2020 IEEE/ASME International Conference on
Advanced Intelligent Mechatronics (AIM), Boston, USA
(Virtual Conference), July 6-9, 2020

Discrete System Linearization using Koopman Operators for Predictive Control and Its Application in Nano-positioning

Shengwen Xie[1] and Juan Ren[1,†]

Abstract— **Predictive control techniques have been broadly used in control of nonlinear nano-positioning systems (e.g., piezo actuators). One common approach to control such systems for precision positioning is to model the system dynamics using a nonlinear model and linearize it using Taylor series, and then apply model predictive control (MPC). However, the control bandwidth and the control performance are quite limited as the obtained linear system is only guaranteed to be accurate within small neighborhood of the linearization point. To address this issue, we propose to linearize the nonlinear system model using Koopman operators and then use the obtained linear parameter-varying model for predictive control. This linearization scheme can significantly decrease the overall approximation error within the MPC prediction horizon, and thus, lead to improved tracking performance. The proposed approach was validated through trajectory tracking of piezo actuators in simulation.**

I. Introduction

Predictive control techniques have been broadly used in control of nano-positioning systems such as piezo actuators (PEAs). The key in the success of predictive control is the accuracy of the system model used. Lately, recurrent neural networks (RNNs) have been shown effective in modeling the nonlinear dynamics of PEA systems, the nonlinearities of which are mostly induced by hysteresis and creep effects [1]. Although the modeling accuracy over broad bandwidth and/or amplitude range has been improved notably compared to other modeling approaches, the nonlinearities itself makes it challenging to design computationally efficient controllers. While model predictive control (MPC) can be extended to the nonlinear case, the resulted optimization problem is non-convex and thus intractable in general [1], [2].

This issue can be avoided through linearizing the original nonlinear system. Taylor series has been broadly used in nonlinear system linearization [2]. However, the disadvantage is that the approximation accuracy is only guaranteed within small neighborhood of the linearization point. Therefore, the modeling uncertainties will surge as the prediction horizon increases, and then limit the performance of MPC. Feedback linearization and flatness can also be used for linearization but they are not effective for general nonlinear systems [3], [4]. Other options include Carleman linearization, "polyflow", and Koopman operator approach [5]–[7]. Among these, only Koopman approach is data-driven [8], [9]. Koopman operator approach uses a finite number of functions (i.e., observations) to form the state of the linearized model, and is not restricted by the form of the nonlinear model. Data-driven Koopman model predictive control has been proposed for controlling nonlinear partial differential equations, where the nonlinear dynamics is linearized [10]–[12].

In this work, we propose to achieve precision positioning control using Koopman operator approach. Specifically, a discrete nonlinear system represented by a RNN is linearized using data-driven Koopman operator approach, and a predictive controller is designed based on the linearized model [10]. Instead of using a linear model to approximate the system nonlinear dynamics over the entire control process, only the dynamics in the future N_p (i.e., prediction horizon) steps is approximated (i.e., linearized). Since the ultimate goal is to realize precision positioning in real time, the order of the linear system is set to be small for the sake of computation efficiency. We show that when the observation functions are linear to the original state, Koopman approach is similar to performing system identification locally. Different from the approach in [10], we also extend Koopman approach to a bilinear model and both approaches are evaluated through simulations. Note that even though Koopman approach can approximate the original dynamics more accurately than the Taylor series, the linearization accuracy decreases as the increment of prediction horizon and/or the frequency of the input. This issue can be mitigated through introducing a second linear model in the predictive controller. Specifically, the first linear model (generated with Koopman approach) will be used to compute the optimal inputs which provide the anchor for identifying the second model, which is also generated using Koopman approach. For demonstration, the control performance of using single linear model and two linear models are compared with that of the Taylor approach in simulation.

II. Preliminaries

A. Koopman operators

Consider a system represented by a discrete model as

$$x_{k+1} = f(x_k), \quad x_k \in \mathbb{R}^n. \tag{1}$$

Define the system observation or output $y_k \in \mathbb{R}^m$ as $y_k = g(x_k)$. The Koopman operator $\mathcal{K}$ is defined as

$$\mathcal{K}g(x_k) \triangleq g \circ f(x_k) = g(x_{k+1}), \quad \circ : composition. \tag{2}$$

Therefore, the linearity of Koopman operator can be verified by

$$\mathcal{K}(\alpha g(x_k) + \beta g(x_k)) = \alpha \mathcal{K} g(x_k) + \beta \mathcal{K} g(x_k). \tag{3}$$

[1]S. Xie and [1]J. Ren are with the Department of Mechanical Engineering, Iowa State University, Ames, IA 50011, USA swxie@iastate.edu,juanren@iastate.edu
† Corresponding author.

The Koopman eigenfunction ϕ satisfies $\mathcal{K}\phi = \lambda\phi$ where λ is the corresponding eigenvalue. The evolution of the outputs can be expressed as a linear combination of infinite number of eigenfunctions. In practice, only a finite number of eigenfunctions will be used [13]. To analyze the non-autonomous system, the Koopman operator is extended to system with exogenous input as shown in Eq. (4) [10].

$$x_{k+1} = f(x_k, u_k) \tag{4}$$

Let $(u_i)_k^\infty$ denote the input sequence starting from time k. Suppose $x_k \in \mathcal{X}$ and $u_k \in \mathbb{U}$, define $\mathcal{L} = \{(u_i)_k^\infty | u_i \in \mathbb{U}\}$, the state space $\mathcal{X}$ is extended to $\mathcal{X} \times \mathcal{L}$. Accordingly, the Koopman operator $\mathcal{K}$ for the system Eq. (4) is defined as

$$\mathcal{K}g(x_k, (u_i)_k^\infty) \triangleq g(f(x_k, u_k), (u_i)_{k+1}^\infty). \tag{5}$$

With the newly defined Koopman operator (Eq. (5)), the established Koopman operator theories can be applied for the non-autonomous system Eq. (4).

B. RNN-based nonlinear dynamics

RNN can be used to capture complex dynamics in theory given enough number of parameters. In this work, we consider the RNN represented by Eq. (6) which is used to model the dynamics of piezo actuators (PEAs),

$$\begin{aligned} x_{k+1} &= \tanh(W_1 x_k + B_2 + B_1 u_k) \\ y_k &= W_2 x_k + B_3 \end{aligned}, \tag{6}$$

where $x_k \in \mathbb{R}^{N\times 1}$, u_k, and y_k are the system state, input and output at the sampling instant k, respectively [1]. The nonlinearity of Eq. (6) is mostly induced by hysteresis which is both frequency- and amplitude-dependent. Such nonlinearity makes it very challenging to design controllers with large frequency and/or amplitude bandwidth and high accuracy.

C. Linearization with Taylor series

Eq. (6) can be linearized with Taylor series. Note that Taylor approximation linearizes the system around a chosen fixed point and only guarantees the accuracy around it. Let $h(x) = \tanh(x)$. At the sampling instant k, Eq. (6) can be linearized at the fixed point $(\bar{x}_k, \bar{u}_k)$ as

$$\begin{aligned} x_{k+1} &= h(W_1\bar{x}_k + B_2 + B_1\bar{u}_k) + \frac{\partial h}{\partial x_k}\Big|_{\substack{x_k=\bar{x}_k \\ u_k=\bar{u}_k}} (x_k - \bar{x}_k) \\ &\quad + \frac{\partial h}{\partial u_k}\Big|_{\substack{x_k=\bar{x}_k \\ u_k=\bar{u}_k}} (u_k - \bar{u}_k) \\ &= A_k x_k + B_k u_k + M_k \end{aligned} \tag{7}$$

where $A_k = \frac{\partial h}{\partial x_k}\Big|_{\substack{x_k=\bar{x}_k \\ u_k=\bar{u}_k}}$, $B_k = \frac{\partial h}{\partial u_k}\Big|_{\substack{x_k=\bar{x}_k \\ u_k=\bar{u}_k}}$, and $M_k = h(W_1\bar{x}_k + B_2 + B_1\bar{u}_k) - \frac{\partial h}{\partial x_k}\Big|_{\substack{x_k=\bar{x}_k \\ u_k=\bar{u}_k}} \bar{x}_k - \frac{\partial h}{\partial u_k}\Big|_{\substack{x_k=\bar{x}_k \\ u_k=\bar{u}_k}} \bar{u}_k$. By assuming that A_k, B_k, and M_k remain fixed for the future N_p (i.e., prediction horizon) steps, Eq. (7) can be used to predict the future inputs with unknowns $[u_{k+1}, u_{k+1}, \cdots, u_{k+N_p}]$.

III. Koopman Operator Approach

A. Linearization based on Koopman operators

The EDMD algorithm proposed in [14] is used to linearize Eq. (6). Let $z_k = [g_1(x_k), g_2(x_k), \cdots, g_m(x_k)]^T$, where g_is are observations. Suppose that the Koopman operator defined in Eq. (5) is in the span of $[z_k^T, u_k]^T$. Then we have

$$z_{k+1} = \begin{bmatrix} A & B \end{bmatrix} \begin{bmatrix} z_k \\ u_k \end{bmatrix} = Az_k + Bu_k. \tag{8}$$

At the sampling instant k, the current state x_k and input u_k are known. The goal is to use Eq. (8) to model the dynamics in the future N_p steps, where N_p is the prediction horizon of the predictive controller. Since the inputs for the future N_p steps are unknown, one can try different input sequences $(u_i)_k^{k+N_p}$. For the following matrix,

$$\mathcal{U} = \begin{bmatrix} u_{1,k} & u_{1,k+1} & \cdots & u_{1,k+N_p} \\ u_{2,k} & u_{2,k+1} & \cdots & u_{2,k+N_p} \\ \vdots & \vdots & \ddots & \vdots \\ u_{l,k} & u_{l,k+1} & \cdots & u_{l,k+N_p} \end{bmatrix}, \tag{9}$$

each row of $\mathcal{U}$ is a possible sequence of inputs for the next N_p steps. With the ith row $[u_{i,k}, u_{i,k+1}, \cdots, u_{i,k+N_p}]$ as inputs to the nonlinear system Eq. (6), we can obtain the future states $[x_{i,k+1}, x_{i,k+2}, \cdots, x_{i,k+N_p}]$. Then $z_{i,k}$ can be constructed as follows.

$$\begin{aligned} z_{i,k} &= [g_1(x_{i,k}), \cdots, g_m(x_{i,k})]^T \\ z_{i,k+1} &= [g_1(x_{i,k+1}), \cdots, g_m(x_{i,k+1})]^T \end{aligned} \tag{10}$$

By Eq. (8), we have $z_{i,k+1} = Az_{i,k} + B\mathcal{U}_{i,k}$, where $\mathcal{U}_{i,k}$ is the entry (i,k) of $\mathcal{U}$. It follows that

$$\begin{aligned} Z_i^+ &= [z_{i,k+1}, \cdots, z_{i,k+N_p}] \\ &= A[z_{i,k}, \cdots, z_{i,k+N_p-1}] + B[\mathcal{U}_{i,k+1}, \cdots, \mathcal{U}_{i,K+N_p-1}] \\ &= \begin{bmatrix} A & B \end{bmatrix} \begin{bmatrix} Z_i \\ \mathcal{U}_i \end{bmatrix} \end{aligned} \tag{11}$$

where $\mathcal{U}_i$ is the ith row of $\mathcal{U}$. For each $i = 1, 2, \cdots, m$, a similar equation can be derived, and they can be combined as

$$Z^+ = \begin{bmatrix} Z_1^+ & Z_2^+ & \cdots & Z_l^+ \end{bmatrix} = \begin{bmatrix} A & B \end{bmatrix} \begin{bmatrix} Z_1 & Z_2 & \cdots & Z_l \\ \mathcal{U}_1 & \mathcal{U}_2 & \cdots & \mathcal{U}_l \end{bmatrix} = \begin{bmatrix} A & B \end{bmatrix} Z. \tag{12}$$

One can solve the following least-square optimization problem to obtain A and B.

$$\min_{A,B} || \ Z^+ - [A \ B]Z \ || \tag{13}$$

The solution will be $[A, B] = Z^+ Z^{-1}$ with Z^{-1} as the pseudoinverse of Z.

If we assume that the Koopman operator defined in Eq. (5) is in the span of $[z_k^T, u_k, z_k^T u_k]^T$, then the nonlinear dynamics will be approximated with the following bilinear form.

$$z_{k+1} = \begin{bmatrix} A & B & \bar{B} \end{bmatrix} \begin{bmatrix} z_k \\ u_k \\ z_k u_k \end{bmatrix} = Az_k + Bu_k + \bar{B} z_k u_k. \tag{14}$$

To obtain the parameters $(A, B, \bar{B})$, Eq. (12) can be modified as

$$\begin{aligned} Z^+ &= \begin{bmatrix} Z_1^+ & Z_2^+ & \cdots & Z_l^+ \end{bmatrix} = \begin{bmatrix} A & B & \bar{B} \end{bmatrix} \begin{bmatrix} Z_1 & Z_2 & \cdots & Z_l \\ \mathcal{U}_1 & \mathcal{U}_2 & \cdots & \mathcal{U}_l \\ T_1 & T_2 & \cdots & T_l \end{bmatrix} \\ &= \begin{bmatrix} A & B & \bar{B} \end{bmatrix} Z \end{aligned} \tag{15}$$

where T_i is the term due to $z_k u_k$ and can be computed as $T_i(p,q) = Z_i(p,q) \times \mathcal{U}_i(j)$. Accordingly, Eq. (13) is changed to

$$\min_{A,B,\bar{B}} || \; Z^+ - [A \; B \; \bar{B}] Z \; ||. \tag{16}$$

Stabilization of the bilinear system has been studied in [15], [16]. However, Eq. (14) is not suitable for designing predictive controllers due to the nonlinear term, even though the approximation accuracy might be improved. Note that if $g_1(x_k) = W_2 x_k + B_3$, then dynamics of the linearized model in the future N_p steps is as follows.

$$\begin{aligned} z_{k+1} &= Az_k + Bu_k \\ y_k &= [1 \; 0 \; \cdots \; 0] z_k = Cz_k \end{aligned} \tag{17}$$

Remark 1. One special case is that the g_i is a linear function of x_k, e.g., $z_k = Mx_k$ for some invertible matrix M. Then Eq. (8) is equivalent to

$$x_{k+1} = M^{-1}AMx_k + M^{-1}Bu_k \quad . \tag{18}$$

It seems that Eq. (18) is similar to Taylor approximation. However, it is still possible that Eq. (18) is better than Taylor approximation as the latter is optimal around the linearization point, while Eq. (18) is optimal in the sense that the prediction error over the future N_p steps is minimum. Mathematically, for any other linear approximation $\mathcal{M}$ in the space of $[x_k, u_k]$, there exists $\varepsilon > 0$ such as that Taylor approximation is more accurate in the open ball $B_\varepsilon([x_k, u_k])$ than $\mathcal{M}$.

In other words, the model Eq. (18) may sacrifice some accuracy at the linearization point to achieve overall accuracy, which is illustrated by the simulation results in Fig. 1: for the dynamical system $x_{k+1} = x_k^2$, clearly, $L1$ is optimal locally at point A1 and $L2$ is optimal for the range between A1 and A2.

Remark 2. Another important factor that may affect the approximation accuracy is the generation of $\mathcal{U}$. The nonlinearity of the system Eq. (6) depends on the input u_k, therefore, the range of u_k defined as $RG = \max_{k+1 \le i \le k+N_p} u_i - \min_{k+1 \le i \le k+N_p} u_i$ may determine how local the linear approximation is. In this work, $\mathcal{U}$ is randomly generated with the RG changes adaptively according to the trajectory to be tracked, i.e., $\mathcal{U}_{i,k}$ is uniformly distributed in the range $[-RG + u_k, RG + u_k]$. In particular, RG is set to be proportional to $\bar{RG}$ i.e., $RG = \rho_1 \bar{RG}$, where $\bar{RG} = \max_{k+1 \le i \le k+N_p} r_i - \min_{k+1 \le i \le k+N_p} r_i$ is the range of the reference signal r_k to be tracked in the future N_p steps. Ideally, ρ_1 should be determined by the inversion model of the system and can be estimated if the inversion model is unavailable. This strategy ensures that Koopman approach outperforms Taylor approximation for both high-frequency and low-frequency inputs.

B. Predictive Control

In this section, MPC is reviewed first. Then how to incorporate two linear models to enhance the controller performance is presented.

At the sampling instance k, suppose Eq. (6) is linearized with Koopman approach and the obtained linear model is Eq. (8). Let $U = [u_{k+1}, u_{k+2}, \cdots, u_{k+N_p}]^T$, $\Delta U = [u_{k+1} - u_k, u_{k+2} - u_{k+1}, \cdots, u_{k+N_p} - u_{k+N_p-1}]^T$, and $\mathbf{1}_n = [1, 1, \cdots, 1]^T$. The predicted N_p future outputs Y of the linear system Eq. (8) are

$$\begin{aligned} Y &= Gz_k + HU + Fu_k \\ &= Gz_k + H(S\Delta U + \mathbf{1}u_k) + Fu_k, \\ &= Gz_k + HS\Delta U + (F + H\mathbf{1})u_k \end{aligned} \tag{19}$$

with

$$Y = \begin{bmatrix} y_{k+1} \\ y_{k+2} \\ \vdots \\ y_{k+N_p} \end{bmatrix}_{N_p \times 1}, \; G = \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^{N_p} \end{bmatrix}_{N_p \times 1}, \; U = \begin{bmatrix} u_{k+1} \\ u_{k+2} \\ \vdots \\ u_{k+N_p} \end{bmatrix}_{N_p \times 1}$$

$$H = \begin{bmatrix} 0 & 0 & \dots & 0 \\ CB & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{N_p-2}B & CA^{N_p-3}B & \dots & 0 \end{bmatrix}_{N_p \times N_p}$$

$$S = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 1 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \dots & 1 \end{bmatrix}, \; F = \begin{bmatrix} CB \\ CAB \\ \vdots \\ CA^{N_p-1}B \end{bmatrix}_{N_p \times 1}.$$

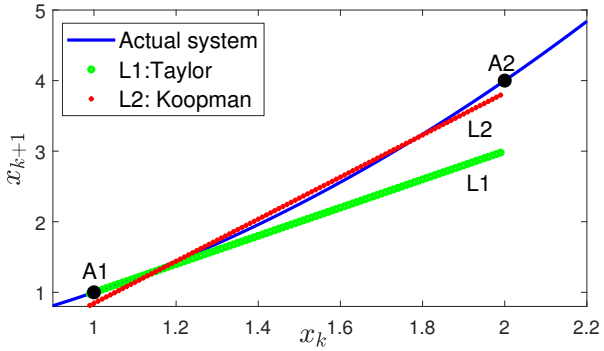

Fig. 1: Comparison of Taylor approximation and Koopman approach in simulation.

The cost function to be minimized for MPC is

$$\begin{aligned} J &= (Y-R_k)^T(Y-R_k)+\rho\Delta U^T\Delta U \\ &= \Delta U^T(S^TH^THS+\rho\mathbf{I})\Delta U+2\Delta U^TS^TH^TE+E^TE \end{aligned}, \tag{20}$$

where $R_k=[r_{k+1},r_{k+2},\cdots,r_{k+N_p}]^T$, $\mathbf{I}$ is the identity matrix, $\rho>0$, and $E=Gx_k+(F+H\mathbf{1})u_k-R_k$. The objective is to minimize J under certain constraints, which is formulated as the following optimization problem

$$\begin{aligned} \min_{\Delta U} \quad & J=\frac{1}{2}\Delta U^T\Psi\Delta U+\Phi^T\Delta U \\ \text{s.t.} \quad & |\Delta U|\le\bar{U} \end{aligned}, \tag{21}$$

where $\Psi=S^TH^THS+\rho\mathbf{I}$ and $\Phi=S^TH^TE$.

For Eq. (6), the nonlinearity depends on u_k, if the prediction horizon remains the same and the frequency of u_k increases, the modeling accuracy of Eq. (8) will decrease resulting in downgraded performance in trajectory tracking. One possible way to address this issue is to add another linear model to approximate the dynamics, i.e., use two linear models to approximate the dynamic of the system for the future N_p steps. The computation consumption may increase since the same optimization problem Eq. (21) has to be solved twice. The detailed algorithm is shown below.

First, at the sampling instant k, obtain the linear model with Koopman approach and solve the optimization problem Eq. (21). Then the solved optimal N_p inputs are $U^*=[u^*_{k+1},u^*_{k+2},\cdots,u^*_{k+N_p}]$, and choose the step where the second linear model starts working. For instance, we can select the starting point at u^*_{k+q} with $q=N_p/2$. Use the same procedure to obtain the second model with the starting point at $[x_{k+q},u^*_{k+q}]$. Therefore, the dynamics in the future N_p steps can be described with the following model.

$$z_{i+1}=\begin{cases} A_1z_i+B_1u_i & \text{if } i=k,k+1,\cdots,k+q \\ A_2z_i+B_2u_i & \text{if } i=k+q+1,k+q+2,\cdots,k+N_p \end{cases} \tag{22}$$

Accordingly, the parameters G, F, and H should be modified as follows. The algorithm which incorporates the two linear models is summarized in **Algorithm 1**.

$$G=\begin{bmatrix} CA_1 \\ \vdots \\ CA_1^q \\ CA_2A_1^q \\ \vdots \\ CA_2^{N_p-q}A_1^q \end{bmatrix}_{N_p\times 1} \qquad F=\begin{bmatrix} CB_1 \\ \vdots \\ CA_1^{q-1}B_1 \\ CA_2A_1^{q-1}B_1 \\ \vdots \\ CA_2^{N_p-q}A_1^{q-1}B_1 \end{bmatrix}_{N_p\times 1} \tag{23}$$

Algorithm 1: Predictive control through incorporating two linear models.

Input: Current state x_k, previous input u_k, prediction horizon N_p, starting step $k+q$ for the second model, reference signal r_k.

Output: input for next step u_{k+1}.

Compute $\bar{RG}$ of $[r_{k+1},r_{k+1},\ldots,r_{k+N_p}]$
$RG\leftarrow\rho\bar{RG}$
Generate $\mathcal{U}$ with $\mathcal{U}(i,j)\in[-RG+u_k,RG+u_k]$ randomly generated
$i\leftarrow 1$
while $i<l+1$ **do**
  Start from x_k, with the input sequence $\mathcal{U}(i,1:N_p)$ using Eq. (6) to compute $X=[x_{k+1},\cdots,x_{k+N_p}]$.
  For each state $x_k\in X$, compute the observations $z_{i,k}=[g_1(x_k),g_2(x_k),\cdots,g_m(x_k)]^T$.
Construct matrices Z^+ and Z in Eq. (12) with $z_{i,k}$ and $\mathcal{U}$
Solve the optimization problem Eq. (13) to obtain $[A_1\ B_1]$
Solve the optimization problem Eq. (21) to obtain $U^*=[u^*_{k+1},u^*_{k+2},\cdots,u^*_{k+N_p}]^T$.
Use U^* to compute x_{k+q} with Eq. (6)
With x_{k+q} and u^*_{k+q}, repeat steps 1-8 to obtain the second linear model $[A_2\ B_2]$
Update the MPC parameters G, F, and H according to Eq. (24)
Solve the optimization problem Eq. (21) to obtain the next input u_{k+1}

$$H=\begin{bmatrix} 0 & 0 & \cdots & 0 \\ CB_1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA_1^{q-2}B_1 & CA_1^{q-3}B_1 & \cdots & 0 \\ CA_2A_1^{q-2}B_1 & CA_2A_1^{q-3}B_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA_2^{N_p-q}A_1^{q-2}B_1 & CA_2^{N_p-q-1}A_1^{q-3}B_1 & \cdots & 0 \end{bmatrix}_{N_p\times N_p} \tag{24}$$

Remark 3. Note that the choice of q is important to the accuracy of Eq. (22). It has to be properly chosen to ensure that Eq. (22) is more accurate than the single model Eq. (8). The purpose of introducing the second linear model is that in case RG is very large, the modeling uncertainty of the model Eq. (8) after q steps may be very large, thus re-linearizing the nonlinear system model can improve the overall linearization accuracy for the entire future N_p steps.

IV. Simulation Results

In the simulation, the RNN model [1] which captures the nonlinear dynamics of a PEA was used.

A. Evaluate the linearization accuracy

When evaluating the linearization accuracy, the input to the nonlinear system was pre-designed and thus no controller

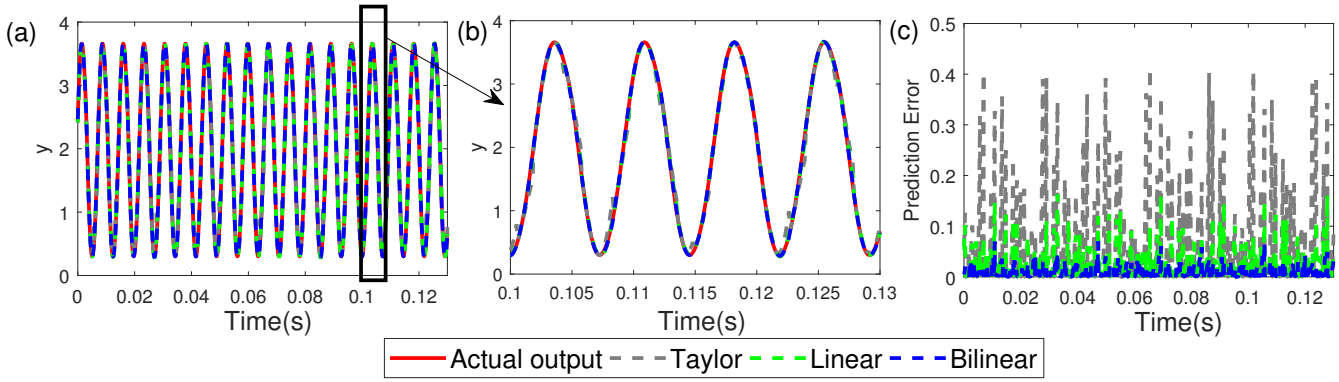

Fig. 2: Comparison of the prediction accuracy of Taylor, Koopman linear and bilinear approaches for 137Hz sinusoidal input. (a) Prediction results, (b) zoomed-in view of (a), (c) prediction errors.

was involved. Specifically, Eq. (6) was linearized every $N_1 = 20$ sampling intervals, and for the following N_1 steps the same input was fed into the nonlinear system and the linearized system to compare the outputs. Input signal with different frequencies (12Hz, 137Hz, and 352.6Hz sinusoidal signal) were tested.

Three approaches—the Taylor approximation (Eq. (7)), the linear model (Eq. (8)), and the bilinear model (Eq. (14)) were compared. As an example, prediction errors in the time domain are shown in Fig. 2 for 137Hz sinusoidal input. The numerical comparison results are presented in Table. I. Suppose the tracking error vector is E, 2-norm, ∞-norm, and variance of E are shown in Table. I. The output for the future 20 steps are predicted. The size of $\mathcal{U}$ is 60×15 and $RG = 0.5\bar{RG}$. For the observations in Koopman operator approach, $g_1 = W_2 x_k + B_3$ and $g_i = \bar{W}_{gi} \tanh(W_{gi} x_k + B_{gi}) + \bar{B}_{gi}, i > 1$, where $\bar{W}_{gi}, W_{gi}, B_{gi}$, and $\bar{B}_{gi}$ were randomly generated.

From Table. I, it can be seen that both the Koopman linear and bilinear model improved the prediction errors by at least 50% for all the three inputs compared to the Taylor approximation. As seen in Fig. 2, the Taylor approach can predict well at the points close to the linearization point but deviates away from the actual outputs at points away from the linearization point.

In Table. I, compared to the linear model, the bilinear model can achieve higher accuracy by at least 30%, which is also verified in the Fig. 2. In addition, it is worthwhile to note that the prediction errors of the linear and bilinear approach are not always lower than that of Taylor approximation for every point as shown in Fig. 2, considering that they may sacrifice the accuracies around the linearization point for better approximation accuracy over N_p steps as explained in **Remark 1.**

B. Performances of predictive control with different linearized models

With the linearized system, predictive controller can be applied. The constraints on the input were ignored, however, they can be trivially added to the controller. We chose $N_p = 20$ and $\rho = 0.5$. Koopman approach using single linear model is denoted by "Koopman1" in Table. II while the one with two linear models is "Koopman2". The order of the linear model for Koopman approach was 15.

TABLE I: Prediction error comparison.

Inputs		Taylor	Linear	Bilinear
12Hz	N2	0.0250	0.0025	1.36e-7
	N∞	0.0238	0.0017	8.88e-8
	Var	0.0079	0.0005	1.14e-8
137Hz	N2	3.6631	1.3871	0.5336
	N∞	0.4123	0.1636	0.0701
	Var	0.0096	0.0013	0.0002
352.6Hz	N2	19.9141	6.6388	4.0187
	N∞	2.0258	0.5598	0.5247
	Var	0.2831	0.0269	0.0113

Overall, Koopman approach can eliminate the tracking error by at least 80% for all the trajectories as shown in Table. II, as well as Fig. 3. This improvement can be attributed to two factors. One is the linearization accuracy over the prediction horizon as validated in the previous subsection. Another reason is the strategy of dynamically changing the exploration range $[-RG, RG]$ in which $\mathcal{U}$ was generated for obtaining the model parameters. This range will determine how local the dynamics is as explained in **Remark 2.** When tracking low-frequency trajectory (i.e., 13Hz), incorporating two linear models was not advantageous over the single linear model. But when the trajectory frequency was increased, "Koopman2" could improve the performance by about 50% compared to "Koopman1". This is due to the soaring prediction error for high-frequency input, thus introducing second linear model can mitigate the surging prediction errors.

Compared to the nonlinear predictive controller in [1], Koopman approach transforms the intractable nonlinear optimization problem to a tractable convex optimization problem while maintaining the tracking accuracy. The computation cost depends on the order of the linearized model. It is possible to use low order linearized model to ensure the

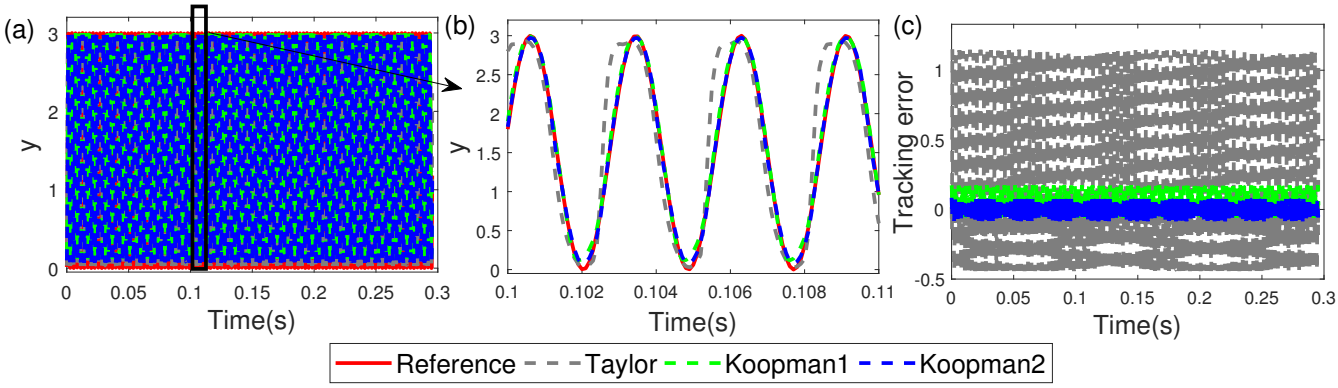

Fig. 3: Comparison of the simulated tracking results of Taylor, Koopman1, and Koopman2 approaches for 356.2Hz trajectory. (a) Tracking results, (b) zoomed-in view of (a), (c) tracking errors.

TABLE II: Tracking performance comparison in simulation.

Refs.	13Hz		187Hz		352.6Hz	
$Error(\%)$	E_{rms}	E_{max}	E_{rms}	E_{max}	E_{rms}	E_{max}
Taylor	0.462	0.582	6.685	10.435	15.208	23.740
Koopman1	0.010	0.021	1.498	2.137	3.206	4.001
Koopman2	0.017	0.026	0.531	1.083	1.054	2.217

control accuracy and computation efficiency, as the case demonstrated in simulations here. Moreover, Koopman approach does not depend on the form of the nonlinear model which is just used for generating data, thus it can be applied on other discrete nonlinear systems (e.g., the systems in [17], [18]). Therefore, Koopman approach is more suitable for precision control provided that the nonlinear system model is known.

V. Conclusions and Future Work

In this work, linearization based on Koopman operators has been proposed for predictive control in precision positioning system. Koopman approach linearizes the nonlinear dynamics in the way that the prediction error over future N_p steps is minimized thus is suitable for predictive control. Furthermore, by introducing two linear models, the control performance can be further improved. The simulation results showed that Koopman approach is much better than the Taylor approximation for system linearization.

Acknowledgment

This work was supported by the National Science Foundation (NSF) (CMMI-1751503) and Iowa State University.

References

[1] S. Xie and J. Ren, "Recurrent-neural-network-based predictive control of piezo actuators for trajectory tracking," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 6, pp. 2885–2896, 2019.

[2] M. Ławryńczuk, *Computationally efficient model predictive control algorithms*. Springer, 2014.

[3] J. Levine, *Analysis and control of nonlinear systems: A flatness-based approach*. Springer Science & Business Media, 2009.

[4] H. K. Khalil, "Nonlinear systems," *Upper Saddle River*, 2002.

[5] A. Rauh, J. Minisini, and H. Aschemann, "Carleman linearization for control and for state and disturbance estimation of nonlinear dynamical processes," *IFAC Proceedings Volumes*, vol. 42, no. 13, pp. 455–460, 2009.

[6] R. M. Jungers and P. Tabuada, "Non-local linearization of nonlinear differential equations via polyflows," in *2019 American Control Conference (ACC)*, pp. 1–6, IEEE, 2019.

[7] I. Mezić, "Spectral properties of dynamical systems, model reduction and decompositions," *Nonlinear Dynamics*, vol. 41, no. 1-3, pp. 309–325, 2005.

[8] H. Arbabi and I. Mezic, "Ergodic theory, dynamic mode decomposition, and computation of spectral properties of the koopman operator," *SIAM Journal on Applied Dynamical Systems*, vol. 16, no. 4, pp. 2096–2126, 2017.

[9] I. Mezić, "Analysis of fluid flows via spectral properties of the koopman operator," *Annual Review of Fluid Mechanics*, vol. 45, pp. 357–378, 2013.

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

[11] H. Arbabi, M. Korda, and I. Mezic, "A data-driven koopman model predictive control framework for nonlinear flows," *arXiv preprint arXiv:1804.05291*, 2018.

[12] I. Abraham, G. De La Torre, and T. D. Murphey, "Model-based control using koopman operators," *arXiv preprint arXiv:1709.01568*, 2017.

[13] M. Korda and I. Mezić, "On convergence of extended dynamic mode decomposition to the koopman operator," *Journal of Nonlinear Science*, vol. 28, no. 2, pp. 687–710, 2018.

[14] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A data–driven approximation of the koopman operator: Extending dynamic mode decomposition," *Journal of Nonlinear Science*, vol. 25, no. 6, pp. 1307–1346, 2015.

[15] J.-S. Chiou, F.-C. Kung, and T.-H. Li, "Robust stabilization of a class of singularly perturbed discrete bilinear systems," *IEEE Transactions on Automatic Control*, vol. 45, no. 6, pp. 1187–1191, 2000.

[16] B. Huang, X. Ma, and U. Vaidya, "Feedback stabilization using koopman operator," in *2018 IEEE Conference on Decision and Control (CDC)*, pp. 6434–6439, IEEE, 2018.

[17] Z. Xu, S. Saha, B. Hu, S. Mishra, and A. A. Julius, "Advisory temporal logic inference and controller design for semiautonomous robots," *IEEE Transactions on Automation Science and Engineering*, vol. 16, no. 1, pp. 459–477, 2018.

[18] L. Cheng, W. Liu, Z.-G. Hou, J. Yu, and M. Tan, "Neural-network-based nonlinear model predictive control for piezoelectric actuators," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 12, pp. 7717–7727, 2015.