

Sparse Bayesian Learning for Koopman Based System Identification

Selin Ezgi Özcan¹ and Mustafa Mert Ankaralı²

Abstract—Modeling nonlinear dynamical systems is important for control, prediction, and decision making in numerous engineering fields. While robust methods for identifying linear systems are well established, traditional approaches often struggle with complex nonlinear behavior, particularly when data is noisy or limited. Recently, the Koopman operator framework has gained popularity as it provides global linear representations of nonlinear systems, which enables the use of linear systems theory tools. However, existing data driven Koopman operator methods, such as extended dynamic mode decomposition (EDMD), can yield degraded performance under noisy conditions and typically require large amounts of training data. In this paper, we propose a new approach for learning Koopman models using sparse Bayesian learning (SBL). Our method identifies the beliefs over Koopman model parameters directly from model evidence while inherently promoting sparsity. Additionally, our approach demonstrates improved noise robustness and effective performance even with limited data. Unlike existing methods, the SBL approach provides valuable uncertainty quantification for the learned models, which is particularly beneficial for control applications requiring probabilistic performance guarantees.

I. INTRODUCTION

Modeling and identification of dynamical systems have been a crucial step in the analysis, control, and optimization of various engineering applications. Among all types of applications, nonlinear systems are the most commonly encountered in fluid dynamics [1], [2], robotics [3]–[7] and weather forecasting [8]. Accurate system models are essential for the design of robust controllers and decision-making algorithms, particularly in applications characterized by complex, nonlinear behaviors. Existing methods for nonlinear system identification such as neural network [9] or Gaussian process [10] based approaches lack easy interpretability and scalability. Moreover, many well established tools and methods for linear systems are not straightforward for nonlinear models.

Recently, the Koopman operator theory has stood out as a powerful framework to overcome some of the shortcomings of traditional nonlinear system identification methods [11], [12]. The Koopman operator essentially lifts the dynamics to an infinite dimensional function space, where the time evolution representation is linear. Hence many of the usual linear system analysis and control tools become readily available for nonlinear systems. This operator theoretic perspective has

¹Selin Ezgi Özcan is with Department of Electrical and Electronics Engineering, Middle East Technical University and Türk Telekom Inc., Ankara, Türkiye. selinezgi.ozcan2@turktelekom.com.tr

²Mustafa Mert Ankaralı is with the Department of Electrical and Electronics Engineering and Center for Robotics and Artificial Intelligence, Middle East Technical University, Ankara, Türkiye. mertan@metu.edu.tr

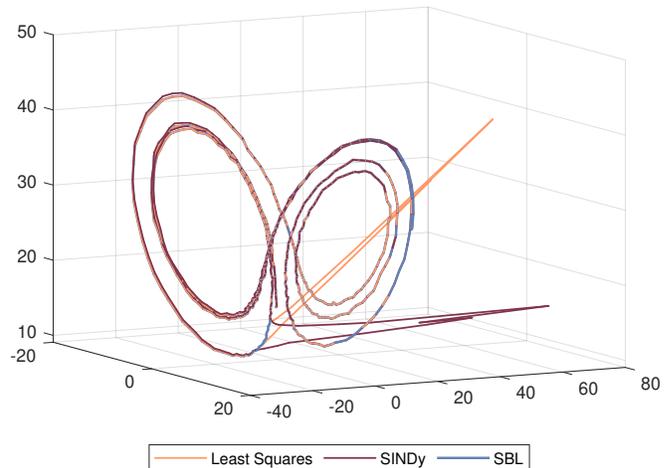


Fig. 1. One step prediction results of the Koopman model trained with 2000 data samples for the chaotic Lorenz attractor. Least squares and sparsity promoting SINDy algorithm diverge occasionally while SBL solution closely follows the expected trajectory.

gained increasing attention in control and decision systems due to its ability to capture complex behaviors without requiring explicit knowledge of the system’s nonlinear structure [1], [5], [7]. Famous data driven algorithms that employ the Koopman operator begin with a dictionary of functions embedding the state space to a higher dimensional space, in which the time evolution of the system is described. Methods expanding the Koopman operator to data driven domain such as dynamic mode decomposition (DMD) [13], extended dynamic mode decomposition (EDMD) [14] and neural network based solutions [15] have proven to be useful in numerous applications.

Despite the growing interest in the Koopman operator methods, many challenges remain open, including the systematic identification of the operator from finite and noisy data as well as the selection of dictionary of functions [4], [16], [17]. DMD type approaches are known to cause mode mixing under noisy data, leading to inaccurate models [18], [19]. Some methods promoting certain properties such as stability or sparsity of the approximation have been proposed to address certain shortcomings [20], [21]. In particular, regularization based approaches such as SINDy [21] aim to accurately capture the system dynamics under noisy observations by promoting sparsity. The algorithm implements the so called sequential thresholded least squares, where small enough matrix entries are iteratively forced to zero. Despite the advantage of direct influence of the user on the level of sparsity, the algorithm needs to be tuned for data amount

and the noise level. Moreover, forcing small weights to zero under a single threshold for all functions in the dictionary is unintuitive, since the scale of the functions can be drastically different and may cause arbitrarily large or small weights without becoming unimportant. Lastly, many least squares based data driven Koopman approaches require large datasets to perform reliably in predictive tasks [17].

In this paper, we propose sparse Bayesian learning (SBL) [22] of the EDMD based Koopman model for dynamical systems. SBL provides a probabilistic framework for promoting sparsity in the learned model by automatically pruning irrelevant functions in the dictionary through Bayesian inference, thereby improving generalization in the presence of noisy or limited data. Further, unlike frequentist approaches where model parameters are considered as deterministic entities to be extracted, Bayesian framework allows uncertainty quantification [23], which is promising for safety guaranteed control applications.

The contributions of this paper are threefold: (i) we reformulate the Koopman model identification problem as a SBL problem; (ii) we verify that sparse Bayesian Koopman model learning is more robust against measurement noise than some conventional approaches; and (iii) we test both the proposed approach and the existing methods with small datasets and conclude that sparse Bayesian Koopman model learning can work reliably with small training data while existing methods underperform.

II. BACKGROUND

A. Discrete Time Koopman Operator

Let a discrete time system be described by the map $\mathbf{S} : X \rightarrow X$,

$$\mathbf{x}_{k+1} = \mathbf{S}(\mathbf{x}_k), \quad (1)$$

where X is the state space of the system. Let the functions $\psi : X \rightarrow \mathbb{R}$ defined on the state space be denoted as observables and let them constitute a Hilbert space H . The discrete time Koopman operator $\mathcal{K} : H \rightarrow H$ acts on the observables by a composition [12]

$$\mathcal{K}\psi = \psi \circ \mathbf{S}. \quad (2)$$

\mathcal{K} is a linear operator on H by nature of composition. It thus provides a possibly infinite dimensional global linear representation for the system unlike linearization around a point in the state space. Several data driven approaches have been developed to approximate \mathcal{K} in finite dimensions [11].

B. Extended Dynamic Mode Decomposition (EDMD) Algorithm

EDMD is one of the most popular algorithms for data driven approximation of \mathcal{K} [14]. The algorithm relies on a preselected dictionary of observables that ideally constitute an invariant set under \mathcal{K} . Given a set of snapshot pairs $\{(\mathbf{x}[i], \mathbf{x}[i'])\}_{i=1}^m$, where $\mathbf{x}[i] \in \mathbb{R}^n$ represents the state of the dynamical system at time t and $\mathbf{x}[i']$ is the corresponding state at time $t + \Delta t$, EDMD approximates the Koopman operator \mathcal{K} as finite dimensional matrix $\hat{\mathbf{K}}$ by using the dictionary

of functions $\phi(\mathbf{x}) = [\phi_1(\mathbf{x}) \dots \phi_L(\mathbf{x})]^T$. These observables lift the state space into a higher dimensional feature space where $\hat{\mathbf{K}}$ describes a linear dynamical system in terms of the lifted states $\phi[k] = \phi(\mathbf{x}[k])$

$$\phi^T[k+1] = \phi^T[k]\hat{\mathbf{K}}. \quad (3)$$

Common selections of dictionary functions are orthogonal polynomials, Fourier basis or Gaussian kernels [11] with $\phi_j(\mathbf{x}) = x_j$ for $j = 1, \dots, n$.

Once the dictionary is set, the so called lifted state matrices \mathbf{X} and \mathbf{X}' are constructed by evaluating the dictionary of functions for each data pair in the data set:

$$\mathbf{X} = [\phi(\mathbf{x}[1]) \mid \phi(\mathbf{x}[2]) \mid \dots \mid \phi(\mathbf{x}[m])]^T \in \mathbb{R}^{m \times L}, \quad (4)$$

$$\mathbf{X}' = [\phi(\mathbf{x}[1']) \mid \phi(\mathbf{x}[2']) \mid \dots \mid \phi(\mathbf{x}[m'])]^T \in \mathbb{R}^{m \times L}. \quad (5)$$

Then, the best-fit linear operator $\hat{\mathbf{K}}$ that satisfies $\mathbf{X}' \approx \hat{\mathbf{K}}\mathbf{X}$ is computed by solving the minimization problem

$$\hat{\mathbf{K}} = \arg \min_{\mathbf{K}} \|\mathbf{X}' - \mathbf{X}\mathbf{K}\|_F^2. \quad (6)$$

The approximation of \mathcal{K} (6) does not immediately apply to controlled systems. In order to allow for control inputs, EDMD procedure is modified [24]. In the controlled case, \mathbf{X} is concatenated with the corresponding control inputs $\mathbf{u}[i] \in \mathbb{R}^l$

$$\mathbf{X}_U = \begin{bmatrix} \phi(\mathbf{x}[1]) & \dots & \phi(\mathbf{x}[m]) \\ \mathbf{u}[1] & \dots & \mathbf{u}[m] \end{bmatrix}^T \in \mathbb{R}^{m \times (L+l)}. \quad (7)$$

The optimization problem (6) is then reformulated by replacing \mathbf{X} with \mathbf{X}_U . The solution provides the approximate system and input matrices $\hat{\mathbf{A}}$ and $\hat{\mathbf{B}}$, respectively.

$$\hat{\mathbf{K}} = \begin{bmatrix} \hat{\mathbf{A}} \\ \hat{\mathbf{B}} \end{bmatrix} = \arg \min_{\mathbf{K}} \|\mathbf{X}' - \mathbf{X}_U\mathbf{K}\|_F^2. \quad (8)$$

The output matrix $\hat{\mathbf{C}}$ is determined by the inverse map from the lifted space to the state space. Assuming no feedforward dynamics, the so called Koopman model of the system is formed as a linear dynamical system

$$\begin{aligned} \phi^T[k+1] &= \phi^T[k]\hat{\mathbf{A}} + \mathbf{u}^T[k]\hat{\mathbf{B}} \\ \mathbf{x}[k] &= \hat{\mathbf{C}}\phi[k]. \end{aligned} \quad (9)$$

C. Sparse Bayesian Learning (SBL) Framework

SBL is a powerful supervised learning method that uses Bayesian inference and boosts sparsity in the model parameters [22]. The goal of SBL is to identify a preferably sparse model for measured inputs and target outputs. SBL is especially useful in parameter estimation of high dimensional models, making it favorable in the Koopman model context.

Given input variables $\mathbf{x} = [x_1, \dots, x_n]^T$, basis functions $\phi(\mathbf{x}) = [\phi_1(\mathbf{x}), \dots, \phi_L(\mathbf{x})]^T$ and weights $\mathbf{w} = [w_1, \dots, w_L]^T$, the target variable is modeled as

$$t = \phi(\mathbf{x})^T \mathbf{w} + v, \quad (10)$$

where v is zero mean Gaussian noise with precision ρ . Assume $\mathbf{X} = [\mathbf{x}_1 \dots \mathbf{x}_m]^T$ and $\mathbf{t} = [t_1 \dots t_m]^T$ represent the

collection of m input and target variables, respectively. Then, \mathbf{t} can be compactly expressed as

$$\mathbf{t} = \Phi \mathbf{w} + \mathbf{v}, \quad (11)$$

where

$$\Phi = \begin{bmatrix} \phi(\mathbf{x}_1)^T \\ \vdots \\ \phi(\mathbf{x}_m)^T \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} v_1 \\ \vdots \\ v_m \end{bmatrix}. \quad (12)$$

As the number of parameters increases, maximum likelihood estimate (MLE) of (11) can overfit. SBL addresses this problem by assigning independent conditional prior distributions to each weight, which is a zero mean Gaussian prior such that

$$p(\mathbf{w} | \alpha) = \prod_{\ell=0}^L \mathcal{N}(w_\ell; 0, \alpha_\ell^{-1}), \quad (13)$$

where each α_ℓ is the precision of the corresponding weight and $\alpha = [\alpha_1, \dots, \alpha_L]^T$. Lastly, each hyperparameter is associated with a hyperprior. A common hyperprior is Gamma distribution, resulting in

$$p(\alpha) = \prod_{\ell=0}^L \text{Gamma}(\alpha_\ell; a, b), \quad (14)$$

$$p(\rho) = \text{Gamma}(\rho; c, d), \quad (15)$$

where Gamma distribution is specified with shape and rate parameters. The mean and covariance of the weight posterior $p(\mathbf{w} | \mathbf{t}, \alpha, \rho) = \mathcal{N}(\mathbf{w}; \mu, \Sigma)$ can be shown to equal [25], [26]

$$\Sigma = \left(\rho \Phi^T \Phi + \text{diag}(\alpha) \right)^{-1}, \quad \mu = \rho \Sigma \Phi^T \mathbf{t}. \quad (16)$$

Under the SBL framework, the aim is to find the maximum a posteriori (MAP) estimates of parameters α and ρ ; namely,

$$\{\hat{\alpha}_{\text{MAP}}, \hat{\rho}_{\text{MAP}}\} = \arg \max_{\alpha, \rho} p(\mathbf{t} | \alpha, \rho) p(\alpha) p(\rho). \quad (17)$$

Using hyperpriors with $a = c = 1$ and $b, d \rightarrow 0$ permits the MAP estimate of the parameters α and ρ to correspond to MLE [18], [27]. We can then employ expectation maximization (EM) algorithm to iteratively find the estimates for all the parameters [28]. Letting μ^i and Σ^i be estimates of weights posterior mean and covariance at the i -th iteration, the estimates for α and ρ are found using the M-step [26]

$$\begin{aligned} \hat{\alpha}_\ell^{i+1} &= \frac{2a-1}{(\mu_\ell^i)^2 + \Sigma_{\ell\ell}^i + 2b}, \\ \hat{\rho}^{i+1} &= \frac{m+2c-2}{\|\mathbf{t} - \Phi \mu^i\|^2 + \text{tr}(\Sigma^i \Phi^T \Phi) + 2d}, \end{aligned} \quad (18)$$

where μ_ℓ and α_ℓ are the ℓ -th elements of μ and α . As a result of the E-step of the algorithm, the posterior mean and covariance of weights are iteratively calculated as [26]

$$\Sigma^{i+1} = \left(\hat{\rho}^i \Phi^T \Phi + \text{diag}(\hat{\alpha}^i) \right)^{-1}, \quad \mu^{i+1} = \hat{\rho}^i \Sigma^i \Phi^T \mathbf{t}. \quad (19)$$

The EM updates (18) and (19) are carried out until convergence. In this regard, EM can have slow convergence rates and take many iterations to converge. Moreover, it can

reach suboptimal minima [26], [28]. On the other hand, SBL leads to improved sparsity and recovery performance without requiring manual tuning of regularization parameters. Furthermore, its Bayesian formulation provides uncertainty estimates, making it especially useful in settings where robustness and interpretability are important.

III. METHOD

Following the notation introduced in Section II, let $\phi = [\phi_1 \dots \phi_L]^T$ be the dictionary of observables. The Koopman model (9) relates the values of the observables at consecutive time steps. Let $D = [\mathbf{x}[1] \dots \mathbf{x}[m]]$ be the set of measurements obtained from a dynamical system at discrete times $1, \dots, m$. We try to fit the model between the values of the observables at time instants $k = 1, \dots, m-1$ and $k' = 2, \dots, m$. If the system is controlled, we may also denote the control inputs at time instants k by $\mathbf{U} = [\mathbf{u}[1] \dots \mathbf{u}[m-1]]$. In the SBL framework, we construct the lifted data matrix Φ similarly to (12) by augmenting the control inputs if they are available:

$$\Phi = \begin{bmatrix} \phi_1(\mathbf{x}[1]) & \dots & \phi_L(\mathbf{x}[1]) & \mathbf{u}^T[1] \\ \vdots & \ddots & \vdots & \vdots \\ \phi_1(\mathbf{x}[m-1]) & \dots & \phi_L(\mathbf{x}[m-1]) & \mathbf{u}^T[m-1] \end{bmatrix}. \quad (20)$$

Target variables are set as the values of observables at time instants k' . Here, we assume the fit of each observable is independent of others and solve L independent SBL problems by setting the target variable in (11) accordingly. For values of each observable in the dictionary, the input-target relation is formulated individually as

$$\mathbf{t}_j = \Phi \mathbf{w}_j + \mathbf{v}_j, \quad (21)$$

where $\mathbf{t}_j = [\phi_j(\mathbf{x}[2]) \dots \phi_j(\mathbf{x}[m])]^T$ for $j = 1, \dots, L$ and the aim is to estimate \mathbf{w}_j . The SBL model form (21) facilitates the EM parameter updates (18) and (19) to identify the Koopman matrix entries column by column for L independent problems. The Koopman matrix $\hat{\mathbf{K}}$ is given as

$$\hat{\mathbf{K}} = [\mathbf{w}_1 | \dots | \mathbf{w}_L]. \quad (22)$$

We hence identify a Koopman model with process noise:

$$\phi^T[k+1] = \phi^T[k] \hat{\mathbf{A}} + \mathbf{u}^T[k] \hat{\mathbf{B}} + \varepsilon, \quad (23)$$

where $\varepsilon \sim \mathcal{N}(\varepsilon; \mathbf{0}, \text{diag}(\rho_1^{-1}, \dots, \rho_L^{-1}))$.

IV. RESULTS

In this section, we present system identification results obtained via two simulated dynamical systems, one autonomous and one controlled, to validate our proposed system identification approach. The autonomous system is a Lorenz attractor, while the controlled system is an unmanned surface vehicle (USV). To examine algorithm performance under uncertainty, we introduce artificial zero mean Gaussian measurement noise at different standard deviation levels into the data. We apply noise with the same variance level uniformly across all states within each noise level experiment. Due to differences in the mean and scale of the states, the resulting

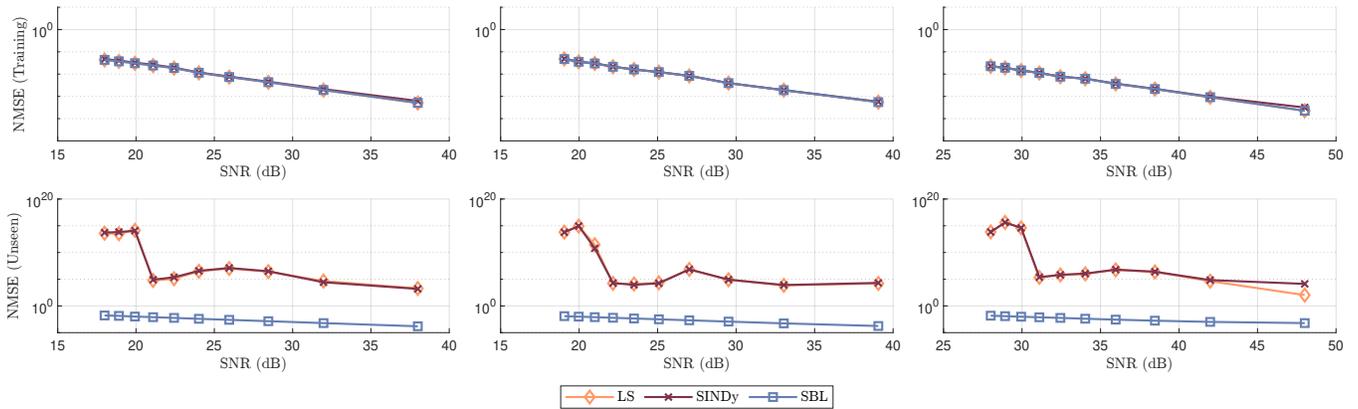


Fig. 2. One step prediction NMSE of least squares solution (LS), SINDy and SBL of Lorenz attractor Koopman model identification. Top row: Results on training data. Bottom row: Results on previously unseen test data. Left column: Results for x component. Middle column: Results for y component. Right column: Results for z component.

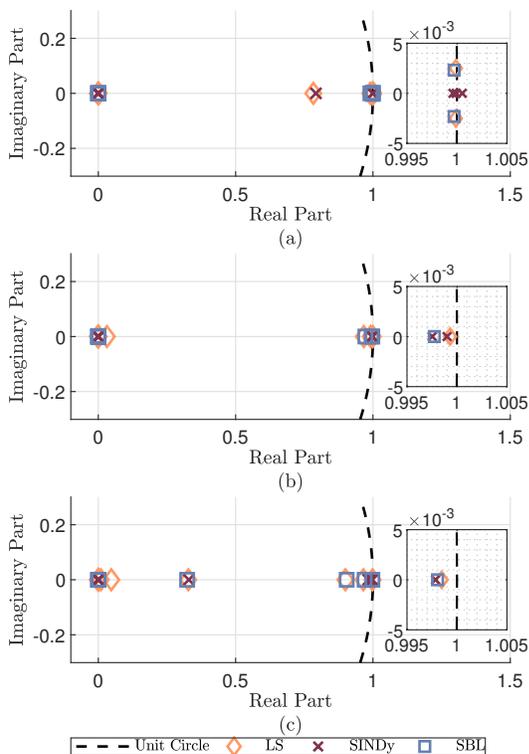


Fig. 3. Eigenvalues of the system matrix of Lorenz attractor Koopman model found using least squares solution (LS), SINDy and SBL for different measurement noise standard deviations. (a): $\sigma = 0.1$, (b): $\sigma = 0.5$, (c): $\sigma = 1$

signal-to-noise ratio (SNR) varies among the individual state variables. We compare the results of our proposed method under various noise levels with those of the ordinary least squares solution ((6) or (8)) and the SINDy method [21], which implements an algorithm called sequential thresholded least squares. SINDy sets a sparsity threshold λ and iteratively discards observables whose weights are smaller than λ in magnitude. This approach provides a sparsity knob, enabling users to control the sparsity level of the resulting model. For all experiments involving our method, we set $a = c = 1$ and $b = d = 10^{-8}$.

A. Lorenz attractor

The Lorenz attractor is a three-dimensional chaotic nonlinear system with coupled differential dynamics [29]

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} \sigma(y-x) \\ x(\rho-z)-y \\ xy-\beta z \end{bmatrix}. \quad (24)$$

We set the parameters $\sigma = 10$, $\beta = 8/3$, $\rho = 28$ and sample the continuous system at 1000 Hz for 1 second for training. The observables in the dictionary are selected as identity functionals and Gaussian kernels, in accordance with [17].

In Figure 2, we study the effect of SNR on the one step normalized mean squared error (NMSE) performance of the proposed method. SINDy threshold is selected as $\lambda = 0.05$ for comparison. While the performances are comparable for all three methods in the training data, SBL drastically outperforms existing methods in unseen data. Moreover, performance of the proposed method decays with diminishing SNR; however, it is seen in Figure 2 that other methods are far more susceptible to SNR. A reason behind the difference in performances can be because SINDy requires λ to be tuned, which should be handled according to each noise level. Hence, setting a fixed λ under different noise levels can affect the performance. In cases where Φ in (12) only contains the measurements, adding white noise to input variables is equivalent to adding regularization term in least squares algorithms. Nevertheless, in the Koopman operator framework, the noise is added to measurements while the inputs to the least squares problem are possibly nonlinear functions of measurements. This causes the measurement noise to have unprecedented effects on numerical stability. Figure 2 is indicative of the need for higher regularization in the least squares type solutions. In the case of SINDy, the sparsity threshold must be handled individually for each observable since the effect of white measurement noise results in possibly nonsymmetric, biased, nonwhite or non-Gaussian noise after each nonlinear transform. Further, here the Koopman model is fit with a small dataset on purpose to highlight the need of Bayesian methods for less data than

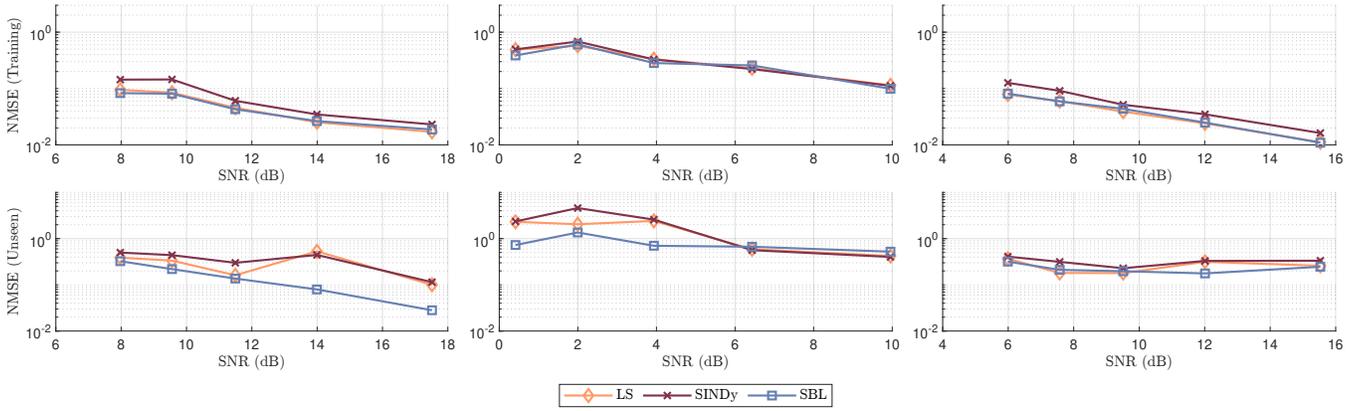


Fig. 4. One step prediction NMSE of least squares solution (LS), SINDy and SBL of USV Koopman model identification. Top row: Results on training data. Bottom row: Results on previously unseen test data. Left column: Results for surge velocity state. Middle column: Results for sway velocity state. Right column: Results for angular velocity state.

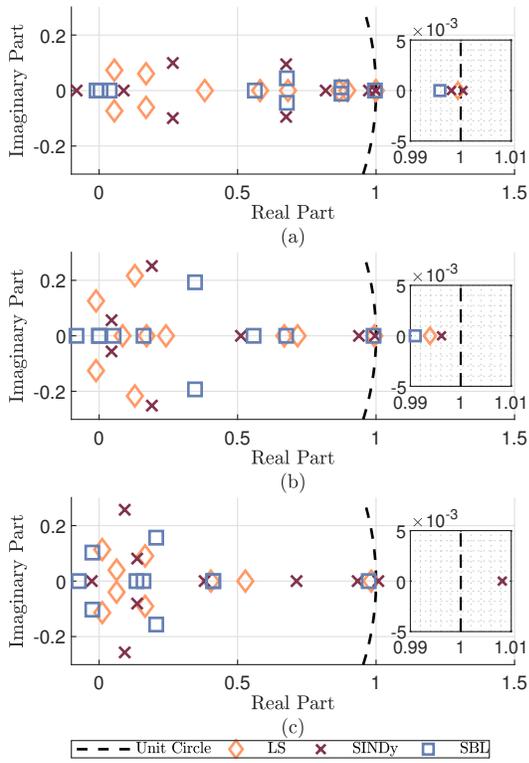


Fig. 5. Eigenvalues of the system matrix of USV Koopman model found using least squares solution (LS), SINDy and SBL for different measurement noise standard deviations. (a): $\sigma = 0.1$, (b): $\sigma = 0.2$, (c): $\sigma = 0.3$

that of frequentist approaches [23]. In Figure 1, we increase the training data duration to 2 seconds. Still, other methods occasionally peak while SBL maintains the expected trajectory. Our experiments of Lorenz attractor demonstrate that SBL is more robust against SNR and data amount without parameter tuning.

Next, we investigate the eigenvalues of the Koopman system matrix \hat{A} in (9). Once again, all of the parameters for SBL and the sparsity threshold for SINDy are kept constant while SNR is varied. Figure 3 depicts that SINDy may

be unstable if λ is not properly tuned for the noise level even though the original system is stable. In this regard, SBL seems more numerically reliable since the identified model at least agrees with the stability of the system. This is because SBL does not force a predefined sparsity on the model, it updates the belief of weights and precisions based on evidence instead.

B. Underactuated unmanned surface vehicle (USV)

In this section, we test the proposed method on a controlled system to assess the algorithm in a more practical use case. We consider the simulation of an underactuated USV that enjoys the dynamics [30]

$$M\dot{v} + C(v)v + D(v)v = \tau. \quad (25)$$

In (25), $v \in \mathbb{R}^3$ contains the surge, sway and angular velocity states of the vehicle. M , C and D stand for mass, coriolis and damping matrices, respectively. The thruster force vector, denoted by $\tau \in \mathbb{R}^3$, is considered as the control input to the system. The effect of τ is inherently influenced by the placement of the thrusters on the vehicle. In the present underactuated configuration, the thruster forces have no direct influence on the sway velocity state [31]. The system dynamics are simulated, with state and control data sampled at a frequency of 10 Hz over a duration of 180 seconds for training. The control inputs are modeled using a zero order hold assumption when performing state predictions with the Koopman operator framework. Similar to Section IV-A, λ for SINDy is fixed at 0.05 for comparison. Observables ϕ are set as identity functionals and Gaussian kernels.

Figure 4 illustrates that SBL is superior in terms of NMSE especially in previously unseen data prediction performance while other methods fail. The difference between the training dataset errors and test dataset errors of other methods indicates overfitting and can presumably be quite detrimental for the long term prediction performance of the model. This is concerning since the Koopman model has been used in model predictive control of dynamical systems increasingly frequently [4].

Figure 5 shows that other methods are less robust to noise since they can possibly identify unstable systems. SBL indeed reinforces robustness against measurement noise when compared to other methods as it treats the problem probabilistically because the probabilistic approach is numerically less dramatically affected by outliers. Besides, that other methods identify unstable system matrices but achieve stable NMSE performance in Figure 4 can be a sign of mode mixing between the observables and the control inputs.

V. CONCLUSION

In this work, we find the parameters of the Koopman model in (9) using SBL framework. Unlike ordinary least squares or sparsity inducing algorithms such as SINDy, SBL allows the weight probability distributions to be updated via evidence maximization. The probabilistic approach grants an inherent notion of certainty of the parameters, where the precision of each model parameter is also derived as a hyperparameter. Rather than forcing sparsity via a threshold, SBL dynamically shrinks the weights if necessary. This eliminates threshold tuning, remedying the scenarios with unknown measurement noise. Additionally, our method does not discard functions based on weight magnitudes, as the lifting function scales can be different. The probabilistic treatment that SBL offers is particularly suited to small datasets and to model based control tasks with safety bound requirements. Extensions of the proposed method can introduce sequential elimination of observables like SINDy when the dictionary entries have comparable norms.

ACKNOWLEDGMENT

This work was partially supported by the Scientific and Technological Research Council of Türkiye (TÜBİTAK) 1001 Program under project number 124E232. This work was also partially supported by TÜBİTAK 1515 Frontier R&D Laboratories Support Program for Türk Telekom neXt Generation Technologies Lab (XGeNTT) under project number 5249902. The authors would like to thank Umut Orguner for his valuable insights and Mustafa Yıldırım for part of the simulation data.

REFERENCES

- [1] I. Mezić, "Analysis of fluid flows via spectral properties of the Koopman operator," *Annu. Rev. Fluid Mech.*, vol. 45, no. 1, pp. 357–378, 2013.
- [2] S. Bagheri, "Koopman-mode decomposition of the cylinder wake," *J. Fluid Mech.*, 2013.
- [3] J. Wang, B. Xu, J. Lai, Y. Wang, C. Hu, H. Li, and A. Song, "An improved Koopman-MPC framework for data-driven modeling and control of soft actuators," *IEEE Robot. Autom. Lett.*, vol. 8, no. 2, pp. 616–623, 2022.
- [4] J. Pan, D. Li, J. Wang, P. Zhang, J. Shao, and J. Yu, "Autogeneration of mission-oriented robot controllers using Bayesian-based Koopman operator," *IEEE Trans. Robot.*, vol. 40, pp. 903–918, 2023.
- [5] I. Abraham, G. De La Torre, and T. D. Murphey, "Model-based control using Koopman operators," *arXiv preprint arXiv:1709.01568*, 2017.
- [6] S. E. Otto and C. W. Rowley, "Koopman operators for estimation and control of dynamical systems," *Annu. Rev. Control, Robot. Auton. Syst.*, vol. 4, no. 1, pp. 59–87, 2021.
- [7] L. Shi, M. Haseli, G. Mamakoukas, D. Bruder, I. Abraham, T. Murphey, J. Cortes, and K. Karydis, "Koopman operators in robot learning," *arXiv preprint arXiv:2408.04200*, 2024.
- [8] J. Hogg, M. Fonoberova, and I. Mezić, "Exponentially decaying modes and long-term prediction of sea ice concentration using Koopman mode decomposition," *Sci. Rep.*, vol. 10, no. 1, p. 16313, 2020.
- [9] A. Dong, A. Starr, and Y. Z. and, "Neural network-based parametric system identification: a review," *Int. J. Syst. Sci.*, vol. 54, no. 13, pp. 2676–2688, 2023. [Online]. Available: <https://doi.org/10.1080/00207721.2023.2241957>
- [10] J. Umlauf and S. Hirche, "Feedback linearization based on Gaussian processes with event-triggered online learning," *IEEE Trans. Autom. Control*, vol. 65, no. 10, pp. 4154–4169, 2019.
- [11] A. Mauroy, Y. Susuki, and I. Mezić, *Koopman operator in systems and control*. Springer, 2020, vol. 484.
- [12] B. O. Koopman, "Hamiltonian systems and transformation in Hilbert space," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 17, no. 5, pp. 315–318, 1931.
- [13] P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," *J. Fluid Mech.*, vol. 656, pp. 5–28, 2010.
- [14] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A data-driven approximation of the Koopman operator: Extending dynamic mode decomposition," *J. Nonlinear Sci.*, vol. 25, pp. 1307–1346, 2015.
- [15] B. Lusch, J. N. Kutz, and S. L. Brunton, "Deep learning for universal linear embeddings of nonlinear dynamics," *Nat. Commun.*, vol. 9, no. 1, p. 4950, 2018.
- [16] Q. Li, F. Dietrich, E. M. Bollt, and I. G. Kevrekidis, "Extended dynamic mode decomposition with dictionary learning: A data-driven adaptive spectral decomposition of the Koopman operator," *Chaos Interdiscip. J. Nonlinear Sci.*, vol. 27, no. 10, 2017.
- [17] M. Khosravi, "Representer theorem for learning Koopman operators," *IEEE Trans. Autom. Control*, vol. 68, no. 5, pp. 2995–3010, 2023.
- [18] A. U. Secen, "Bayesian inference methods for detection of power system oscillations," Master's thesis, Middle East Technical University, 2023.
- [19] J. N. Kutz, X. Fu, and S. L. Brunton, "Multiresolution dynamic mode decomposition," *SIAM J. Appl. Dyn. Syst.*, vol. 15, no. 2, pp. 713–735, 2016.
- [20] G. Mamakoukas, I. Abraham, and T. D. Murphey, "Learning stable models for prediction and control," *IEEE Trans. Robot.*, vol. 39, no. 3, pp. 2255–2275, 2023.
- [21] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 113, no. 15, pp. 3932–3937, 2016.
- [22] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, no. Jun, pp. 211–244, 2001.
- [23] S. Särkkä and L. Svensson, *Bayesian filtering and smoothing*. Cambridge university press, 2023, vol. 17.
- [24] J. L. Proctor, S. L. Brunton, and J. N. Kutz, "Generalizing Koopman theory to allow for inputs and control," *SIAM J. Appl. Dyn. Syst.*, vol. 17, no. 1, pp. 909–930, 2018.
- [25] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*. Springer, 2006, vol. 4, no. 4.
- [26] D. G. Tzikas, A. C. Likas, and N. P. Galatsanos, "The variational approximation for Bayesian inference," *IEEE Signal Process. Mag.*, vol. 25, no. 6, pp. 131–146, 2008.
- [27] M. Tipping, "The relevance vector machine," *Adv. Neural Inf. Process. Syst.*, vol. 12, 1999.
- [28] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. R. Stat. Soc. Ser. B Methodol.*, vol. 39, no. 1, pp. 1–22, 1977.
- [29] E. N. Lorenz, "Deterministic nonperiodic flow," *J. Atmos. Sci.*, vol. 20, no. 2, pp. 130 – 141, 1963.
- [30] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons, 2011.
- [31] S. Atasoy, O. K. Karagöz, and M. M. Ankarali, "Trajectory-free motion planning of an unmanned surface vehicle based on MPC and sparse neighborhood graph," *IEEE Access*, vol. 11, pp. 47 690–47 700, 2023.