Proceedings of Machine Learning Research vol 242:1–12, 2024

Deep model-free KKL observer: A switching approach

Johan Peralez JOHAN.PERALEZ@INSA-LYON.FR
Univ Lyon, INSA Lyon, Inria, CITI, 69621 Villeurbanne, France

Madiha Nadri MADIHA.NADRI-WOLF@UNIV-LYON1.FR
Univ Lyon, Univ Claude Bernard Lyon 1, LAGEPP, 69621 Villeurbanne, France

Abstract

This paper presents a new model-free methodology to learn Kazantzis-Kravaris-Luenberger (KKL) observers for nonlinear systems. We address three major difficulties arising in observer design: the peaking phenomenon, the noise sensitivity and the trade-off between convergence speed and robustness. We formulate the learning objective as an optimization problem, strictly minimizing the error of the observer estimates, without the need of adding explicit constraints or regularization terms. We further improve the performance with a switching approach, efficiently transitioning between two observers, respectively designed for the transient phase and the asymptotic convergence. Numerical results on the Van der Pol system, the Rössler attractor and on a bioreactor illustrate the gain of the method regarding the literature, in term of performance and robustness.

Code available online: `https://github.com/jolindien-git/DeepKKL`

Keywords: observer, nonlinear systems, deep learning, KKL

1. Introduction

The significance of state estimation algorithms (observer design), and their applications in control system design, fault detection, and various domains, stems from the fact that in most practical applications, including robotics, chemical engineering etc., access to the complete state is often unavailable. Instead, we rely on indirect or partial measurements obtained from sensors. Despite a growing body of literature, the design of observers for nonlinear systems remains an open problem, as no universal methods have yet been established (Bernard et al., 2022).

In this context, the extended Luenberger (Kazantzis-Kravaris-Luenberger, "KKL") theory emerges as a promising approach that establishes the existence of an observer for a large class of systems. This method relies on immersing nonlinear systems into a latent linear system of higher dimension with an output injection. While numerous papers are dedicated to KKL theory, addressing both discrete-time (Kazantzis and Kravaris, 2001; Brivadis et al., 2019) and continuous-time (Andrieu and Praly, 2006; Bernard and Andrieu, 2018) scenarios, practical implementations of such observers remain limited. The challenge lies in identifying the transformation of the coordinates, which is particularly difficult because obtaining an explicit representation is non-trivial, except in certain special cases.

As a consequence, this task has been recently addressed with machine learning, by approximating the change of coordinates. This involves encoding the original state into a high-dimensional latent space using neural networks. In recent years, several approaches have been explored, such as supervised learning (Ramos et al., 2020), physics-informed neural networks (Peralez and Nadri, 2021; Niazi et al., 2023), or neural ODEs (Miao and Gatsis, 2023).

However, maintaining the desired performance of the observer in presence of measurement noise and disturbances, particularly outliers, persists as a significant challenge. In this context, we introduce a flexible observer design methodology based on multi-observer concepts that can be used to address various trade-offs between robustness to measurement noise and convergence speed. Consequently, we propose improvements at three levels. First, we formulate the learning objective as an optimization problem, aiming to minimize the estimation error. Unlike previous efforts, which also aimed to optimize the latent dynamics (Buisson-Fenet et al., 2023; Miao and Gatsis, 2023), we enhance the adaptability of linear dynamics to incorporate arbitrary complex eigenvalues, thereby augmenting overall performance. Next, we improve performance through the deployment of a multi-observer, designed to optimize both transient phase response and asymptotic convergence. Finally, the proposed strategy is shown to effectively address the peaking phenomena through a novel extension of the mappings.

2. Problem statement and background

Let us consider nonlinear dynamical systems under the following general form

$$\begin{cases} \dot{x} &= f^c(x, u) \\ y &= h(x) + w \end{cases} \tag{1}$$

where $x \in \mathbb{R}^{n_x}$ is the state we want to estimate, unknown at test time but supposed to be known during training, $y \in \mathbb{R}^{d_y}$ the output corrupted by noise w, and $u \in \mathbb{R}^{d_u}$ a known control input. We consider the model-free (data-driven) case, where the functions f^c and h are unknown and where inputs and outputs are sampled at the same time-steps t_k. The training data is supposed to be available as batches of output trajectories $Y_{1:K}$ and associated state trajectories $X_{1:K}$.

2.1. Estimation problem

The observer task is to provide an online estimate $\hat{x}$ of the state value given the (past) measurements y, such that $\hat{x}$ converges towards x, or towards a neighborhood of x in presence of noise. While we want to achieve good robustness against measurement noise, *our aim is to minimize the estimation error*, i.e. to minimize $||\hat{x} - x||$.

In a concrete implementation and from the observer viewpoint, system (1) is equivalently represented by the following discrete-time system:

$$\begin{cases} x_{k+1} &= f(x_k, u_k) \\ y_k &= h(x_k) + w_k \end{cases} \tag{2}$$

where f is equivalent to f^c, i.e. such that $f(x, u) \triangleq x + \int_0^{\delta_t} f^c(x, u)dt$, resulting in the same values for x and y at the sampling-times t_k. We argue for the benefits of the latter representation for two reasons, i) potential improvement in accuracy by using an observer designed directly on the discrete-time representation (Brivadis et al., 2019) ii) expressing the update equation instead of the derivative bears similarities with recurrent neural networks (Cho et al., 2014; Hochreiter and Schmidhuber, 1997) and the more recently developed state-space models (Gu et al., 2021).

Notations We denote by $X_k(x_0)$ the value at time k of the solution of system (2) evolving in the compact set $\mathcal{X}$ when initialized at $x_0 \in \mathcal{X}_0$, and $Y_k(x_0, w_k)$ the corresponding output. A temporal sequence X_k for $k \in [\![1, K]\!]$ is denoted $X_{1:K}$.

2.2. KKL observer theory

The KKL concept as introduced by Kazantzis and Kravaris (2001) is illustrated in Figure 1: the state x is transformed into $T(x)$, of higher dimension d_z, whose linear dynamics defines a contraction with A a Schur stable matrix[1]. Hence one can arbitrarily initialize the latent state z, the contraction properties guarantees its converge towards $T(x)$. Since T is continuous and injective, an estimate $\hat{x} = T^{-1}(z)$ can be computed which converges towards x. To see that the latent dynamics $z_{k+1} = Az_k + By_k$ defines a contraction, one can pick two initial values z_0^a and z_0^b and see that their trajectories will evolve as $z_{k+1}^a - z_{k+1}^b = Az_k^a + By_k - (Az_k^b + By_k) = A(z_k^a - z_k^b)$ that tends to 0 for $k \to +\infty$ (as A is Schur stable).

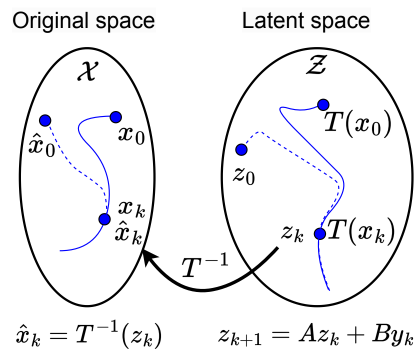


Figure 1: KKL observer principle.

In the autonomous case, under mild assumptions[2] on system (2), Brivadis et al. (2019) have shown that for nearly all Schur stable matrices $A \in \mathbb{R}^{d_z \times d_z}$ with $d_z \geq d_y(2d_x + 1)$ and taking $B = 1_{d_z \times d_y}$ the mapping T exists and is unique. An observer for (2) is then given by:

$$\left\{ \begin{array}{lll} z_{k+1} & = & Az_k + By_k, \; z_0 \in T(\mathcal{X}_0) \\ \hat{x}_k & = & T^{-1}(z_k). \end{array} \right. \tag{3}$$

However learning a good pair (A, T^{-1}) may not be sufficient to attain a good behavior during the transient phase: as illustrated in Figure 2, there is no guarantee that a latent state z that is not initialized exactly at $T(x_0)$ remains in the set $T(\mathcal{X})$. In such situations, the behavior of the observer (3) is undefined as z is outside the definition domain of T^{-1}.

To address this issue, an extension $\mathcal{T}^*$ of the mapping T^{-1} must be defined. To this end, Bernard and Andrieu (2018) proposed to project z into $T(\mathcal{X})$ before applying T^{-1}.

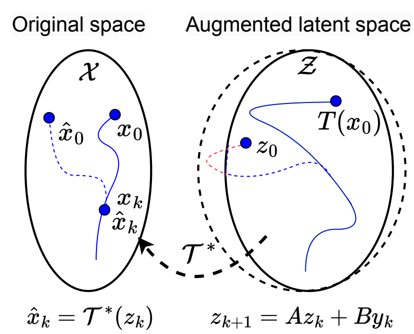


Figure 2: The main theoretical issue.

Although the theory has proved the existence of such observers, no general method is given for finding analytic expressions for the transformations involved. Moreover, the projection method for extending T^{-1} relies on estimating $T(\mathcal{X})$, which is a challenge in the model-free setting. Another open question concerns the choice of matrix A, about which little is known (Brivadis et al., 2023): we know that the modulus of its eigenvalues affects convergence speed and sensitivity to noise; but we still don't know whether optimizing all the coefficients of A can significantly improve overall performance. In the next Section, we propose a novel deep-learning method to answer these questions.

1. A square matrix is Schur stable if the eigen-values of A are inside the unitary circle, i.e. $\rho(A) < 1$.
2. The system is required to be time reversible and backward distinguishable. f^{-1} and h are assumed to be continuous. More formal formulation of these assumptions can be found in (Brivadis et al., 2019).

3. Method: Neural Networks based KKL observer

In the following, the observer design problem is formalized as an optimal problem. Two distinct methods are introduced for training deep networks to obtain, respectively, optimized transient and asymptotic behaviors, which are then efficiently combined through a switching approach.

Neural network structure. Introducing a switching variable $i \in \{1, 2\}$, the observer has the following computational structure. Given a sequence of observations y_k,

$$z_0 = \phi^{(i)}(.), \tag{4}$$
$$z_{k+1} = A^{(i)} z_k + B y_k, \tag{5}$$
$$\hat{x}_k = \psi^{(i)}(z_k), \tag{6}$$

where $\phi^{(i)}$, $A^{(i)}$ and $\psi^{(i)}$ depend on the desired behavior, asymptotic $(i = 1)$ or transient $(i = 2)$, and $\phi^{(i)}$ are initialization functions of the latent state based on information available at initialization. These components will be derived further below for each of the two behaviors (Sections 3.1-3.2). In what follows we will sometimes omit the superscript $^{(i)}$ to ease notation.

Learning from trajectories. During execution[3], at time-step k the observer provides an estimate $\hat{X}_k$ of the state, relying only on the (past) history of the output $Y_{1:k-1}$. However during the training phase, the observer leverages insights on the true state values X_k. Following the same model-free framework as (Miao and Gatsis, 2023), we rely on ground truth state trajectory observations during training. This assumption corresponds to numerous applications where measurements are available during the development phase of an industrial product, while a less expensive sensor network is used for mass production. Another important use case is sim2real transfer, for instance in robotics, where ground-truth states are available during training in simulation.

3.1. Asymptotic Observer

The asymptotic observer is shown in Figure 3: the latent space is initialized as $z_0 = \phi^{(1)} \triangleq T(x_0)$, hence the dynamics $A^{(1)}$ is not required to be fast and is optimized for noise rejection. During execution, at a time-step h, the latent state can be initialized with the transient observer estimation $\hat{x}_h^{transient}$. The transformation $\psi^{(1)}$ reconstructing the original state x is defined as the inverse of the first mapping, $\psi^{(1)} \triangleq T^{-1}$. The overall asymptotic model is end-to-end trained, optimizing the components $\{T, T^{-1}, A^{(1)}\}$.

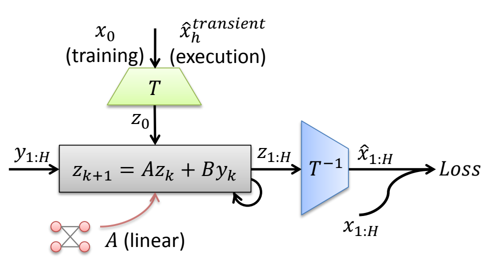


Figure 3: Asymptotic observer layout.

3.2. Transient Observer

In contrary to previous work, which initializes the latent state z at a fixed value, we propose to encode y to the latent state with a trainable function ϕ_θ, $z_0 = \phi^{(2)} \triangleq \phi_\theta(y_0)$.

3. We refer to *execution*, as opposed to *training*, the use of the observer in its nominal operation.

This is motivated by the desire to initialize z close to $T(x)$ (based on the information available during execution) and then improve the transient behavior.

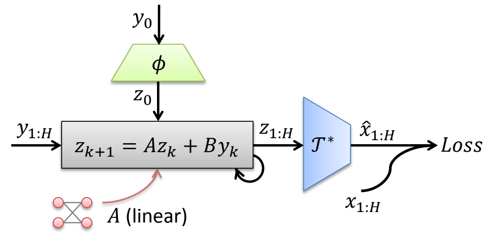


Figure 4: Transient observer layout.

The transient variant also learns an extension $\psi^{(2)} \stackrel{\Delta}{=} \mathcal{T}^*$ in order to address the practical problem of peaking; z being initialized only on the knowledge of the output y_0 — i.e. possibly initialized far from $T(x_0)$ — the latent dynamics is driven to be fast. The overall transient model is end-to-end trained, optimizing the components $\{\phi^{(2)}, \mathcal{T}^*, A^{(2)}\}$.

Learned extension $\mathcal{T}^*$. Let $\mathbb{Z}(\mathcal{X}_0, w, z_0)$ be the set of all possible values of z generated by the latent dynamics $z_{k+1} = Az_k + B\left(h(x_k) + w_k\right)$ initialized at $\phi^{(2)}(z_0)$ coupled to the system dynamics (2). The adopted neural network scheme, aims to learn the mapping $\mathcal{T}^* : \mathbb{Z} \to \mathcal{X}$ such that $\mathcal{T}^*(z) = T^{-1}(z)$ for all $z \in \mathbb{Z} \cap T(\mathcal{X})$. The last component of the computational chain (6) is in charge to complete the definition of $\mathcal{T}^*$, by assigning images to all $z \in \mathbb{Z} \setminus T(\mathcal{X})$, i.e. assigning some estimation $\hat{x}$ for the points z that are not in $T(\mathcal{X})$. In contrary to the mapping T^{-1} that is unique for a given matrix A, the extension $\mathcal{T}^*$ is optimized through the computational chain of Figure 4, offering an additional opportunity to improve the observer behavior during the transient phase.

3.3. Optimizing the latent dynamics

Optimization problem. System (3) is an observer, meaning that $\hat{x}$ converges towards x in absence of noise, with a rate of convergence depending on the matrix A. If the noise is bounded, then the estimation error will also be bounded with a magnitude that also depends on A. Here appears the necessary trade-off between convergence speed and robustness: the faster the latent dynamics, the faster the convergence of the observer value $\hat{x}$ towards x; on the other hand noise can be more efficiently filtered at the price of a slower convergence.

Hence, we formulate our objective as an optimization problem. The observer parameters, namely the coefficients of matrix A, must be chosen to minimize the expected error $\mathcal{L}$ of the observer along a trajectory:

$$\mathcal{L} = \mathbb{E}_{w,x_0} \sum_{k=1}^{K} ||X_k(x_0) - \hat{X}_k||_2. \tag{7}$$

Parametrization of the latent dynamics. We propose two representations that both allow A to possess arbitrary complex eigenvalues. The first one is totally unconstrained: the matrix $A \in \mathbb{R}^{d_z \times d_z}$ is viewed as d_z^2 parameters to optimize, allowing the best possible expressivity of the high-capacity deep network. The drawback is that A is not ensured to be stable. Although in practice we found that the learned matrices are stable provided training trajectories are sufficiently long (see Section 4), we propose a second representation that guarantees stability. This second representation of A is the following parametrized block-diagonal matrix:

$$A = \text{diag}\left(\frac{1}{1+\exp^{-\alpha}} \begin{bmatrix} \cos(\omega) & -\sin(\omega) \\ \sin(\omega) & \cos(\omega) \end{bmatrix}\right), \tag{8}$$

where eigenvalues are some pairs of complex conjugates, each of them being parametrized by a pair of real values (α, ω). Notice that the use of the Sigmoid function — the first factor in (8) — constrains the eigenvalues to remain inside the unitary circle, while maintaining differentiability.

3.4. Switching approach

Our objective, considering two modes (transient and asymptotic), is to identify an efficient instance for transitioning between them. We base this decision on a performance evaluation of each mode, leveraging the information accessible during execution: the output y and the observer internal variable z. Specifically, we build upon the methodology proposed by Petri et al. (2023) that we adapt to the KKL change of coordinates.

Monitoring variables. Let the monitoring variables $\eta^{(i)} \in \mathbb{R}_{\geqslant 0}$, with dynamics defined by

$$\eta_{k+1}^{(i)} = a\eta_k^{(i)} + \epsilon_k^{(i)\top} \Lambda \epsilon_k^{(i)}$$

where $0 \leq a \leq 1$ and $\Lambda \in \mathbb{R}^{d_x \times d_x}$ a definite positive matrix are parameters, and $\epsilon^{(i)}$ a vector estimating the current performance in the state estimation. Monitoring variables $\eta^{(i)}$ hence act as filters on performance estimations, integrating them with a common forgetting factor a. Matrix Λ can be used as a scaling parameter for each component of the state x (in the case all components of x are similar in scale, a simple choice for Λ is the Identity matrix).

To design the performance criteria ϵ of a KKL observer, notice that the evolution of the state can be expressed as

$$\begin{aligned} x_{k+1} &= T^{-1}\left(T(x_{k+1})\right. \\ &= T^{-1}\left(AT(x_k) + By_k\right). \end{aligned} \tag{9}$$

Hence, in absence of error, the state estimation (or *open-loop* estimate) would evolve as

$$\begin{aligned} \hat{x}_{k+1} &= T^{-1}\left(AT(\hat{x}_k) + B\hat{y}_k\right), \ \hat{y}_k \triangleq h(\hat{x}_k) \\ &= T^{-1}\left(Az_k + B\hat{y}_k\right), \end{aligned} \tag{10}$$

while from (3) the actual state estimation (or *corrected* estimate) can be expressed as

$$\hat{x}_{k+1} = T^{-1}\left(Az_k + By_k\right). \tag{11}$$

Considering, the difference between corrected estimate and open-loop dynamics, we obtain the following approximation on the estimation error:

$$\epsilon_{k+1} = T^{-1}\left(Az_k + By_k\right) - T^{-1}\left(Az_k + B\hat{y}_k\right), \tag{12}$$

or, for a specific observer i

$$\epsilon_{k+1}^{(i)} = \psi^{(i)}\left(A^{(i)}z_k^{(i)} + By_k\right) - \psi^{(i)}\left(A^{(i)}z_k^{(i)} + B\hat{y}_k^{(i)}\right), \ \hat{y}_k^{(i)} \triangleq h(\hat{x}_k^{(i)}). \tag{13}$$

Switching condition. The monitoring variables $\eta^{(i)}$ hence provide a filtered estimation of the error of each observer. The state estimate $\hat{x}_k$ of the hybrid observer at time-step k is then the one corresponding to the lower value $\eta_k^{(i)}$.

Remark 1 *A simpler evaluation than (12) could be used by considering an estimation of the error $T(x) - z$ in the latent space through the computation of $\tilde{\epsilon}_k \triangleq B\left(y_k - h(\hat{x}_k)\right)$. But this results in a less reliable estimation of the error, as a lower distance in the latent space does not necessarily imply a lower distance in the original space.*

4. Numerical results

In this section the proposed switching approach for KKL observers is tested on three different benchmarks: first two classical systems, namely the Van der Pol system and the Rössler attractor, illustrate the gain of the method when compared to state-of-art KKL-observers; the third benchmark is a bioprocess, on which we show that the optimization of latent dynamics significantly improves the transient behavior compared to existing an analytic KKL observer.

Example 1 (Van der Pol) *We consider the Van der Pol oscillator as seen, for example, by Ramos et al. (2020); Peralez and Nadri (2021); Miao and Gatsis (2023):*

$$\begin{cases} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \left(1 - x_1^2\right) x_2 - x_1 \\ y &= x_1 \end{cases} \tag{14}$$

To apply the methodology presented in Section 3, a set of training trajectories are generated over the time interval $[0, 4]$, with initial states picked from a uniform random distribution. To demonstrate the gain of the method regarding the literature, the performance of transient, asymptotic and hybrid observers are assessed following the same procedure as in (Miao and Gatsis, 2023): the root mean square error (RMSE) over the time interval $[0, 50]$ for a batch of 1000 initial conditions are reported in Table 1. When compared to best known results, the RMSE for our hybrid strategy is reduced by a factor of 2 in presence of noise and by a factor of 40 without noise. The transient and asymptotic observers behavior is further illustrated in Figure 5: while the transient observer is the faster, the asymptotic one exhibits a better accuracy and a better robustness with respect to measurement noises, particularly in presence of outliers.

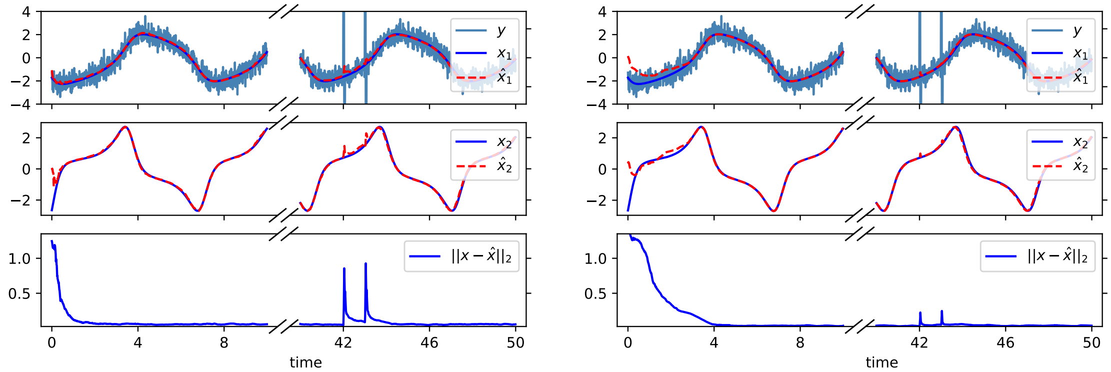
No noise			
Observer	RMSE on time intervals:		
	$[0, 50]$	$[0, 4]$	$[4, 50]$
Supervised*	0.0548	n.a.**	n.a.
Neural ODEs*	0.0603	n.a.	n.a.
Transient	0.0018	0.0069	0.0013
Asymptotic	0.0068	0.0626	0.0007
Hybrid	$\mathbf{0.0014}$	0.0069	0.0008

Noisy measurements			
Observer	RMSE on time intervals:		
	$[0, 50]$	$[0, 4]$	$[4, 50]$
Supervised*	0.1160	n.a.	n.a.
Neural ODEs*	0.0667	n.a.	n.a.
Transient	0.0715	0.121	0.0660
Asymptotic	0.0532	0.701	0.0275
Hybrid	$\mathbf{0.0340}$	0.121	0.0244

Table 1: Van der Pol benchmark results: comparison of root mean square error (RMSE).
* Results presented by Miao and Gatsis (2023). ** Not available.

Example 2 (Rössler attractor) *We consider the following nonlinear system, where the parameter values are fixed as Niazi et al. (2023) ($a = 0.2$, $b = 0.2$, and $c = 5.7$):*

$$\begin{cases} \dot{x}_1 &= -x_2 - x_3 \\ \dot{x}_2 &= x_1 + a\, x_2 \\ \dot{x}_3 &= x_3 \left(x_1 - c\right) \\ y &= x_2 \end{cases} \tag{15}$$



Transient observer. Asymptotic observer.

Figure 5: Illustration of Van der Pol observers with noisy measurement: $w \sim \mathcal{N}(0, 0.5)$. Some measurement *outliers* are introduced at $t = 42$ and $t = 43$ to test their robustness.

Following the procedure from Niazi et al. (2023), we train our observers on a set of trajectories initialized from $\mathcal{X}_0 = [-1, 1]^3$ over the time interval $[0, 50]$ as illustrated in Figure 6. We compare the performance with two (model-based) physically informed neural networks (PINN), obtained with the code provided by Peralez and Nadri (2021) and Niazi et al. (2023). These two methods being not intended to learn the latent dynamics, the same matrix A as the one proposed in Niazi et al. (2023) is used. Although our method is model-free, it allows a drastic reduction in estimation errors as shown in Table 2.

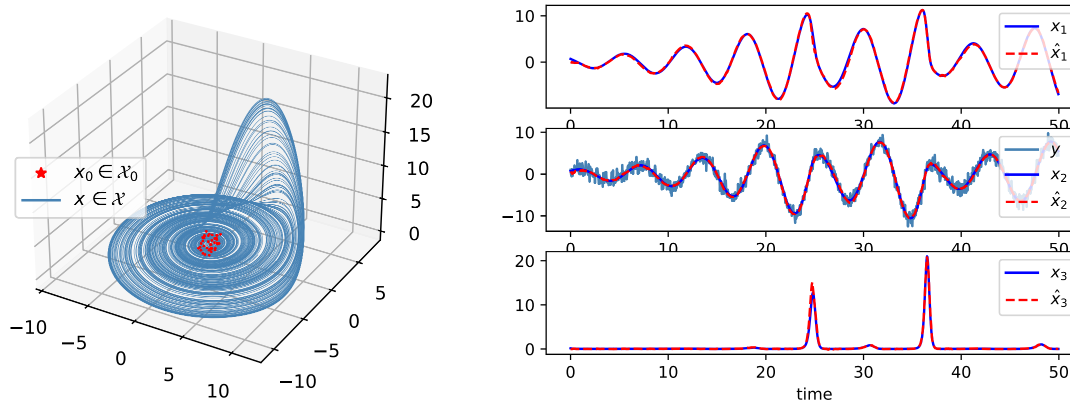
No noise			
Observer	RMSE on time intervals:		
	$[0, 50]$	$[0, 4]$	$[4, 50]$
dPINN*	0.340	0.172	0.453
cPINN**	0.335	0.217	0.348
Transient	0.0146	0.0361	0.0113
Asymptotic	0.0324	0.202	0.0062
Hybrid	**0.0131**	0.0361	0.0065

Noisy measurements			
Observer	RMSE on time intervals:		
	$[0, 50]$	$[0, 4]$	$[4, 50]$
dPINN*	1.220	1.0649	1.3241
cPINN**	n.a.	n.a.	n.a.
Transient	0.210	0.269	0.201
Asymptotic	0.218	0.435	0.185
Hybrid	**0.198**	0.269	0.187

Table 2: Rössler benchmark results: comparison of root mean square error (RMSE). *Code available online: https://github.com/jolindien-git/DeepKKL. **Code available online: https://github.com/Mudhdhoo/ACC_KKL_Observer

Example 3 (Bioprocess) *We consider a bioreactor which consists of a microbial culture which involves a biomass x_1 growing on a substrate of concentration x_2. This bioprocess is fed with an input substrate concentration x_{2in} at a time-varying dilution rate u. Following the "Contois" growth rate, a classical dynamical model of the process is*

$$\left\{ \begin{array}{rcl} \dot{x}_1 & = & \frac{K_1 x_2}{K_2 x_1 + x_2} x_1 - u\, x_1 \\ \dot{x}_2 & = & -K_3 \frac{K_1 x_2}{K_2 x_1 + x_2} - u(x_2 - x_{2in}) \\ y & = & x_1 \end{array} \right. \tag{16}$$



Data: states are initialized in the compact set $\mathcal{X}_0 = [-1, 1]^3 \supset \mathcal{X}$.

Observer: result example in presence of noise, i.e. $y = h(x) + w$, $w \sim \mathcal{N}(0, 1)$.

Figure 6: The Rössler attractor example.

where K_i are constants that are fixed at 1 while $x_{2in} = 0.1$ as in Bernard and Andrieu (2018).

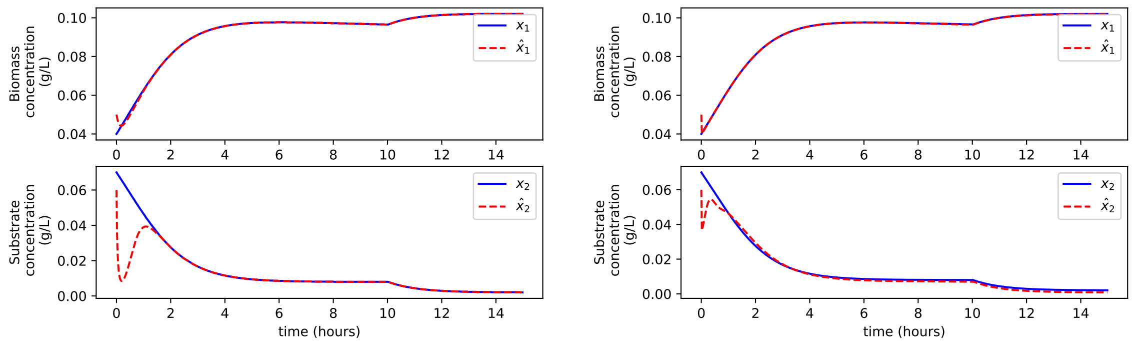
To handle this non autonomous system, the latent dynamics A is first optimized on data obtained for a constant dilution rate $u^0 = 0.05\, h^{-1}$, following a similar methodology to the two previous examples. Then to test the observer behavior in realistic scenarios, i.e. for a time-varying u, the observer (3) is adapted as proposed by Peralez and Nadri (2021):

$$\begin{cases} z_{k+1} &= Az_k + By_k + \Psi(z_k, u_k) \\ \hat{x}_k &= T^{-1}(z_k) \end{cases} \tag{17}$$

with $\Psi(z_k, u_k) := T(f(T^{-1}(z_k), u_k)) - T(f(T^{-1}(z_k), u^0))$. Note that to apply (17), knowledge of the system dynamics f is assumed (the problem of a model-free method for the non-autonomous case will be addressed in a future work). To demonstrate the benefits of optimizing the latent dynamics, the resulting observer is compared to the KKL observer proposed by Bernard and Andrieu (2018). Therein, analytic expressions for the mappings T and T^{-1} are found in the case of diagonal matrices A. Their result is reproduced in Figure 7 and compared to our hybrid observer. Interestingly, our learned latent dynamics exhibit a similar convergence speed, but with a significant improvement during the transient phase, where the peaking phenomenon is prevented.

5. Related work

Theory on KKL-observers. This paper uses the sufficient conditions for the existence of an observer of the form (3) proved for the autonomous case by Brivadis et al. (2019). However therein, no constructive method is given to find the involved transformation, except for the class of system with linear dynamics and polynomial output. Peralez and Nadri (2021) adapted this observer for the non-autonomous case, giving some convergence guarantees for sufficiently small control values. In the continuous-time-KKL framework, Bernard and Andrieu (2018) proposed a method to extend the mapping T^{-1} by projection in the latent space.



Analytic-KKL observer
from Bernard and Andrieu (2018).

Deep-KKL observer
with optimized latent dynamics.

Figure 7: The Bioprocess example.

Deep learning on KKL-observers. Similarly to our work Peralez and Nadri (2021) and Peralez et al. (2022) invoked the discrete-time KKL theory, but developed model-based methods. Peralez et al. (2022) proposed a (model-based) method to learn the extension of T^{-1} by projection. They also exhibited significant gains in asymptotic accuracy thanks to model ensembling that use a set of similar observers. In contrary our approach rely on finding different observers with different properties specialized to different conditions (convergence speed, peaking magnitude, asymptotic accuracy, noise rejection). Miao and Gatsis (2023) developed a data-driven method based on Neural ODEs, learning the latent dynamics, but restricting the choice to some diagonal matrices. Like in our approach they also aimed to minimize the observer error, hence we compared the results of both approaches in Section 4. There are still very few articles devoted to the implementation of Deep-KKL methods for realistic systems (Wang et al., 2022; Li et al., 2023; Buisson-Fenet et al., 2023), which motivated us to consider a bioprocess (example 3). It is worth noting that a few works outside the observers domain use the KKL theory, see e.g. (Janny et al., 2021) for the output prediction or Buisson-Fenet et al. (2022) for system identification.

6. Conclusion

We propose a new model-free methodology to learn the Kazantzis-Kravaris-Luenberger (KKL) observer by addressing three major challenges in observer design: the peaking phenomenon, noise rejection, and the trade-off between convergence speed and robustness. We provide a general framework where two observers are learned for both the transient phase and asymptotic convergence. Subsequently, we design a switching criterion, based on monitoring variables, that selects one mode at any given time by evaluating their performance in the presence of disturbances. This ensures effective noise rejection and alleviates the need for a trade-off between convergence speed and robustness. The gain of the method regarding the literature is illustrated on three classical benchmarks.

Two natural direction for further exploration arise within this established framework: i) Extending the proposed algorithm to encompass a set of more than two observers. ii) Given the fundamental role of state estimation in output regulation issues, a logical extension of this study would involve developing a model-free method tailored for the non-autonomous case.

References

Vincent Andrieu and Laurent Praly. On the existence of a kazantzis–kravaris/luenberger observer. *SIAM Journal on Control and Optimization*, 2006.

Pauline Bernard and Vincent Andrieu. Luenberger observers for nonautonomous nonlinear systems. *IEEE Transactions on Automatic Control*, 2018.

Pauline Bernard, Vincent Andrieu, and Daniele Astolfi. Observer design for continuous-time dynamical systems. *Annual Reviews in Control*, 2022.

Lucas Brivadis, Vincent Andrieu, and Ulysse Serres. Luenberger observers for discrete-time nonlinear systems. In *IEEE Conference on Decision and Control (CDC)*, 2019.

Lucas Brivadis, Vincent Andrieu, Pauline Bernard, and Ulysse Serres. Further remarks on kkl observers. *Systems & Control Letters*, 2023.

Mona Buisson-Fenet, Valery Morgenthaler, Sebastian Trimpe, and Florent Di Meglio. Recognition models to learn dynamics from partial observations with neural odes. *Transactions on Machine Learning Research*, 2022.

Mona Buisson-Fenet, Lukas Bahr, Valery Morgenthaler, and Florent Di Meglio. Towards gain tuning for numerical kkl observers. *IFAC-PapersOnLine*, 2023.

Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder–decoder approaches. *Syntax, Semantics and Structure in Statistical Translation*, 2014.

Albert Gu, Karan Goel, and Christopher Re. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations (ICLR)*, 2021.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 1997.

Steeven Janny, Vincent Andrieu, Madiha Nadri, and Christian Wolf. Deep kkl: Data-driven output prediction for non-linear systems. In *IEEE Conference on Decision and Control (CDC)*, 2021.

Nikolaos Kazantzis and Costas Kravaris. Discrete-time nonlinear observer design using functional equations. *Systems & Control Letters*, 2001.

Yingquan Li, Zhenwen Liang, Ibrahima N'Doye, Xiangliang Zhang, Mohamed-Slim Alouini, and Taous-Meriem Laleg-Kirati. Data-driven state estimation for light-emitting diode underwater optical communication. *IFAC-PapersOnLine*, 2023.

Keyan Miao and Konstantinos Gatsis. Learning robust state observers using neural odes. In *Learning for Dynamics and Control Conference*, 2023.

Muhammad Umar B Niazi, John Cao, Xudong Sun, Amritam Das, and Karl Henrik Johansson. Learning-based design of luenberger observers for autonomous nonlinear systems. In *American Control Conference (ACC)*, 2023.

Johan Peralez and Madiha Nadri. Deep learning-based luenberger observer design for discrete-time nonlinear systems. In *IEEE Conference on Decision and Control (CDC)*, 2021.

Johan Peralez, Madiha Nadri, and Daniele Astolfi. Neural network-based kkl observer for nonlinear discrete-time systems. In *IEEE Conference on Decision and Control (CDC)*, 2022.

E Petri, R Postoyan, D Astolfi, D Nesic, and V Andrieu. Hybrid multi-observer for improving estimation performance. *arXiv preprint arXiv:2303.06936*, 2023.

Louise da C Ramos, Florent Di Meglio, Valery Morgenthaler, Luís F Figueira da Silva, and Pauline Bernard. Numerical design of luenberger observers for nonlinear systems. In *IEEE Conference on Decision and Control (CDC)*, 2020.

Yubin Wang, Yasmine Marani, and Taous Meriem Laleg Kirati. A deep-learning-based observer for state estimation of direct contact membrane distillation system modeled by differential algebraic equations. In *2022 IEEE Conference on Control Technology and Applications (CCTA)*, 2022.