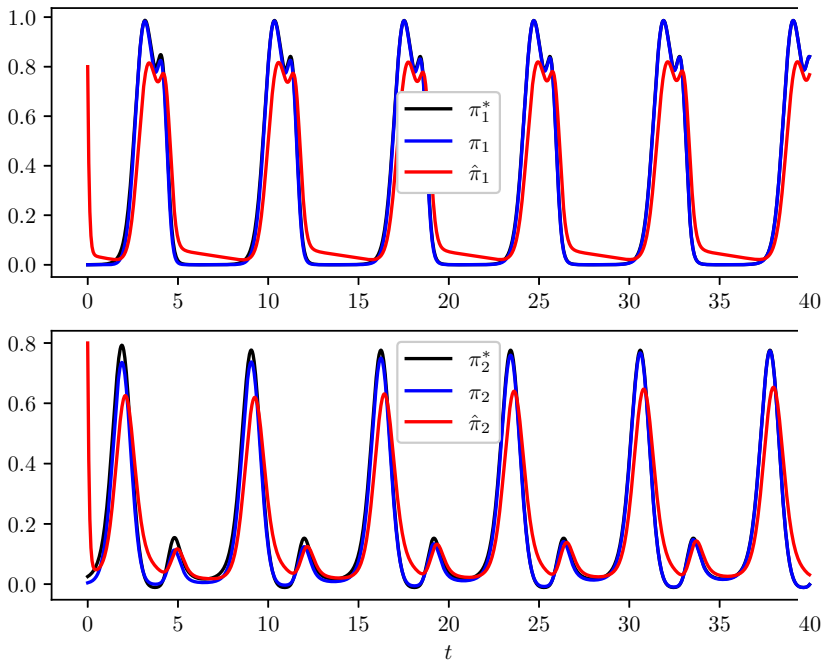


FIGURE 6 Principal components tracked by the Chen-Fliess series under online least-squares for the oscillator system ($\nu = 0.05$, $\sigma = 0.5$, $\lambda_0 = 1$).

lem as a least-squares one for the Chen-Fliess series expansion, thus rendering the problem amenable to an online convex optimization algorithm based on a simple gradient flow. The performance guarantee of this online state observer, under either a persistent excitation condition or a boundedness assumption, can be derived, indicating that the average (squared) observation error is bounded by the average (squared) variation of the states.

When the states are not measurable to make the problem a supervised learning one, a KPCA algorithm as a nonlinear dimensionality reduction routine is proposed to act on a nonlinearly transformed state signal, which is available from the linear dynamics of a KKL observer. Thus, the KPCA-reduced states, which are diffeomorphic to the actual states, can be tracked by the online optimization. In other words, with the help of a KKL observer structure, even when the states are not measured, the problem can be regarded as a self-supervised learning one. The KPCA algorithm implemented here is also an online version, which allows data to be collected in real time and updates the coefficients in principal components recursively. The complexity of the online KPCA is restricted ultimately.

The case studies on a two-dimensional oscillator dynamics (with a limit cycle) and a three-dimensional chaotic dynamics have demonstrated the use of the proposed method. The restriction is, however, that only autonomous systems are considered in this work. For non-autonomous systems involving exogenous inputs, such a data-driven observation can

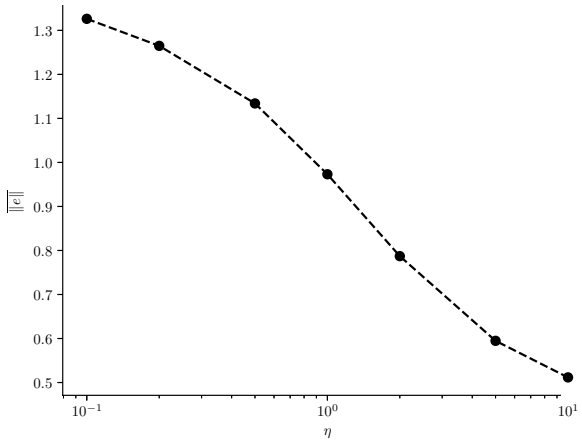


FIGURE 7 Dependence of average observation error on the observer rate for the Lorenz system.

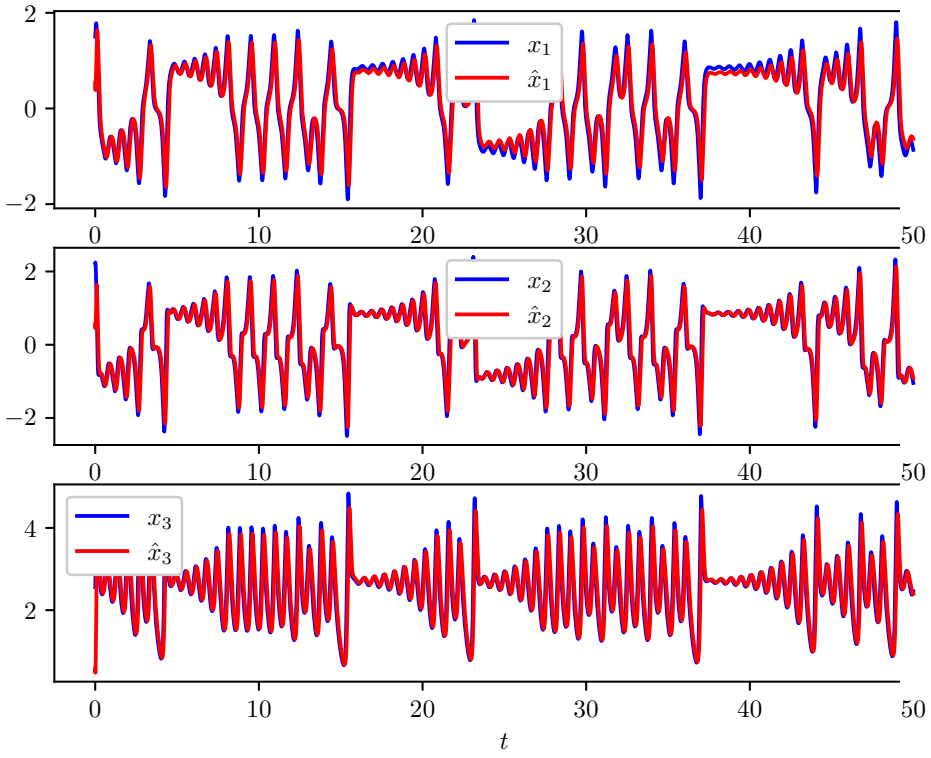


FIGURE 8 Online least squares for the Chen-Fliess series for state observation in the Lorenz system ($\eta = 5$).

be more challenging, since the KKL observer will have a dynamics in the form of $\dot{z} = Az + By + \omega(z)u$ with an unknown term $\omega(z)$ present.[21] This problem, as well as its application in data-driven control approaches will be investigated in future research.

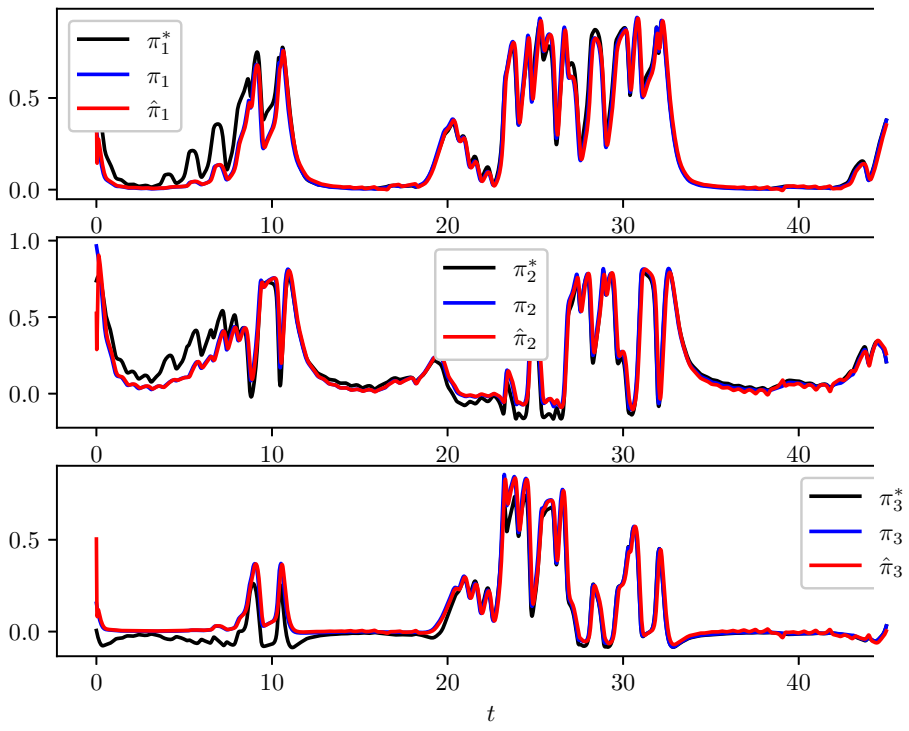


FIGURE 9 Principal components under dimensionality reduction for the Lorenz system ($\eta = 5$, $\nu = 0.05$, $\lambda_0 = 5$, $\sigma = 0.5$).

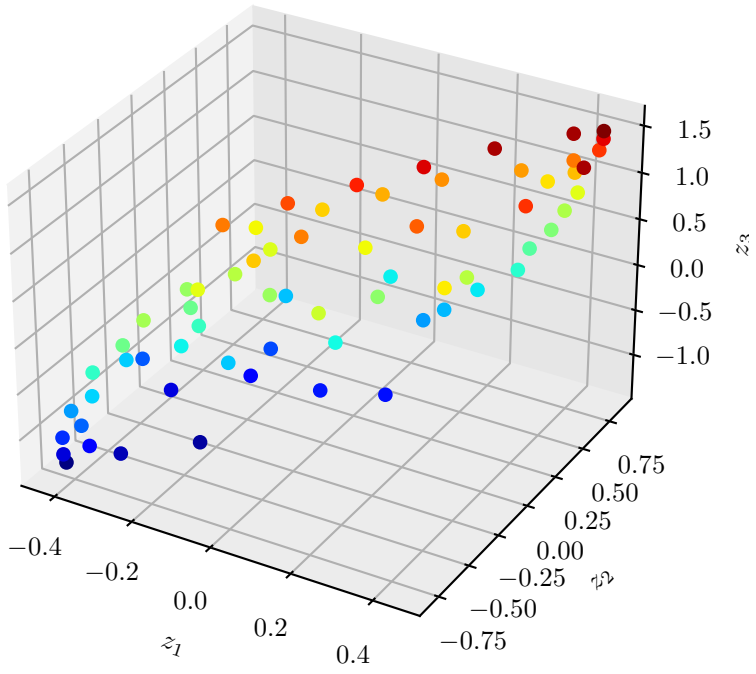


FIGURE 10 Dataset in KPCA for the Lorenz system ($\eta = 5$, $\nu = 0.05$, $\lambda_0 = 5$, $\sigma = 0.5$). The z_4 component is indicated by the marker color under the "jet" color map.

NOMENCLATURE

A, B	Linear dynamics in the KKL observer
$\mathcal{A}_z$	Attractor of observer states
a, b	Index of two initial states
C	Constants
$\mathbb{C}$	Field of complex numbers
D	Diameter
d	Distance
E	Recursive integral
$\mathbb{E}$	Expectation
e	Observer error
F	Process state equation
G	Kernel matrix
g	Observer state equation
H	Process output equation
$\mathcal{H}$	Reproducing kernel Hilbert states
h	Observer output equation
I	Unit matrix
$\mathbb{I}$	Multi-index set
i	Index for process states
id	Identity mapping
Im	Imaginary part
J	Objective function
j	Index for process outputs or data points
K	Maximum multi-index length for truncation
k	Multi-index length; Discrete time
L	Lie derivative
M	Constants
m	Dimension of process outputs
N	Number of data points
n	Dimension of process states
n_z	Dimension of observer states
O	Open set
O	Order of magnitude
P	Projection onto principal coordinates
$\underline{q}, \overline{q}$	Constants in eigenvalue decay
$\mathbb{R}$	Field of real numbers
r	Residual in Chen-Fliess series truncation
Re	Real part
$\mathcal{S}$	Subspace
$\int$	Dummy time variable
T	Immersion transform
t	Time
U	Upper triangular matrix
X	ODE solution
$\mathcal{X}$	Process state space
x	Process states
y	Process outputs
$\mathcal{Z}$	Observer state space
z	Observer states
α	Coefficients in principal vector
α_Δ	Constant in persistent exciation
β	Regression for kernels in online KPCA
γ	Decay rate of eigenvalues
Δ	Time window for Chen-Fliess series
δ	Time duration
ϵ	Small positive real number
ε	Small positive real number
η	Update rate in online least squares
θ	Parameters to be estimated
ϑ	Small positive real number
κ	Kernel function or vector
Λ	Set of eigenvalues of A
λ	Eigenvalues of A; Update rate in online KPCA
μ	Multi-index

ν	Threshold in online KPCA
ξ	Complexity terms
π	Principal components
ρ	Decay rate of observation error
ϱ	Eigenvalues of kernel matrix
σ	Kernel bandwidth
ς	Backward time
τ	Time constant; Dummy time variable
τ^*	Sampling time
Φ	Regressor matrix
ϕ	Basis vector
φ	Feature mapping
ψ	Label vector for regression
ω	Short-hand notation for terms
$\varnothing$	Empty multi-index
∇	Gradient operator
$\sim$	On the order of
$A \succeq (\preceq) B$	$A - B$ is positive (negative) semidefinite
†	Left-inverse mapping
*	Optimal estimate; Offline estimate
$^\circ$	True value
$\bar{\mathcal{X}}$	Closure of set $\mathcal{X}$
$\hat{x}$	Estimation of x
$\dot{x}$	Time derivative of x
$\tilde{x}$	Increment of x

ACKNOWLEDGMENTS

The author appreciates the Department of Chemical and Biomolecular Engineering, NC State University for the startup fund.

AUTHOR CONTRIBUTIONS

Wentao Tang: Conceptualization; formal analysis; funding acquisition; investigation; methodology; project administration; software; writing – original draft, writing – review & editing.

DATA AVAILABILITY STATEMENT

The numerical data from Figures 2, 4, 5, 6, 8, 9, 10, are tabulated in csv format, included in the zip folder in the Supplementary Material, named as "`FigureX.csv`" (`X = 2, ..., 10`) respectively. These figures are created from computer simulations. All the Python codes needed to reproduce the simulations are available at GitHub repository: `https://github.com/WentaoTang-Pack/NonlinearObserver`. The instructions are included in the code comments of "`test.py`". Figures 3 and 7 are plotted from the numerical computations based on computer simulation results. The instructions to generate these two plots, as well as the tabulated data, are included as one pdf file in the Supplementary Material, named as

"Tabulated Data for Figures 3 and 7.pdf".

ORCID

Wentao Tang https://orcid.org/0000-0003-0816-2322

REFERENCES

[1] Ogunnaike B, Ray W. *Process dynamics, modeling, and control*. Oxford University Press. 1994.

[2] Hangos KM, Bokor J, Szederkényi G. *Analysis and control of nonlinear process systems*. Springer. 2004.

[3] Baldea M, Daoutidis P. *Dynamics and nonlinear control of integrated process systems*. Cambridge University Press. 2012.

[4] Rawlings JB, Mayne DQ, Diehl M. *Model predictive control: Theory, computation, and design*. Nob Hill, 2nd ed. 2017.

[5] Kravaris C, Kantor JC. Geometric methods for nonlinear process control. 1. Background. *Ind Eng Chem Res*. 1990;29(12):2295–2310.

[6] Ren YM, Alhajeri MS, Luo J, Chen S, Abdullah F, Wu Z, Christofides PD. A tutorial review of neural network modeling approaches for model predictive control. *Comput Chem Eng*. 2022;165:107956.

[7] Nian R, Liu J, Huang B. A review on reinforcement learning: Introduction and applications in industrial process control. *Comput Chem Eng*. 2020;139:106886.

[8] Korda M, Mezić I. Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control. *Automatica*. 2018;93:149–160.

[9] Narasingam A, Son SH, Kwon JSI. Data-driven feedback stabilisation of nonlinear systems: Koopman-based model predictive control. *Int J Control*. 2023;96(3):770–781.

[10] Hewing L, Kabzan J, Zeilinger MN. Cautious model predictive control using gaussian process regression. *IEEE Trans Control Syst Technol*. 2019;28(6):2736–2743.

[11] Bradford E, Imsland L, Zhang D, del Rio Chanona EA. Stochastic data-driven model predictive control using Gaussian processes. *Comput Chem Eng*. 2020;139:106844.

[12] Brunton SL, Proctor JL, Kutz JN. Discovering governing equations from data by

sparse identification of nonlinear dynamical systems. *Proc Natl Acad Sci USA*. 2016; 113(15):3932–3937.

[13] Lejarza F, Baldea M. Discovering governing equations via moving horizon learning: The case of reacting systems. *AIChE J*. 2022;68(6):e17567.

[14] Tang W, Daoutidis P. Dissipativity learning control (DLC): A framework of input–output data-driven control. *Comput Chem Eng*. 2019;130:106576.

[15] Tang W, Daoutidis P. Dissipativity learning control (DLC): Theoretical foundations of input–output data-driven model-free process control. *Syst Control Lett*. 2021; 147:104831.

[16] Kravaris C, Hahn J, Chu Y. Advances and selected recent developments in state and parameter estimation. *Comput Chem Eng*. 2013;51:111–123.

[17] Luenberger D. Observers for multivariable systems. *IEEE Trans Autom Control*. 1966; 11(2):190–197.

[18] Simon D. *Optimal state estimation: Kalman, H_∞, and nonlinear approaches*. John Wiley & Sons. 2006.

[19] Kazantzis N, Kravaris C. Nonlinear observer design using Lyapunov's auxiliary theorem. *Syst Control Lett*. 1998;34(5):241–247.

[20] Andrieu V, Praly L. On the existence of a Kazantzis-Kravaris/Luenberger observer. *SIAM J Control Optim*. 2006;45(2):432–456.

[21] Bernard P, Andrieu V. Luenberger observers for nonautonomous nonlinear systems. *IEEE Trans Autom Control*. 2018;64(1):270–281.

[22] Brivadis L, Andrieu V, Bernard P, Serres U. Further remarks on KKL observers. *Syst Control Lett*. 2023;172:105429.

[23] Ramos LC, Di Meglio F, Morgenthaler V, da Silva LFF, Bernard P. Numerical design of Luenberger observers for nonlinear systems. In: *59th Conference on Decision and Control (CDC)*. IEEE. 2020; pp. 5435–5442.

[24] Buisson-Fenet M, Bahr L, Di Meglio F. Towards gain tuning for numerical KKL observers. *arXiv preprint*. 2022;arXiv:2204.00318.

[25] Niazi MUB, Cao J, Sun X, Das A, Johansson KH. Learning-based design of Luenberger observers for autonomous nonlinear systems. *arXiv preprint*. 2022;arXiv:2210.01476.

[26] Miao K, Gatsis K. Learning robust state observers using neural ODEs. *arXiv preprint*. 2022;arXiv:2212.00866.

[27] Isidori A. *Nonlinear control systems: An introduction*. Springer. 1985.

[28] Sastry S. *Nonlinear systems: analysis, stability, and control*. Springer. 1999.

[29] Honeine P. Online kernel principal component analysis: A reduced-order model. *IEEE Trans Pattern Anal Mach Intell*. 2011;34(9):1814–1826.

[30] Schölkopf B, Smola A, Müller KR. Kernel principal component analysis. In: *The 7th International Conference on Artificial Neural Networks (ICANN'97)*. 1997; pp. 583–588.

[31] Hyvärinen A, Oja E. Independent component analysis by general nonlinear Hebbian-like learning rules. *Signal Process*. 1998;64(3):301–313.

[32] Kim KI, Franz MO, Schölkopf B. Iterative kernel principal component analysis for image modeling. *IEEE Trans Pattern Anal Mach Intell*. 2005;27(9):1351–1366.

[33] Rogers CA. *Hausdorff measures*. Cambridge University Press. 1998.

[34] Goethals I, Pelckmans K, Suykens JA, De Moor B. Subspace identification of Hammerstein systems using least squares support vector machines. *IEEE Trans Autom Control*. 2005;50(10):1509–1519.

[35] Franz MO, Schölkopf B. A unifying view of Wiener and Volterra theory and polynomial kernel regression. *Neur Comput*. 2006;18(12):3097–3118.

[36] Pillonetto G, Dinuzzo F, Chen T, De Nicolao G, Ljung L. Kernel methods in system identification, machine learning and function estimation: A survey. *Automatica*. 2014;50(3):657–682.

[37] Verdult V, Suykens JA, Boets J, Goethals I, De Moor B. Least squares support vector machines for kernel CCA in nonlinear state-space identification. In: *The 16th International Symposium on Mathematical Theory of Networks and Systems (MTNS)*. IFAC. 2004; pp. 1–11.

[38] Hazan E. Introduction to online convex optimization. *Found Trend Optim*. 2016;2(3-4):157–325.

[39] Zinkevich M. Online convex programming and generalized infinitesimal gradient ascent. In: *The 20th International Conference on Machine Learning*. 2003; pp. 928–936.

[40] Mokhtari A, Shahrampour S, Jadbabaie A, Ribeiro A. Online optimization in dynamic environments: Improved regret rates for strongly convex problems. In: *The 55th Conference on Decision and Control (CDC)*. IEEE. 2016; pp. 7195–7201.

[41] Zhang L, Lu S, Zhou Z. Adaptive online learning in dynamic environments. In: *The 32nd International Conference on Neural Information Processing Systems*. 2018; pp. 1330–1340.

[42] Rudi A, Canas GD, Rosasco L. On the sample complexity of subspace learning. In: *The 26th International Conference on Neural Information Processing Systems*. 2013; p. N/A.

[43] Glansdorff P, Prigogine I. *Thermodynamic theory of structure, stability and fluctuations*. John Wiley & Sons. 1971.

[44] Samardzija N, Greller LD, Wasserman E. Nonlinear chemical kinetic schemes derived from mechanical and electrical dynamical systems. *J Chem Phys*. 1989;90(4):2296–2304.