Finite-Time Prediction Error Analysis for Neural Bilinear Koopman Models Under Noisy Measurements

Zhi Xu, Jun Xiang, Jun Chen, *Senior Member, IEEE*, and Ran Dai, *Senior Member, IEEE*

***Abstract*—Learning-based Koopman models provide a structured framework for modeling nonlinear controlled systems. However, the finite-time prediction reliability of learned Koopman models under noisy measurements has not been sufficiently investigated. This letter studies multi-step prediction error propagation for neural bilinear Koopman models of continuous-time control-affine systems under noisy state and input measurements. We first derive certified properties of the neural lifting function by exploiting its structure, and then analyze the local model mismatch by decomposing it into lifting inconsistency and discretization mismatch. Based on these results, we establish an explicit finite-horizon prediction error bound that captures the joint effects of noisy measurements and local mismatch, and further reveals a trade-off on finite-time prediction error with respect to the sampling period. Numerical experiments are presented to support the theoretical findings.**

***Index Terms*—Machine learning and control, neural networks, neural koopman models, error analysis.**

I. Introduction

KOOPMAN operator theory provides a powerful framework for modeling nonlinear dynamical systems by representing their evolution via infinite-dimensional linear operators acting on observable functions [1], [2], [3]. In practice, realizing this framework requires approximating the operator on a finite-dimensional subspace, making the selection of lifting functions a critical challenge. To overcome the limitations of hand-crafted dictionaries, neural networks have recently emerged as a powerful alternative for learning high-quality lifting functions from data [4], [5], [6]. For controlled nonlinear systems, while early models simply assumed linear lifted dynamics [7], [8], recent paradigms shift toward the bilinear Koopman representation, which is a more principled formulation because the Koopman generator depends affinely on the input and therefore induces bilinear lifted dynamics [9], [10]. Together, data-driven neural lifting and bilinear formulations form a natural and powerful synergy for modeling controlled nonlinear dynamics [11], [12].

Despite these promising developments, the reliability of learning-based Koopman models remains insufficiently investigated, especially for neural bilinear Koopman models deployed under noisy measurements. Existing analytical studies providing theoretical error guarantees exhibit distinct limitations when applied to neural lifting architectures. Specifically, Nüske et al. [13] derived finite-data bounds for the operator estimation and prediction errors, which quantifies the impact of data size on operator estimation. While pioneering, the analysis assumed that the lifting functions are invariant under the Koopman operator, ignoring the lifting inconsistency arising from a finite-dimensional lifted space. Accounting for this limitation, Strässer et al. [14] derived deterministic pointwise error bounds for kernel-based methods within reproducing kernel Hilbert spaces (RKHS), leveraging Wendland kernels to ensure an invariant lifted space by construction. Nevertheless, the analysis is tightly bound to the structures of RKHS and kernel methods, making them incapable of generalizing to highly expressive neural network architectures.

Beyond the error analysis in noise-free settings, accounting for measurement noise is essential for evaluating the reliability of learning-based Koopman models in practical scenarios. Hao et al. [15] characterized and mitigated the impact of noise during training by dynamically updating the tunable observables. More recently, Sakib and Pan [16] developed a robust learning framework against corrupted datasets using Hankel-structured observables and stable parameterizations. While existing noise-aware studies have made important progress, they primarily focus on robust operator identification during training rather than providing explicit error bounds for the deployment phase. During deployment, measurement noise is mapped through the neural lifting function and recursively

Received 16 March 2026; revised 23 May 2026; accepted 16 June 2026. Date of publication 1 July 2026; date of current version 9 July 2026. This work was supported in part by Grant NSF-2334995, Grant NSF-2523830, Grant NSF CCF-2402689, and Grant CCF-2523829. Recommended by Senior Editor T. Oomen.
(Corresponding author: Jun Chen.)

Zhi Xu and Ran Dai are with the School of Aeronautics and Astronautics, Purdue University, West Lafayette, IN 47907 USA (e-mail: xu1710@purdue.edu; randai@purdue.edu).

Jun Xiang and Jun Chen are with the Aerospace Engineering Department, San Diego State University, San Diego, CA 92182 USA (e-mail: jxiang9143@sdsu.edu; jun.chen@sdsu.edu).

Digital Object Identifier 10.1109/LCSYS.2026.3709152

propagated through the lifted dynamics, making its effect difficult to quantify due to the nonlinear and layered nature of the lifting network. This motivates leveraging recent advances in neural network certification, including bound propagation [17], [18] and local Lipschitz estimation [19], which enable rigorous analysis of the output range and perturbation sensitivity of the neural lifting functions.

Motivated by the above research gaps, this letter studies finite-time prediction reliability for neural bilinear Koopman models under noisy measurements. We first derive certified properties of neural lifting functions by leveraging the network structure. We then characterize local prediction mismatch through a decomposition into lifting inconsistency and discretization-induced mismatch. Building on these results, we develop a recursive error propagation analysis and derive an explicit finite-horizon prediction error bound.

The contributions are summarized as follows:

- Certified analysis of general neural lifting functions beyond the Koopman-invariant assumption in [13] and the RKHS settings considered in [14].
- A local mismatch decomposition that explicitly incorporates lifting inconsistency arising from finite-dimensional neural lifting.
- A finite-horizon prediction error bound that characterizes recursive deployment-phase noise propagation and reveals a sampling-period trade-off.

II. Preliminaries

Koopman operator theory represents nonlinear dynamics linearly in a lifted space of observables. Consider the continuous-time control-affine system

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u}) = \boldsymbol{f}_0(\boldsymbol{x}) + \textstyle\sum_{i=1}^m \boldsymbol{f}_i(\boldsymbol{x})u_i, \tag{1}$$

where state $\boldsymbol{x} \in \mathbb{R}^n$, control $\boldsymbol{u} \in \mathbb{R}^m$, dynamics $\boldsymbol{f}_0, \boldsymbol{f}_i$ are locally Lipschitz.

Let $\mathcal{F}$ be a space of locally Lipschitz observables $g : \mathbb{R}^n \to \mathbb{R}$. For a fixed input $\boldsymbol{u}$, the associated Koopman generator is

$$(\mathcal{L}_{\boldsymbol{u}} g)(\boldsymbol{x}) = \nabla g(\boldsymbol{x}) \cdot \left(\boldsymbol{f}_0(\boldsymbol{x}) + \textstyle\sum_{i=1}^m \boldsymbol{f}_i(\boldsymbol{x})u_i\right).$$

Hence, the generator $\mathcal{L}_{\boldsymbol{u}} = \mathcal{L}_0 + \sum_{i=1}^m u_i \mathcal{L}_i$ depends affinely on the input, where

$$(\mathcal{L}_0 g)(\boldsymbol{x}) = \nabla g(\boldsymbol{x}) \cdot \boldsymbol{f}_0(\boldsymbol{x}), \quad (\mathcal{L}_i g)(\boldsymbol{x}) = \nabla g(\boldsymbol{x}) \cdot \boldsymbol{f}_i(\boldsymbol{x}).$$

Assume there exists a finite-dimensional lifting function $\psi(\boldsymbol{x}) = [\psi_1(\boldsymbol{x})\, \psi_2(\boldsymbol{x}) \cdots \psi_N(\boldsymbol{x})]^\top$ whose span is invariant under $\mathcal{L}_0, \mathcal{L}_1, \ldots, \mathcal{L}_m$. Then there exist matrices $K_0, K_1, \ldots, K_m \in \mathbb{R}^{N\times N}$ such that

$$\tfrac{d}{dt}\psi(\boldsymbol{x}) = K_0\psi(\boldsymbol{x}) + \textstyle\sum_{i=1}^m u_i K_i \psi(\boldsymbol{x}).$$

Defining the lifted state $z = \psi(\boldsymbol{x})$, the lifted dynamics becomes

$$\dot{z} = K_0 z + \textstyle\sum_{i=1}^m u_i K_i z, \tag{2}$$

which is bilinear in the input and the lifted state.

III. Problem Formulation

We consider the problem of finite-time prediction using a learned neural bilinear Koopman model for the continuous-time control-affine system in (1). The true system evolves in continuous time, while the learned predictor is a discrete-time lifted model identified from sampled data. In practice, both the initial state measurement and the applied input sequence may be corrupted by noise. Our goal is to characterize how these perturbations, together with the local mismatch between the learned lifted dynamics and the true system flow, affect multi-step state prediction over a finite time.

A. Learned Neural Bilinear Koopman Predictor

We now introduce the discrete-time neural bilinear Koopman model considered in this letter.

We consider a neural lifting function $\psi_\theta : \mathbb{R}^n \to \mathbb{R}^{n_z}$ of the form $\hat{z} = \psi_\theta(\boldsymbol{x}) = [1, \boldsymbol{x}^\top, \Phi_\theta^\top(\boldsymbol{x})]^\top$, where $\Phi_\theta : \mathbb{R}^n \to \mathbb{R}^{n_p}$ is a feedforward neural network with parameter θ. By construction, the lifted state contains the constant observable and the original state. The state is recovered from the lifted state by $\hat{\boldsymbol{x}} = P\hat{z}$ with $P = [0\ \boldsymbol{I}_n\ \boldsymbol{0}_{n\times n_p}] \in \mathbb{R}^{n\times n_z}$.

The nonlinear component Φ_θ is implemented as a feedforward ReLU network, so that ψ_θ is locally Lipschitz and differentiable almost everywhere. Although we focus on this architecture for concreteness, the subsequent analysis can also be applied to other lifting-network structures and activation functions.

The lifted dynamics is modeled in bilinear form as

$$\hat{z}_{k+1} = K_\theta \hat{z}_k + \textstyle\sum_{i=1}^m u_{k,i} K_{\theta,i} \hat{z}_k, \tag{3}$$

where $K_\theta, K_{\theta,1}, \ldots, K_{\theta,m} \in \mathbb{R}^{n_z\times n_z}$ are implemented as linear layers without bias. All trainable parameters of the neural network, including the lifting network weights and operator matrices, are collected in θ.

B. Finite-Time Prediction Error Under Noise

Consider the continuous-time control-affine system (1). The input is implemented using a zero-order-hold (ZOH), i.e., $\boldsymbol{u}(t) = \boldsymbol{u}_k$ for $t \in [t_k, t_{k+1})$, where $t_k = k\Delta t$. Let $\phi_{\boldsymbol{u}_k}^{\Delta t}(\boldsymbol{x}_k)$ denotes the flow map over one sampling interval; the exact discrete-time state evolution is $\boldsymbol{x}_{k+1} = \phi_{\boldsymbol{u}_k}^{\Delta t}(\boldsymbol{x}_k)$.

Assume that the initial state and control sequence are affected by additive noise such that $\tilde{\boldsymbol{x}}_0 = \boldsymbol{x}_0 + \boldsymbol{w}$, $\tilde{\boldsymbol{u}}_k = \boldsymbol{u}_k + \boldsymbol{v}_k$, and there exist constants $\bar{w}, \bar{v}_i > 0$ such that

$$\|\boldsymbol{w}\| \le \bar{w},\ |v_{k,i}| \le \bar{v}_i,\ i = 1, \ldots, m, \tag{4}$$

where $\|\cdot\|$ denotes the 2-norm. The neural Koopman model then generates predictions in the lifted space. Starting from the noisy lifted initial condition $\tilde{z}_0 = \psi_\theta(\tilde{\boldsymbol{x}}_0)$, the lifted trajectory is propagated according to

$$\tilde{z}_{k+1} = K_\theta \tilde{z}_k + \textstyle\sum_{i=1}^m \tilde{u}_{k,i} K_{\theta,i} \tilde{z}_k. \tag{5}$$

The predicted state affected by measurement noise is recovered from the lifted state by $\tilde{\boldsymbol{x}}_k = P\tilde{z}_k$.

For a fixed terminal time $T > 0$, let $H \in \mathbb{N}$ satisfy $T = H\Delta t$. The finite-time state prediction error is therefore defined as

$\boldsymbol{\epsilon}_k := \tilde{\boldsymbol{x}}_k - \boldsymbol{x}_k$, $k = 0, 1, \ldots, H$. The objective of this letter is to derive an explicit upper bound on $\|\boldsymbol{\epsilon}_k\|$ over a finite prediction horizon.

IV. Finite-Time Prediction Error Analysis

A. Certified Bounds for the Neural Lifting Function

To analyze the effect of measurement noise on the learned lifting map ψ_θ, we require certified bounds on the magnitude of the neural network Φ_θ and the Lipschitz constant over a prescribed input region. For the former, we use CROWN-based bound propagation [17], which computes affine lower and upper bounds on the network output over a given input region. For the latter, we use a certified local Lipschitz bound approach [19].

Assume the true state evolves in a compact set $\mathcal{X} \subset \mathbb{R}^n$. Since the initial measurement satisfies $\tilde{\boldsymbol{x}}_0 = \boldsymbol{x}_0 + \boldsymbol{w}$ with $\|\boldsymbol{w}\| \leq \bar{w}$, the noisy initial state belongs to the perturbed set

$$\mathcal{X}_0 := \mathcal{X} \oplus \mathcal{B}(\bar{w}) = \{\boldsymbol{x} + \boldsymbol{w} : \boldsymbol{x} \in \mathcal{X},\ \|\boldsymbol{w}\| \leq \bar{w}\},$$

where $\mathcal{B}(\bar{w})$ denotes the closed norm ball of radius $\bar{w}$.

To apply CROWN-based bound propagation, we enclose the relevant state region in a box $\mathcal{I} := [\underline{\boldsymbol{x}}, \bar{\boldsymbol{x}}] = \{\boldsymbol{x} \in \mathbb{R}^n : \underline{\boldsymbol{x}} \leq \boldsymbol{x} \leq \bar{\boldsymbol{x}}\}$, chosen such that $\mathcal{X}, \mathcal{X}_0 \subseteq \mathcal{I}$. Given the ReLU network $\Phi_\theta : \mathbb{R}^n \to \mathbb{R}^{n_p}$, the bound propagation approach computes affine lower and upper bounds over $\mathcal{I}$ for each output coordinate $j = 1, \ldots, n_p$:

$$\underline{a}_j^\top \boldsymbol{x} + \underline{b}_j \leq [\Phi_\theta(\boldsymbol{x})]_j \leq \bar{a}_j^\top \boldsymbol{x} + \bar{b}_j, \quad \forall \boldsymbol{x} \in \mathcal{I}.$$

Optimizing these affine bounds over the box $\mathcal{I}$ yields certified scalar bounds $\underline{\phi}_{\theta,j} \leq [\Phi_\theta(\boldsymbol{x})]_j \leq \bar{\phi}_{\theta,j}$ for $\boldsymbol{x}$ in $\mathcal{I}$ [17].

In addition, we compute a certified local Lipschitz constant of the network over the region $\mathcal{I}$ via bound propagation [19]. For ReLU networks, this is achieved by upper-bounding the norm of the Clarke Jacobian over $\mathcal{I}$. Let L_Φ be a certified local Lipschitz constant of Φ_θ over $\mathcal{I}$, satisfying

$$\|\Phi_\theta(\boldsymbol{x}_1) - \Phi_\theta(\boldsymbol{x}_2)\| \leq L_\Phi \|\boldsymbol{x}_1 - \boldsymbol{x}_2\|, \quad \forall \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{I}. \tag{6}$$

Lemma 1 (Boundedness of the Lifting Network): For all $\boldsymbol{x} \in \mathcal{I}$, $\|\psi_\theta(\boldsymbol{x})\| \leq \bar{z}$, where $\bar{z} := \left(1 + \sum_{i=1}^n \max\{|\underline{x}_i|, |\bar{x}_i|\}^2 + \sum_{j=1}^{n_p} \max\{|\underline{\phi}_{\theta,j}|, |\bar{\phi}_{\theta,j}|\}^2\right)^{1/2}$.

Proof: Since $\|\psi_\theta(\boldsymbol{x})\|^2 = 1 + \|\boldsymbol{x}\|^2 + \|\Phi_\theta(\boldsymbol{x})\|^2$, for any $\boldsymbol{x} \in \mathcal{I} = [\underline{\boldsymbol{x}}, \bar{\boldsymbol{x}}]$ we have

$$\|\boldsymbol{x}\|^2 = \sum_{i=1}^n x_i^2 \leq \sum_{i=1}^n \max\{|\underline{x}_i|, |\bar{x}_i|\}^2.$$

By the certified componentwise bounds on Φ_θ over $\mathcal{I}$,

$$\|\Phi_\theta(\boldsymbol{x})\|^2 \leq \sum_{j=1}^{n_p} \max\{|\underline{\phi}_{\theta,j}|,\ |\bar{\phi}_{\theta,j}|\}^2.$$

Combining the above inequalities proves the result. ■

Lemma 2 (Lipschitz Continuity of the Lifting Network): For all $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{I}$, $\|\psi_\theta(\boldsymbol{x}_1) - \psi_\theta(\boldsymbol{x}_2)\| \leq L_\psi \|\boldsymbol{x}_1 - \boldsymbol{x}_2\|$, where $L_\psi := \sqrt{1 + L_\Phi^2}$.

Proof: By construction,

$$\psi_\theta(\boldsymbol{x}_1) - \psi_\theta(\boldsymbol{x}_2) = [0,\ (\boldsymbol{x}_1 - \boldsymbol{x}_2)^\top,\ (\Phi_\theta(\boldsymbol{x}_1) - \Phi_\theta(\boldsymbol{x}_2))^\top]^\top.$$

By the certified local Lipschitz constant,

$$\|\psi_\theta(\boldsymbol{x}_1) - \psi_\theta(\boldsymbol{x}_2)\|^2 = \|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2 + \|\Phi_\theta(\boldsymbol{x}_1) - \Phi_\theta(\boldsymbol{x}_2)\|^2$$
$$\leq (1 + L_\Phi^2)\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2.$$

Taking square roots on both sides yields the result. ■

B. Noise-Free Local Model Mismatch

To derive a finite-time prediction error bound, we first characterize the local mismatch between the learned lifted model and the true dynamics over one sampling interval. Although continuous-time control-affine systems (1) may admit an exact bilinear Koopman representation (2), the exact discrete-time flow under zero-order hold would be

$$z_{k+1} = \exp\left(\Delta t \left(K_0 + \sum_{i=1}^m u_{k,i} K_i\right)\right) z_k$$

which is generally not bilinear in the input because the exponential generates higher-order mixed terms. Therefore, the learned discrete-time bilinear Koopman model is interpreted as a structured approximation of the exact continuous flow, and it generally incurs an intrinsic discretization mismatch even in the absence of measurement noise.

Consider the noise-free lifted prediction obtained by applying the learned model to the true state and input:

$$\hat{z}_{k+1} = A_\theta(\boldsymbol{u}_k)\psi_\theta(\boldsymbol{x}_k), \tag{7}$$

where the learned control-affine operator is defined as

$$A_\theta(\boldsymbol{u}) := K_\theta + \sum_{i=1}^m u_i K_{\theta,i}. \tag{8}$$

To facilitate comparison with the true continuous-time lifted dynamics, we associate the learned discrete-time operator $A_\theta(\boldsymbol{u})$ with an auxiliary matrix $B_\theta(\boldsymbol{u}) := (A_\theta(\boldsymbol{u}) - I)/\Delta t$. Accordingly, the learned operator can be rewritten as

$$A_\theta(\boldsymbol{u}) = I + \Delta t B_\theta(\boldsymbol{u}), \tag{9}$$

so that the learned update takes the form of a forward Euler discretization of a lifted linear system with generator $B_\theta(\boldsymbol{u})$.

Let $z_\theta(t) := \psi_\theta(\boldsymbol{x}(t))$ denote the learned lifted state along the true trajectory. Since ψ_θ is locally Lipschitz, the following identity holds for a.e. t:

$$\dot{z}_\theta(t) = J_{\psi_\theta}(\boldsymbol{x}(t))\boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}),$$

where J_{ψ_θ} denotes the Jacobian matrix of ψ_θ.

Since the learned lifting ψ_θ does not generally produce an invariant Koopman subspace, the lifted trajectory $z_\theta(t)$ does not necessarily satisfy the linear dynamics generated by $B_\theta(\boldsymbol{u})$. We therefore introduce the lifted consistency residual

$$\boldsymbol{\rho}_\theta(\boldsymbol{x}, \boldsymbol{u}) := J_{\psi_\theta}(\boldsymbol{x})\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) - B_\theta(\boldsymbol{u})\psi_\theta(\boldsymbol{x}), \tag{10}$$

which measures the lifting inconsistency between the true lifted dynamics and the linear dynamics induced by the learned operator.

Along a time interval $[t_k, t_{k+1})$ with constant input $\boldsymbol{u}_k$, the lifted dynamics satisfies $\dot{z}_\theta(t) = B_\theta(\boldsymbol{u}_k)z_\theta(t) + \boldsymbol{\rho}_\theta(\boldsymbol{x}(t), \boldsymbol{u}_k)$, which yields

$$z_{\theta,k+1} = e^{\Delta t B_\theta(\boldsymbol{u}_k)} z_{\theta,k} + \int_0^{\Delta t} e^{(\Delta t - s)B_\theta(\boldsymbol{u}_k)} \boldsymbol{\rho}_\theta(\boldsymbol{x}(t_k + s), \boldsymbol{u}_k) ds. \tag{11}$$

The local lifted model mismatch is then defined as

$$\boldsymbol{d}_{k+1} := \hat{z}_{k+1} - z_{\theta,k+1}. \tag{12}$$

To obtain an explicit one-step bound, we introduce the following mild boundedness assumption on the learned lifted dynamics over the compact region of interest.

Assumption 1: There exist constants $\bar{B}, \bar{\rho} \geq 0$ such that for all admissible pairs $(\boldsymbol{x}, \boldsymbol{u})$ along the true trajectory, $\|B_\theta(\boldsymbol{u})\| \leq \bar{B}$, $\|\boldsymbol{\rho}_\theta(\boldsymbol{x}, \boldsymbol{u})\| \leq \bar{\rho}$. Moreover, $\boldsymbol{x}(t) \in \mathcal{I}$ on each sampling interval $[t_k, t_{k+1}]$, so that by Lemma 1, $\|z_\theta(t)\| = \|\psi_\theta(\boldsymbol{x}(t))\| \leq \bar{z}$.

Proposition 1 (Local Lifted Model Mismatch Bound): Under Assumption 1, the one-step lifted mismatch satisfies

$$\|\boldsymbol{d}_{k+1}\| \leq \left(e^{\Delta t \bar{B}} - 1 - \Delta t \bar{B}\right)\bar{z} + \Delta t\, e^{\Delta t \bar{B}} \bar{\rho}.$$

Proof: By definition, $\hat{z}_{k+1} = A_\theta(\boldsymbol{u}_k) z_{\theta,k} = (I + \Delta t B_\theta(\boldsymbol{u}_k)) z_{\theta,k}$. Substituting this and (11) into (12), taking norms and using the triangle inequality, we have

$$\begin{aligned}\|\boldsymbol{d}_{k+1}\| \leq &\left\| e^{\Delta t B_\theta(\boldsymbol{u}_k)} - I - \Delta t B_\theta(\boldsymbol{u}_k)\right\| \|z_{\theta,k}\| \\ &+ \int_0^{\Delta t} \left\| e^{(\Delta t - s) B_\theta(\boldsymbol{u}_k)}\right\| \|\boldsymbol{\rho}_\theta(\boldsymbol{x}(t_k + s), \boldsymbol{u}_k)\|\, ds\ .\end{aligned}$$

Using Assumption 1 and the matrix exponential bound $\|e^{\tau B}\| \leq e^{\tau \|B\|}$, we obtain

$$\|\boldsymbol{d}_{k+1}\| \leq \left\| e^{\Delta t B_\theta(\boldsymbol{u}_k)} - I - \Delta t B_\theta(\boldsymbol{u}_k)\right\| \bar{z} + \Delta t\, e^{\Delta t \bar{B}} \bar{\rho}.$$

By the series expansion of the matrix exponential

$$e^{\Delta t \bar{B}} - I - \Delta t \bar{B} = \sum_{j=2}^{\infty} \frac{(\Delta t \bar{B})^j}{j!},$$

and therefore,

$$\begin{aligned}\left\| e^{\Delta t B_\theta(\boldsymbol{u}_k)} - I - \Delta t B_\theta(\boldsymbol{u}_k)\right\| &\leq \sum_{j=2}^{\infty} \frac{\Delta t^j \|B_\theta(\boldsymbol{u}_k)\|^j}{j!} \\ &\leq \sum_{j=2}^{\infty} \frac{(\Delta t \bar{B})^j}{j!} = e^{\Delta t \bar{B}} - 1 - \Delta t \bar{B}.\end{aligned}$$

Combining the above inequalities proves the result. ■

Remark 1: Proposition 1 shows that the one-step lifted mismatch consists of two parts: the discretization-induced error $(e^{\Delta t \bar{B}} - 1 - \Delta t \bar{B})\bar{z}$, which scales as $O(\Delta t^2)$ by Taylor expansion, and the lifted consistency error $\Delta t\, e^{\Delta t \bar{B}} \bar{\rho}$ of order $O(\Delta t)$ with small Δt. In particular, even in the ideal learning scenario where the lifted consistency residual vanishes, i.e., $\bar{\rho} = 0$, the learned model still exhibits a discretization-induced local mismatch of order $O(\Delta t^2)$. This reflects the fact that a continuous-time bilinear Koopman representation does not, in general, yield an exactly bilinear discrete-time update over a finite sampling interval.

C. Finite-Time Prediction Error With Noise

Recall that $z_{\theta,k} := \psi_\theta(\boldsymbol{x}_k)$ denotes the lifted state evaluated along the true trajectory. We first define the lifted state error along the trajectory considering measurement noise as $\boldsymbol{e}_k := \tilde{z}_k - z_{\theta,k}$, $k = 0, 1, \ldots, H$. Since the lifting function explicitly contains the original state, we have $P z_{\theta,k} = \boldsymbol{x}_k$. The state prediction error can therefore be written as $\boldsymbol{\epsilon}_k = \tilde{\boldsymbol{x}}_k - \boldsymbol{x}_k = P \boldsymbol{e}_k$.

The noisy lifted trajectory propagation (5) can be written compactly using the operator A_θ and the definition of $\tilde{\boldsymbol{u}}_k$ as

$$\tilde{z}_{k+1} = A_\theta(\tilde{\boldsymbol{u}}_k)\tilde{z}_k = \left(A_\theta(\boldsymbol{u}_k) + \textstyle\sum_{i=1}^m v_{k,i} K_{\theta,i}\right)\tilde{z}_k. \quad (13)$$

Using the local lifted model mismatch $\boldsymbol{d}_{k+1}$ defined in (12), we have

$$z_{\theta,k+1} = \hat{z}_{k+1} - \boldsymbol{d}_{k+1} = A_\theta(\boldsymbol{u}_k) z_{\theta,k} - \boldsymbol{d}_{k+1}. \quad (14)$$

Combining (13) and (14), the lifted prediction error $\boldsymbol{e}_k = \tilde{z}_k - z_{\theta,k}$ admits the recursion

$$\boldsymbol{e}_{k+1} = A_\theta(\boldsymbol{u}_k)\boldsymbol{e}_k + \textstyle\sum_{i=1}^m v_{k,i} K_{\theta,i}\tilde{z}_k + \boldsymbol{d}_{k+1}. \quad (15)$$

Theorem 1 (Finite-Time Prediction Error Bound): Under Assumption 1, for all $k = 0, 1, \ldots, H-1$, the lifted prediction error satisfies

$$\|\boldsymbol{e}_{k+1}\| \leq \alpha \|\boldsymbol{e}_k\| + \bar{V}\bar{z} + \bar{d}, \quad (16)$$

where $\bar{d}$ denotes the one-step model mismatch bound in Proposition 1, and

$$\bar{V} := \textstyle\sum_{i=1}^m \bar{v}_i \|K_{\theta,i}\|, \quad \alpha := 1 + \Delta t \bar{B} + \bar{V}. \quad (17)$$

Consequently, for all $k = 0, 1, \ldots, H$,

$$\|\boldsymbol{\epsilon}_k\| \leq \alpha^k L_\psi \bar{w} + \tfrac{\alpha^k - 1}{\alpha - 1}(\bar{V}\bar{z} + \bar{d}) \quad (18)$$

Proof: We first derive a one-step bound for the lifted error. Consider the recursion (15), taking norms and using the triangle inequality yields

$$\|\boldsymbol{e}_{k+1}\| \leq \|A_\theta(\boldsymbol{u}_k)\| \|\boldsymbol{e}_k\| + \textstyle\sum_{i=1}^m |v_{k,i}| \|K_{\theta,i}\| \|\tilde{z}_k\| + \|\boldsymbol{d}_{k+1}\|.$$

Using Assumption 1, relation (9) implies

$$\|A_\theta(\boldsymbol{u}_k)\| \leq \|I\| + \Delta t \|B_\theta(\boldsymbol{u}_k)\| \leq 1 + \Delta t \bar{B}.$$

By the definition of $\bar{V}$ and the noise bound $|v_{k,i}| \leq \bar{v}_i$,

$$\textstyle\sum_{i=1}^m |v_{k,i}| \|K_{\theta,i}\| \leq \sum_{i=1}^m \bar{v}_i \|K_{\theta,i}\| = \bar{V}.$$

Using $\boldsymbol{e}_k = \tilde{z}_k - z_{\theta,k}$, we have

$$\|\tilde{z}_k\| \leq \|\boldsymbol{e}_k\| + \|z_{\theta,k}\| \leq \|\boldsymbol{e}_k\| + \bar{z},$$

Substituting the above inequalities into the one-step bound and using the definition of α in (17), we obtain

$$\begin{aligned}\|\boldsymbol{e}_{k+1}\| &\leq (1 + \Delta t \bar{B})\|\boldsymbol{e}_k\| + \bar{V}(\|\boldsymbol{e}_k\| + \bar{z}) + \bar{d} \\ &= \alpha \|\boldsymbol{e}_k\| + \bar{V}\bar{z} + \bar{d},\end{aligned}$$

which proves the recursive bound (16).

By Lemma 2 and the noise bound $\|\boldsymbol{w}\| \leq \bar{w}$, the initial lifted error is bounded by

$$\|\boldsymbol{e}_0\| = \|\psi_\theta(\tilde{\boldsymbol{x}}_0) - \psi_\theta(\boldsymbol{x}_0)\| \leq L_\psi \|\tilde{\boldsymbol{x}}_0 - \boldsymbol{x}_0\| \leq L_\psi \bar{w}.$$

Applying (16) recursively and using $\|\boldsymbol{e}_0\| \leq L_\psi \bar{w}$ gives

$$\|\boldsymbol{e}_k\| \leq \alpha^k L_\psi \bar{w} + \textstyle\sum_{j=0}^{k-1} \alpha^j (\bar{V}\bar{z} + \bar{d}).$$

Evaluating the geometric sum yields

$$\|\boldsymbol{e}_k\| \leq \alpha^k L_\psi \bar{w} + \tfrac{\alpha^k - 1}{\alpha - 1}(\bar{V}\bar{z} + \bar{d}).$$

Finally, since $\boldsymbol{\epsilon}_k = P\boldsymbol{e}_k$ and $P = [0\ \boldsymbol{I}_n\ \boldsymbol{0}]$ is a coordinate projection, we have $\|P\| = 1$, hence

$$\|\boldsymbol{\epsilon}_k\| = \|P\boldsymbol{e}_k\| \leq \|P\| \|\boldsymbol{e}_k\| \leq \|\boldsymbol{e}_k\|.$$

Combining this with the lifted state error bound completes the proof. ■

Corollary 1 (Terminal Prediction Error): Fix a terminal time $T > 0$, and let $\Delta t \in [\underline{\Delta}, \bar{\Delta}]$ satisfy $H = T/\Delta t \in \mathbb{N}$. Then there exist nonnegative constants λ, c_1, c_2, c_3, such that

$$\|\boldsymbol{\epsilon}_H\| \leq e^{\lambda T} L_\psi \bar{w} + \tfrac{e^{\lambda T} - 1}{\lambda}\left(\tfrac{c_1}{\Delta t} + c_2 + c_3 \Delta t\right). \quad (19)$$

Proof: By Remark 1, there exist nonnegative constants c_2, c_3 for Δt in the bounded interval such that $\bar{d} \le c_2\Delta t + c_3\Delta t^2$, therefore

$$\bar{V}\bar{z} + \bar{d} \le c_1 + c_2\Delta t + c_3\Delta t^2, \quad c_1 := \bar{V}\bar{z}.$$

Let $\lambda = \bar{B} + \bar{V}/\underline{\Delta}$, so that $\alpha = 1 + \Delta t\bar{B} + \bar{V} \le 1 + \lambda\Delta t$, then

$$\alpha^H \le (1 + \lambda\Delta t)^H \le e^{\lambda H\Delta t} = e^{\lambda T},$$

$$\tfrac{\alpha^H - 1}{\alpha - 1} = \sum_{j=0}^{H-1} \alpha^j \le \sum_{j=0}^{H-1} e^{j\lambda\Delta t} = \frac{e^{\lambda T} - 1}{e^{\lambda\Delta t} - 1} \le \frac{e^{\lambda T} - 1}{\lambda\Delta t}.$$

Substituting the above inequalities into Theorem 1, we have

$$\|\boldsymbol{\epsilon}_H\| \le e^{\lambda T} L_\psi \bar{w} + \tfrac{e^{\lambda T} - 1}{\lambda}\left(\tfrac{c_1}{\Delta t} + c_2 + c_3\Delta t\right).$$

■

Remark 2: Corollary 1 makes explicit the role of the sampling period in finite-time prediction. In particular, the term $c_1/\Delta t$ reflects the accumulation of per-step noise-induced error over $H = T/\Delta t$, the term $c_3\Delta t$ captures the growth of discretization-induced mismatch as Δt increases, and the constant term c_2 corresponds to the cumulative effect of the $O(\Delta t)$ component in the one-step mismatch. Although the sampling period minimizing this upper bound need not coincide with that yielding the best empirical prediction performance, the corollary reflects the following trade-off: decreasing Δt improves the one-step approximation accuracy, while increasing Δt reduces the number of recursive error injections over a fixed finite horizon. Consequently, the prediction error need not vary monotonically with Δt, and an intermediate sampling period may perform best, as observed in our experiments.

V. Simulation Results

In this section, we present simulation studies to validate the proposed prediction error analysis. We compare the certified bound with the actual prediction error under noisy initial states and inputs, and investigate how the sampling period affects prediction accuracy through the trade-off between discretization accuracy and multi-step error accumulation.

A. Prediction Bound Evaluation

We use a cart-pole system and a 2D quadrotor as simulation examples [20]. For data generation, the initial state and control of the cart-pole system is sampled uniformly with cart position $p \in [-1, 1]$, pole angle $q \in [-\pi/4, \pi/4]$, corresponding velocities $\dot{p}, \dot{q} \in [-0.1, 0.1]$, and the horizontal force $u \in [-1, 1]$. The 2D quadrotor is sampled with horizontal and vertical positions $x, z \in [-0.75, 0.75]$, pitch angle $\theta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$, corresponding velocities $\dot{x}, \dot{z}, \dot{\theta} \in [-4, 4]$, and rotor thrusts $u_1, u_2 \in [0, 3]$. The sampling period is set to $\Delta t = 0.1$ s and the terminal time is $T = 2$ s, yielding a prediction horizon of 20 steps. In total, 500 trajectories are generated and split into training, validation, and test sets with proportions 80%, 10%, and 10%, respectively.

To investigate the effects of network structure, we apply a 3-layer lifting network (width 64, output dimension 32) to the cart-pole system, and a 5-layer lifting network (width 128, output dimension 64) to the 2D quadrotor. The models are trained by minimizing a standard loss function

$$\mathcal{L}(\theta) = \tfrac{1}{N}\sum_{j=1}^{N}\sum_{k=0}^{H-1}\left\|\hat{\boldsymbol{x}}_{k+1}^{(j)} - \boldsymbol{x}_{k+1}^{(j)}\right\|^2 + w\|\theta\|^2,$$

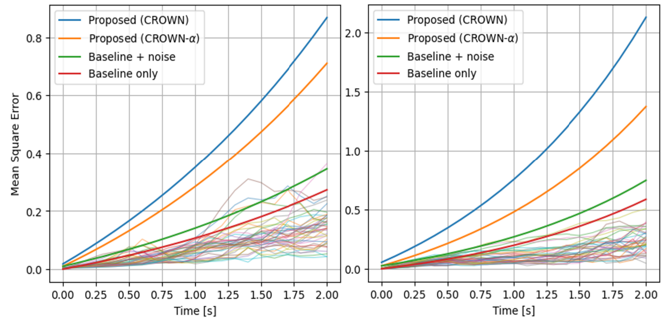

Fig. 1. Comparison of theoretical bounds and actual prediction errors (left: cart-pole system, right: 2D quadrotor). Thick lines: proposed and baseline bounds. Thin lines: actual prediction errors from test trajectories with measurement noise.

where N is the number of training trajectories, $\hat{\boldsymbol{x}}_{k+1}^{(j)}$ is the one-step state prediction generated by the neural Koopman model for the jth trajectory, $\boldsymbol{x}_{k+1}^{(j)}$ is the corresponding ground-truth state, and $w = 10^{-4}$ is the regularization coefficient. The models are trained using the Adam optimizer with a learning rate 10^{-3}.

For bound evaluation, the box region $\mathcal{I}$ is constructed from the maximum absolute value attained over all trajectories. The constant $\bar{B}$ is obtained directly from the norm of the trained linear operator. To obtain the upper bound $\bar{\rho}$, we construct a finite grid $\mathcal{G}$ covering $\mathcal{X} \times \mathcal{U}$ with a grid radius $\epsilon > 0$. By leveraging Lemma 2 and the composition rules, the certified Lipschitz constant L_ρ of the the lifting inconsistency $\boldsymbol{\rho}_\theta$ over $\mathcal{X} \times \mathcal{U}$ can be computed. Consequently, the lifting inconsistency is strictly bounded by $\bar{\rho} = \max_{(x_i,u_i)\in\mathcal{G}} \boldsymbol{\rho}_\theta(x_i, u_i) + L_\rho\epsilon$, which completely eliminates any sampling uncertainty.

For comparison, we consider four methods: 1) the proposed method with CROWN certification [17]. 2) The proposed method with CROWN-α certification, which provides a tighter bound propagation by optimizing auxiliary dual variables in convex relaxation [18]. 3) A baseline obtained by replacing the one-step mismatch bound $\bar{d}$ with an error bound surrogate derived from the operator approximation bound in [13] at a confidence level of 95%. 4) The same baseline without considering recursive measurement noise propagation. Figure 1 compares the the resulting bounds with the actual prediction errors under a noise level at 1%. The proposed method provides valid upper bounds for both systems, whereas the baseline bounds are violated due to neglecting the lifting inconsistency. Moreover, comparing the two baseline variants shows that ignoring recursive noise propagation further underestimates the multi-step prediction error. Comparison across the two examples shows that standard CROWN yields more conservative bounds in the 2D quadrotor example due to looser bound propagation for larger and deeper lifting networks. Replacing standard CROWN with CROWN-α produces tighter certified bounds in Lemma 2 and consequently improves the overall prediction bound tightness. This observation suggests that integrating more advanced neural network certification techniques could further reduce conservativeness. Overall, these results demonstrate the necessity of explicitly accounting for both lifting inconsistency and measurement noise

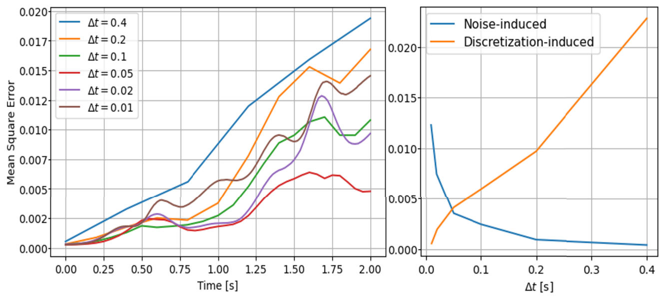

Fig. 2. Comparison of prediction errors (left) and evaluation of terminal errors (right) for different sampling periods.

propagation, as well as the direct influence of neural certification quality on the prediction bounds.

B. Sampling Period Trade-Off

To investigate the effects of the sampling period Δt, we use the cart-pole system as an example and compare different neural Koopman models trained on datasets generated using $\Delta t \in \{0.4, 0.2, 0.1, 0.05, 0.02, 0.01\}$. For all sampling periods, the data distribution and training setup are kept the same as in the previous subsection. Each neural Koopman model is trained to convergence with the same data size and optimization settings, so that the comparison reflects the intrinsic effect of the sampling period rather than differences in training quality.

Figure 2 compares the prediction errors for different sampling periods. To further analyze the trade-off, the discretization-induced term of the terminal prediction errors is evaluated, along with the noise-induced term obtained via comparison with noise-free predictions. The numerical results shown in Figure 2 reflect the corresponding trend of discretization-induced mismatch and measurement noise accumulation derived in Corollary 1. Specifically, with a large $\Delta t = 0.4$s, the error is dominated by the discretization mismatch, which accumulates at a rate of $O(\Delta t)$ as the higher-order terms in the continuous flow cannot be adequately captured by the bilinear approximation. Conversely, decreasing Δt to a small value (0.01s) does not yield the lowest terminal error as the high-frequency recursive injections of measurement noise blow up at a rate of $O(1/\Delta t)$. As observed in Figure 2, an intermediate sampling period around $\Delta t = 0.05$s achieves the best performance with an optimal balance between these two factors. This insight provides practical guidelines for selecting the optimal sampling period in learning-based Koopman modeling for noise-aware applications.

VI. Conclusion

In this work, we have established an explicit finite-time prediction error bound for neural bilinear Koopman models subjected to measurement noise. Based on the certified properties for the lifting network, our framework quantifies the joint effects of discretization mismatches, lifting inconsistency, and recursive noise propagation through structural decomposition. The theoretical derivation further reveal a non-monotonic terminal prediction error trade-off between discretization and noise-induced errors with respect to the sampling period. Numerical results illustrate the practical relevance of the derived bounds and support the theoretical findings. Beyond theoretical analysis, these findings provide a foundation for noise-aware Koopman model predictive control design and offer practical guidelines for optimal sampling period selection in data-driven control.

References

[1] S. L. Brunton, M. Budišić, E. Kaiser, and J. N. Kutz, "Modern Koopman theory for dynamical systems," *SIAM Rev.*, vol. 64, no. 2, pp. 229–340, May 2022.

[2] L. Shi et al., "Koopman operators in robot learning," *IEEE Trans. Robot.*, vol. 42, pp. 1088–1107, 2026.

[3] R. Strässer, K. Worthmann, I. Mezić, J. Berberich, M. Schaller, and F. Allgöwer, "An overview of Koopman-based control: From error bounds to closed-loop guarantees," *Annu. Rev. Control*, vol. 61, 2026, Art. no. 101035. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1367578825000495

[4] N. Takeishi, Y. Kawahara, and T. Yairi, "Learning Koopman invariant subspaces for dynamic mode decomposition," in *Proc. Adv. Neural Inf. Process. Syst.* 30, Dec. 2017, pp. 1130–1140.

[5] 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, Interdiscipl. J. Nonlinear Sci.*, vol. 27, no. 10, Oct. 2017, Art. no. 103111.

[6] B. Lusch, J. N. Kutz, and S. L. Brunton, "Deep learning for universal linear embeddings of nonlinear dynamics," *Nature Commun.*, vol. 9, no. 1, p. 4950, Nov. 2018.

[7] Y. Han, W. Hao, and U. Vaidya, "Deep learning of Koopman representation for control," in *Proc. 59th IEEE Conf. Decis. Control (CDC)*, Dec. 2020, pp. 1890–1895.

[8] H. Shi and M. Q.-H. Meng, "Deep Koopman operator with control for nonlinear systems," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 7700–7707, Jul. 2022.

[9] D. Bruder, X. Fu, and R. Vasudevan, "Advantages of bilinear koopman realizations for the modeling and control of systems with unknown dynamics," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 4369–4376, Jul. 2021.

[10] C. Folkestad, S. X. Wei, and J. W. Burdick, "KoopNet: Joint learning of Koopman bilinear models and function dictionaries with application to quadrotor trajectory tracking," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, Philadelphia, PA, USA, May 2022, pp. 1344–1350.

[11] D. Zhao, B. Li, F. Lu, J. She, and S. Yan, "Deep bilinear Koopman model predictive control for nonlinear dynamical systems," *IEEE Trans. Ind. Electron.*, vol. 71, no. 12, pp. 16077–16086, Dec. 2024.

[12] M. Wang, X. Lou, and B. Cui, "Deep bilinear Koopman realization for dynamics modeling and predictive control," *Int. J. Mach. Learn. Cybern.*, vol. 15, no. 8, pp. 3327–3352, Aug. 2024.

[13] F. Nüske, S. Peitz, F. Philipp, M. Schaller, and K. Worthmann, "Finite-data error bounds for Koopman-based prediction and control," *J. Nonlinear Sci.*, vol. 33, no. 1, p. 14, Feb. 2023.

[14] R. Strässer, M. Schaller, J. Berberich, K. Worthmann, and F. Allgöwer, "Kernel-based error bounds of bilinear Koopman surrogate models for nonlinear data-driven control," *IEEE Control Syst. Lett.*, vol. 9, pp. 1892–1897, 2025.

[15] W. Hao, D. Upadhyay, and S. Mou, "Deep Koopman learning using noisy data," *Trans. Mach. Learn. Res.*, 2025. [Online]. Available: https://openreview.net/forum?id=j6Rm6T2lFU

[16] S. A. Sakib and S. Pan, "Learning noise-robust stable Koopman operator for control with Hankel DMD," *IEEE Trans. Control Syst. Technol.*, vol. 34, no. 1, pp. 139–153, Jan. 2026.

[17] H. Zhang, T.-W. Weng, P.-Y. Chen, C.-J. Hsieh, and L. Daniel, "Efficient neural network robustness certification with general activation functions," in *Proc. Adv. Neural Inf. Process. Syst. (NuerIPS)*, vol. 31, Dec. 2018.

[18] K. Xu et al., "Fast and complete: Enabling complete neural network verification with rapid and massively parallel incomplete verifiers," in *Proc. Int. Conf. Learn. Represent.*, 2021. [Online]. Available: https://openreview.net/forum?id=nVZtXBI6LNn

[19] Z. Shi, Y. Wang, H. Zhang, J. Z. Kolter, and C.-J. Hsieh, "Efficiently computing local Lipschitz constants of neural networks via bound propagation," in *Proc. Adv. Neural Inf. Process. Syst.* 35, 2022, pp. 2350–2364.

[20] K. Long, J. Cortés, and N. Atanasov, "Certifying stability of reinforcement learning policies using generalized Lyapunov functions," in *Proc. Adv. Neural Inf. Process. Syst.*, 2025. [Online]. Available: https://openreview.net/forum?id=N67DlqK5C4-eId=7tOkUSRems