

Parameter-Varying Koopman Operator for Nonlinear System Modeling and Control

Changyu Lee, Kiyong Park, and Jinwhan Kim

Abstract—This paper proposes a novel approach for modeling and controlling nonlinear systems with varying parameters. The approach introduces the use of a parameter-varying Koopman operator (PVKO) in a lifted space, which provides an efficient way to understand system behavior and design control algorithms that account for underlying dynamics and changing parameters. The PVKO builds on a conventional Koopman model by incorporating local time-invariant linear systems through interpolation within the lifted space. This paper outlines a procedure for identifying the PVKO and designing a model predictive control using the identified PVKO model. Simulation results demonstrate that the proposed approach improves model accuracy and enables predictions based on future parameter information. The feasibility and stability of the proposed control approach are analyzed, and their effectiveness is demonstrated through simulation.

Index Terms—Parameter-varying system, Koopman operator, Model predictive control

I. INTRODUCTION

Model predictive control (MPC) is a powerful algorithm that has proven to be effective for controlling nonlinear systems in various applications, including robotics and transportation [1]–[3]. MPC offers several advantages, such as the ability to handle state and input constraints and the capacity to tackle multi-input multi-output nonlinear systems. However, nonlinear systems pose challenges in optimizing control due to their non-convex nature, resulting in computational complexity and difficulties in ensuring stability and robustness. Additionally, unreliable models can lead to performance degradation and system failure due to constraint violations [4]. Therefore, obtaining accurate system models and addressing non-convex problems are essential for effective MPC, but these tasks can be challenging in practical applications.

Recently, data-driven Koopman operator (KO)-based system identification has gained popularity in research. The KO provides a linear representation of nonlinear autonomous systems in infinite dimensions [5], which can further be approximated in a finite number of dimensions through data-driven approaches [6]. In this approach, user-defined lifting functions and extended dynamic mode decomposition (EDMD) methods are often utilized [7], [8]. Deep neural networks also offer the capability to simultaneously identify

This research was supported by a grant from National R&D Project "Development of an electric-powered car ferry and a roll-on/roll-off power supply system" funded by Ministry of Oceans and Fisheries, Korea (PMS5530). (Corresponding author: Jinwhan Kim)

The authors are with the Department of Mechanical Engineering, Korea Advanced Institute of Science and Technology, Daejeon 34141, South Korea (e-mail: leeck@kaist.ac.kr; qkrlldyd777@kaist.ac.kr; jinwhan@kaist.ac.kr)

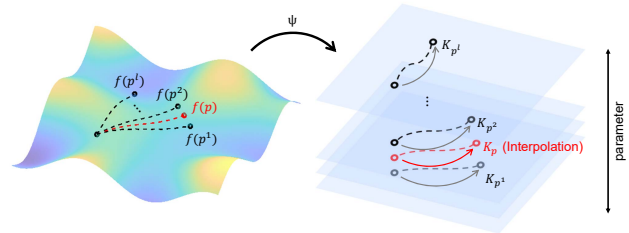


Fig. 1: An illustration of the proposed parameter-varying Koopman operator.

lifting functions as well as the KO [9]. By incorporating a linear MPC algorithm into the linear system in the lifted dimension, the approach can be executed to nonlinear MPC [10]. Furthermore, robust MPC has been developed to address model uncertainty resulting from the identification process based on KOs [11], [12]. These findings suggest the potential of the KO-based approach to address non-convex problems. However, the success of data-driven identification methods heavily depends on the quantity and quality of data, and challenges still remain in this area.

In previous research, linear time-invariant models have often been used to represent nonlinear systems in the lifted space. However, in many real-world systems, the dynamics are dependent on the operating point. For instance, the lateral dynamics of vehicles are influenced by speed, and chemical process models are highly affected by temperature [13], [14]. To address this issue, linear parameter varying (LPV) or quasi-LPV models have been proposed for modeling and designing control systems [15]–[19]. These models account for the influence of exogenous parameters on the system dynamics and provide a more accurate representation of the system behavior. By considering the dependence of the system dynamics on the operating point, LPV models enable the design of controllers that are more robust and effective.

Motivated by recent advances in LPV systems and identification approaches [20], [21], this paper proposes a parameter-varying KO (PVKO) for modeling and controlling nonlinear systems with varying parameters in the lifted space. The proposed approach is based on collecting data for each operating point, identifying a KO for each point, and local interpolation between the KOs is conducted following the approach in [20]. The resulting PVKO provides an accurate and predictable model that accounts for the underlying dynamics and varying parameters. To synthesize the control system, the LPV-MPC approach [19] is used with the PVKO, assuming

the predictability of future parameters. The proposed control system addresses identification uncertainties, and recursive feasibility and stability analysis are provided. Finally, numerical simulations are conducted to verify the effectiveness of the proposed modeling and control approaches.

The rest of this paper is structured as follows. The following section presents the proposed PVKO approach, and Section 3 describes the control system design. The results of the simulations are discussed in Section 4, and the study is concluded in Section 5.

Notation: The notation $I_{n \times m}$ and $\mathbf{0}_{n \times m}$ denote that $n \times m$ identity and zero matrices, respectively. The Minkowski sum and Pontryagin set difference of two sets $\mathbb{X}, \mathbb{Y} \subset \mathbb{R}^n$ are denoted as $\mathbb{X} \oplus \mathbb{Y}$ and $\mathbb{X} \ominus \mathbb{Y}$, respectively. Additionally, $\text{Conv}\{\cdot\}$ represents the convex hull formed by the vertices within $\{\cdot\}$.

II. PARAMETER-VARYING KOOPMAN OPERATOR

Consider the discrete-time nonlinear system defined by:

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

where $\mathbf{x}_k \in \mathbb{X} \subset \mathbb{R}^n$ and $\mathbf{u}_k \in \mathbb{U} \subset \mathbb{R}^m$ denote the state and input vectors, respectively, and the subscript k indicates the time index. Let $\Psi(\mathbf{x}_k, \mathbf{u}_k) \in \mathbb{G} : \mathbb{R}^{n+m} \rightarrow \mathbb{R}^{q+m}$ be an observation function that maps the state and input vectors to the lifted space. The observation function can be defined as follows:

$$\Psi(\mathbf{x}_k, \mathbf{u}_k) = [\psi_1(\mathbf{x}_k), \psi_2(\mathbf{x}_k), \dots, \psi_q(\mathbf{x}_k), \mathbf{u}_k^\top]^\top, \quad (2)$$

where $\psi_i : \mathbb{R}^n \rightarrow \mathbb{R}$ is the i -th component of the observation function. Then, the lifted state vector can be expressed as follows:

$$\mathbf{y}_k = \Psi_{\mathbf{x}}(\mathbf{x}_k) = [\psi_1(\mathbf{x}_k), \psi_2(\mathbf{x}_k), \dots, \psi_q(\mathbf{x}_k)]^\top, \quad (3)$$

where $\mathbf{y}_k \in \mathbb{R}^q$ is the lifted state vector. The KO $\mathcal{K} : \mathbb{G} \rightarrow \mathbb{G}$ can represent the lifted system in the linear form:

$$\mathcal{K}(\Psi(\mathbf{x}_k, \mathbf{u}_k)) = \Psi(\mathbf{x}_{k+1}, \mathbf{u}_{k+1}), \quad (4)$$

which can be approximated in a finite-dimensional space higher than n (typically $q \gg n$) using data. Since this approximation is data-driven, a large amount of data is required, and it is necessary to reduce the dimensions q to a manageable level from a control perspective.

In this paper, we focus on a nonlinear system with exogenous parameters defined as follows:

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k, p_k), \quad (5)$$

where $p_k \in \mathbb{P} \subset \mathbb{R}$ is a bounded parameter that introduces uncertainty into the system. To address this, we propose a new approach for modeling the system as a LPV model in a lifted space. Our approach involves using a PVKO $\mathcal{K}_{p_k} : \mathbb{G} \rightarrow \mathbb{G}$ defined as:

$$\mathcal{K}_{p_k}(\Psi(\mathbf{x}_k, \mathbf{u}_k)) = \Psi(f(\mathbf{x}_k, \mathbf{u}_k, p_k), \mathbf{u}_{k+1}) \quad (6)$$

$$= \Psi(\mathbf{x}_{k+1}, \mathbf{u}_{k+1}), \quad (7)$$

where \mathcal{K}_{p_k} depends on the parameter.

To identify the PVKO, we use an EDMD-based approach, which involves collecting data from the state and input variables of the system at each working point and using the data to identify the KO for each point. We then use an interpolation-based modeling method to find the PVKO. Let

$$\begin{aligned} \mathbf{X}(i) &= [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{M-1}] \in \mathbb{R}^{n \times (M-1)}, \\ \mathbf{X}^+(i) &= [\mathbf{x}_2, \mathbf{x}_3, \dots, \mathbf{x}_M] \in \mathbb{R}^{n \times (M-1)}, \\ \mathbf{U}(i) &= [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_{M-1}] \in \mathbb{R}^{m \times (M-1)}, \end{aligned} \quad (8)$$

denote the collected state and input data at the i -th working point, where M is the number of data points. We then lift the collected data using a lifting function to obtain:

$$\begin{aligned} \mathbf{Y}(i) &= [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_{M-1}] \in \mathbb{R}^{q \times (M-1)}, \\ \mathbf{Y}^+(i) &= [\mathbf{y}_2, \mathbf{y}_3, \dots, \mathbf{y}_M] \in \mathbb{R}^{q \times (M-1)}. \end{aligned} \quad (9)$$

Using the collected data, we can establish the following relations:

$$\begin{aligned} \mathbf{Y}^+(i) &= A(p^i)\mathbf{Y}(i) + B(p^i)\mathbf{U}(i), \\ \mathbf{X}(i) &= C\mathbf{Y}(i), \end{aligned} \quad (10)$$

where p^i represents the parameter at the i -th working point and C is the output matrix. We can then find the state matrix by minimizing the following problems:

$$\begin{aligned} \min_{A(p^i), B(p^i)} & \|\mathbf{Y}^+(i) - (A(p^i)\mathbf{Y}(i) + B(p^i)\mathbf{U}(i))\|_F, \\ \min_C & \|\mathbf{X}(i) - C\mathbf{Y}(i)\|_F, \end{aligned} \quad (11)$$

where $\|\cdot\|_F$ represents the Frobenius norm. We can solve these problems analytically using the pseudo-inverse of the matrix $[\mathbf{Y}(i) \ \mathbf{U}(i)]^\top$ as follows:

$$\begin{aligned} [A(p^i) \ B(p^i)] &= \mathbf{Y}^+(i) [\mathbf{Y}(i) \ \mathbf{U}(i)]^\dagger, \\ C &= \mathbf{X}(i)\mathbf{Y}(i)^\dagger, \end{aligned} \quad (12)$$

where \dagger indicates the Moore–Penrose inverse. To find the pseudo-inverse matrix, we can use the singular value decomposition to decompose $[\mathbf{Y}(i) \ \mathbf{U}(i)]^\top$ as follows:

$$\begin{bmatrix} \mathbf{Y}(i) \\ \mathbf{U}(i) \end{bmatrix} = U\Sigma V^\top. \quad (13)$$

Then, we can approximate $A(p^i)$ and $B(p^i)$ as follows:

$$\begin{aligned} [A(p^i) \ B(p^i)] &\approx \mathbf{Y}^+(i)V\Sigma^{-1}U^\top \\ &= \mathbf{Y}^+(i)V\Sigma^{-1}[U_A \ U_B], \end{aligned} \quad (14)$$

now we can obtain $A(p^i) \approx \mathbf{Y}^+(i)V\Sigma^{-1}U_A$ and $B(p^i) \approx \mathbf{Y}^+(i)V\Sigma^{-1}U_B$.

For a system with $l \in \mathbb{N}$ working points, we can obtain l different $(A(p^i), B(p^i))$ matrices. The PVKO can then be obtained by interpolating these matrices as follows:

$$\begin{aligned} A(p_k) &= \alpha_1(p_k)A(p^1) + \alpha_2(p_k)A(p^2) + \dots + \alpha_l(p_k)A(p^l), \\ B(p_k) &= \alpha_1(p_k)B(p^1) + \alpha_2(p_k)B(p^2) + \dots + \alpha_l(p_k)B(p^l), \end{aligned} \quad (15)$$

where $\alpha_1(p_k), \alpha_2(p_k), \dots, \alpha_l(p_k)$ are weighting coefficients that depend on the parameter p_k . Once we have future parameter information, we can predict the future system matrix

using the identified PVKO $(A(p_k), B(p_k))$. This approach allows us to use LPV-MPC [19].

Remark 1: A subsequent identification procedure is required to determine the functional form of the weighting coefficients. In this paper, we use the simplest interpolation technique, linear interpolation, which is cost-effective and can provide adequate results for many applications.

III. PVKO-BASED MODEL PREDICTIVE CONTROL

Assumption 1: We assumed that the uncertainty of the model approximation, \mathbf{w}_k , is unknown and bounded, i.e., $\mathbf{w}_k = \mathbf{y}_{k+1} - (A(p_k)\mathbf{y}_k + B(p_k)\mathbf{u}_k) \in \mathbb{W} \subset \mathbb{R}^q$.

We propose a method for synthesizing the LPV-MPC algorithm on the lifted space, named PVKO-MPC, based on the identified PVKO. The LPV system with bounded uncertainty \mathbf{w}_k (as stated in Assumption 1) can be represented in the lifted space as follows:

$$\begin{aligned} \mathbf{y}_{k+1} &= A(p_k)\mathbf{y}_k + B(p_k)\mathbf{u}_k + \mathbf{w}_k, \\ \text{s.t. } \mathbf{y}_k &\in \mathbb{Y}, \\ \mathbf{u}_k &\in \mathbb{U}, \\ \mathbf{w}_k &\in \mathbb{W}. \end{aligned} \quad (16)$$

Let the nominal system of (16) be represented as:

$$\bar{\mathbf{y}}_{k+1} = A(p_k)\bar{\mathbf{y}}_k + B(p_k)\bar{\mathbf{u}}_k, \quad (17)$$

where $\bar{\mathbf{u}}_k$ and $\bar{\mathbf{y}}_k$ are the nominal input and state vectors that correspond to the system without uncertainty. The control input of the system (16) is then designed as follows:

$$\mathbf{u}_k = \bar{\mathbf{u}}_k + K(\mathbf{y}_k - \bar{\mathbf{y}}_k), \quad (18)$$

where the second term in (18) is the auxiliary state feedback control that compensates for the error.

Definition 1 (Robust positively invariant set): A set Ω is a robust positively invariant (RPI) set of the system $\mathbf{e}_{k+1} = (A(p_k) + B(p_k)K)\mathbf{e}_k + \mathbf{w}_k$, if $(A(p_k) + B(p_k)K)\mathbf{e}_k + \mathbf{w}_k \in \Omega$ for all $\mathbf{e}_k \in \Omega$, $p_k \in \mathbb{P}$, and $\mathbf{w}_k \in \mathbb{W}$.

Definition 2 (Quadratic stability): The system $\mathbf{y}_{k+1} = A^c(p_k)\mathbf{y}_k$ is quadratically stable if there exists $P > 0$ such that $A^c(p_k)^\top P A^c(p_k) - P \leq -Q - K^\top R K$ for all $p_k \in \mathbb{P}$, where $A^c(p_k) = A(p_k) + B(p_k)K$.

A. Uncertainty compensation and RPI set calculation

Let the error vector be described by $\mathbf{e}_k = \mathbf{y}_k - \bar{\mathbf{y}}_k$. The error system can be represented using (16)-(18) as follows:

$$\begin{aligned} \mathbf{e}_{k+1} &= A(p_k)(\mathbf{y}_k - \bar{\mathbf{y}}_k) + B(p_k)(\mathbf{u}_k - \bar{\mathbf{u}}_k) + \mathbf{w}_k \\ &= (A(p_k) + B(p_k)K)\mathbf{e}_k + \mathbf{w}_k \\ &= A^c(p_k)\mathbf{e}_k + \mathbf{w}_k. \end{aligned} \quad (19)$$

Assumption 2: The system (19) is quadratically stable.

Under the Assumption 2, the state feedback controller that minimizes the worst-case cost can be obtained by solving the following semidefinite programming problem:

$$\begin{aligned} \min_{P, K} \text{tr}(P) \\ \text{s.t. } A^c(p^i)^\top P A^c(p^i) - P &\leq -Q - K^\top R K, \\ &\text{for } i = 1, 2, \dots, l, \end{aligned} \quad (20)$$

where Q, R are weight matrices. We can transform the optimization problem (20) into the following problem using the Schur complement as follows:

$$\begin{bmatrix} P - Q - K^\top R K & A^c(p^i)^\top \\ A^c(p^i) & P^{-1} \end{bmatrix} \geq 0, \text{ for } i = 1, 2, \dots, l. \quad (21)$$

Then, by performing a congruence transformation with $S = P^{-1}$ and introducing $Y = KS$ [22], we can transform the problem into the following form:

$$\begin{aligned} \max_{S, Y} \text{tr}(S) \\ \text{s.t.} \\ \begin{bmatrix} S & SA(p^i)^\top + Y^\top B^\top & SQ^{1/2} & Y^\top R^{1/2} \\ A(p^i)S + BY & S & \mathbf{0}_{q \times q} & \mathbf{0}_{q \times m} \\ Q^{1/2}S & \mathbf{0}_{q \times q} & I_{q \times q} & \mathbf{0}_{q \times m} \\ R^{1/2}Y & \mathbf{0}_{m \times q} & \mathbf{0}_{m \times q} & I_{m \times m} \end{bmatrix} \\ \geq 0, \text{ for } i = 1, 2, \dots, l. \end{aligned} \quad (22)$$

The problem (22) can be solved by convex optimization software, YALMIP [23]. Once the state feedback gain K is obtained, the Assumption 2 is satisfied, and then the RPI set \mathbb{S} of the error system (19) can be calculated as follows:

$$\begin{aligned} \mathbb{S} &= \mathbb{W} \oplus \text{Conv}\{A^c(p^i)\mathbb{W}, \forall i \in \{1, 2, \dots, l\}\} \\ &\quad \oplus \text{Conv}\{A^c(p^i)A^c(p^j)\mathbb{W}, \forall i, j \in \{1, 2, \dots, l\}\} \\ &\quad \oplus \dots \end{aligned} \quad (23)$$

B. Robust MPC strategy

The nominal control input can be computed using the following MPC problem with the RPI set:

$$\min_{\bar{\mathbf{y}}(\cdot), \bar{\mathbf{u}}(\cdot)} \sum_{k=0}^{N-1} (\|\bar{\mathbf{y}}_{k|t}\|_Q^2 + \|\bar{\mathbf{u}}_{k|t}\|_R^2) + \|\bar{\mathbf{y}}_{N|t}\|_P^2, \quad (24)$$

$$\begin{aligned} \text{s.t. } \bar{\mathbf{y}}_{0|t} &= \Psi_{\mathbf{x}}(\mathbf{x}_{0|t}), \\ \bar{\mathbf{y}}_{k+1|t} &= A(p_{k|t})\bar{\mathbf{y}}_{k|t} + B(p_{k|t})\bar{\mathbf{u}}_{k|t}, \\ &\quad k = 0, \dots, N-1, \\ C\bar{\mathbf{y}}_{k|t} &\in \mathbb{X} \ominus C\mathbb{S}, \quad k = 0, \dots, N-1, \\ \bar{\mathbf{u}}_{k|t} &\in \mathbb{U} \ominus C\mathbb{K}\mathbb{S}, \quad k = 0, \dots, N-1, \\ \bar{\mathbf{y}}_{N|t} &\in \mathbb{Y}_f \ominus \mathbb{S}, \end{aligned} \quad (25)$$

where N is the prediction horizon, Q, R , and P penalize the state, input, and terminal state, respectively, the subscript $(\cdot)_{k|t}$ represents the value at time $t+k$ predicted at time t , and \mathbb{Y}_f is the terminal set.

Definition 3 (Maximal positively invariant set): A set $\Omega_\infty \subset \mathbb{Y}$ is a maximal positively invariant set (MPI) set of the system $\mathbf{y}_{k+1} = A^c(p_k)\mathbf{y}_k + \mathbf{w}_k$ if Ω_∞ is invariant and all RPI sets are contained.

In MPC design, the state feedback gain K , obtained from (22) and its corresponding P matrix, can be used to establish recursive feasibility and stability through a terminal set and cost [24]. The terminal set is obtained by implementing the

terminal control input strategy $\bar{\mathbf{u}}_{N|t} = K\bar{\mathbf{y}}_{N|t}$. The set is designed to ensure the satisfaction of the following condition:

$$\mathbf{y}_{N|t} \in \mathbb{Y}_f \Rightarrow \mathbf{y}_{N+1|t} \in \mathbb{Y}_f, \forall t \in \mathbb{N}^+, C\mathbb{Y}_f \subset \mathbb{X}. \quad (26)$$

The MPI set is often chosen as the terminal set, but in practice, the RPI set can be used if the nominal system (17) is stable.

C. Recursive feasibility and stability analysis

Assumption 3: At the initial time, a feasible solution exists for the nominal PVKO-MPC problem.

Assumption 4: The model parameter p_k is known over the prediction horizon.

Assumption 5: The stage cost and terminal cost are positive definite functions, i.e., they are strictly positive and only equal to zero at the origin.

Theorem 1: Assume that Assumptions 3 and 4 hold. Then, for any time t , a feasible solution to the PVKO-MPC problem (24) always exists.

Proof: Let the initial time be t , and let the feasible optimal control sequence and the corresponding state sequence be as follows:

$$\begin{aligned} \bar{U}_t^* &= [\bar{\mathbf{u}}_{0|t}^*, \bar{\mathbf{u}}_{1|t}^*, \dots, \bar{\mathbf{u}}_{N-1|t}^*], \\ \bar{Y}_t^* &= [\bar{\mathbf{y}}_{0|t}^*, \bar{\mathbf{y}}_{1|t}^*, \dots, \bar{\mathbf{y}}_{N|t}^*]. \end{aligned} \quad (27)$$

At the next time $t+1$, we can obtain the predicted state sequence with the control law $\bar{U}_{t+1} = [\bar{\mathbf{u}}_{1|t}^*, \bar{\mathbf{u}}_{2|t}^*, \dots, \bar{\mathbf{u}}_{N-1|t}^*, K\bar{\mathbf{y}}_{N|t}^*]$ as $\bar{Y}_{t+1} = [\bar{\mathbf{y}}_{1|t}^*, \bar{\mathbf{y}}_{2|t}^*, \dots, \bar{\mathbf{y}}_{N|t}^*, A^c(p_{N-1|t+1})\bar{\mathbf{y}}_{N|t}^*]$. Under the Assumption 3, the terminal state $\bar{\mathbf{y}}_{N|t}^*$ at time t satisfies the terminal constraints. Then under the condition of the terminal set (26), $A^c(p_{N-1|t+1})\bar{\mathbf{y}}_{N|t}^*$ also satisfies the terminal constraints. As a result, the MPC problem (24) is recursively feasible due to the above recursion. \square

Theorem 2: Suppose that Assumptions 3 to 5 hold, the system (17) is asymptotically stable under the solution to the MPC problem (24).

Proof: Let J_t be a Lyapunov function defined as follows:

$$J_t = \sum_{k=0}^{N-1} (\|\bar{\mathbf{y}}_{k|t}\|_Q^2 + \|\bar{\mathbf{u}}_{k|t}\|_R^2) + \|\bar{\mathbf{y}}_{N|t}\|_P^2. \quad (28)$$

Let J_t^* be the optimal cost at time t , which can be computed by (27), and also let \hat{J}_{t+1} be the cost at time $t+1$, which can be computed by \bar{U}_{t+1} and \bar{Y}_{t+1} as follows:

$$\begin{aligned} \hat{J}_{t+1} &= \sum_{k=0}^{N-1} (\|\bar{\mathbf{y}}_{k|t}\|_Q^2 + \|\bar{\mathbf{u}}_{k|t}\|_R^2) - \underbrace{(\|\bar{\mathbf{y}}_{0|t}\|_Q^2 + \|\bar{\mathbf{u}}_{0|t}\|_R^2)}_{\geq 0 \text{ (Assumption 5)}} \\ &= \underbrace{J_t^* - \|\bar{\mathbf{y}}_{N|t}^*\|_P^2}_{\leq J_t^* - \|\bar{\mathbf{y}}_{N|t}^*\|_P^2} \\ &\quad + (\|\bar{\mathbf{y}}_{N|t}^*\|_Q^2 + \|K\bar{\mathbf{y}}_{N|t}^*\|_R^2) + \|\bar{\mathbf{y}}_{N|t+1}\|_P^2 \\ &\leq J_t^* - \|\bar{\mathbf{y}}_{N|t}^*\|_P^2 + \|\bar{\mathbf{y}}_{N|t}^*\|_Q^2 + \|\bar{\mathbf{y}}_{N|t}^*\|_{K^T R K}^2 \\ &\quad + \|\bar{\mathbf{y}}_{N|t}^*\|_{A^c P A^c}^2 \\ &\leq J_t^* + \underbrace{\|\bar{\mathbf{y}}_{N|t}^*\|_{A^c P A^c - P + Q + K^T R K}^2}_{\leq 0 \text{ (Assumption 2)}} \\ &\leq J_t^*, \end{aligned} \quad (29)$$

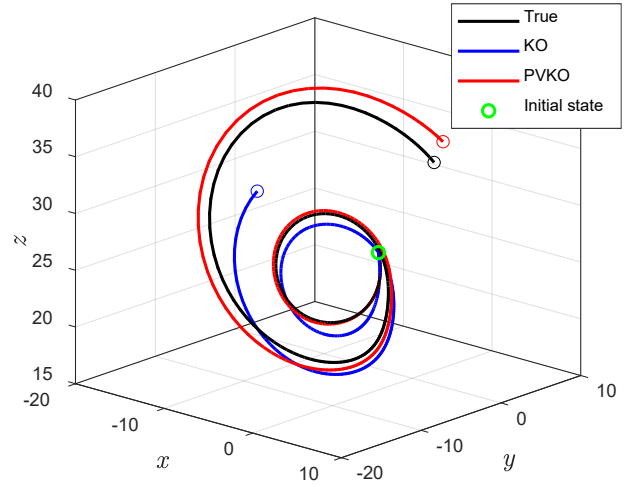


Fig. 2: Comparison of predicted trajectories.

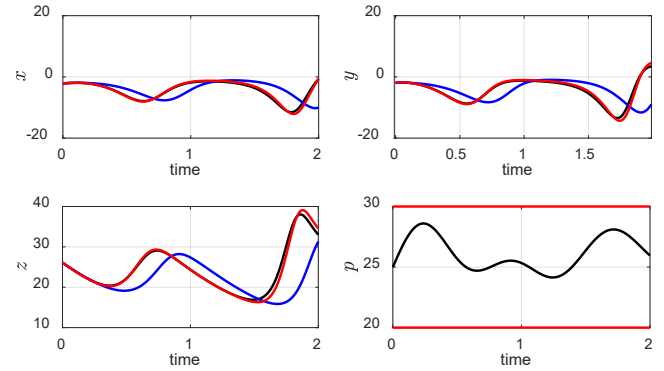


Fig. 3: Time trajectories of the states and parameter.

where $A^c = A^c(p_{N-1|t+1})$. As $J_{t+1}^* \leq \hat{J}_{t+1}$, we can obtain $J_{t+1}^* \leq J_t^*$. Thus, with the proposed controller (24), the nominal system (17) converges to zero as $t \rightarrow \infty$. \square

IV. SIMULATION RESULTS

The performance of the modeling accuracy and control system was verified through two simulations.

A. Prediction Performance

The prediction performance of the proposed PVKO approach is evaluated using the Lorenz model, which is defined by the following set of equations:

$$\begin{aligned} \dot{x} &= 10(y - x), \\ \dot{y} &= px - y - xz, \\ \dot{z} &= xy - z, \end{aligned} \quad (30)$$

where p is a time-varying parameter defined as $p(t) = 25 + \sum_{i=1}^{20} a_i \sin(f_i t)$, where $\sum_{i=1}^{20} a_i = 5$ and $a_i > 0$. The value of f_i is selected from a uniform distribution from 0 to 10. The proposed PVKO approach is applied using 50 thin-plate radial basis functions for the lifting functions, with three working points, $p = 20, 25, 30$. The simulation data with a 50 s duration and a 0.01 s sampling time are used for each working point's KO modeling, while 150 s of

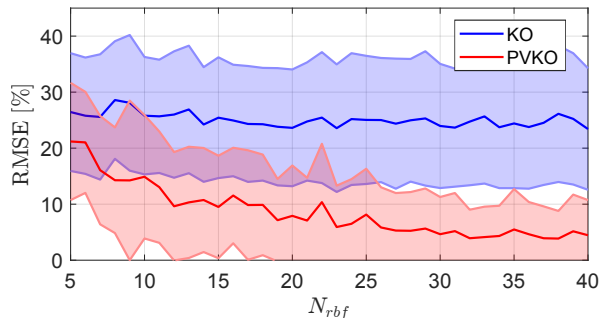


Fig. 4: Comparison of performance according to the order of lifting function (The shaded region represents one standard deviation from the mean).

TABLE I: Parameters of controllers

Symbol	Value	Symbol	Value	Symbol	Value
N	50	T_s	0.01	$[p, \bar{p}]$	$[1, 5]$
Q	$\text{diag}([1, 1])$	R	0.1	$[\underline{u}, \bar{u}]$	$[-3, 3]$
K	$[-0.2036, -0.3152, 0.0117, 5.3363 \cdot 10^{-5}, -0.0062, 0.0489, -0.0147, -4.3624 \cdot 10^{-5}, 0.0035]$				

simulation data were used for conventional (time-invariant) KO modeling. The prediction is performed for 2 s, and the resulting trajectory and parameter over time are shown in Figs. 2 and 3.

To evaluate the quantitative performance and the effects of the order of lifting function, a Monte-Carlo simulation was conducted. For each order, 500 prediction simulations with a 200-step prediction (2 seconds) were conducted, and the prediction accuracy was computed using the root mean square error (RMSE) as follows:

$$\text{RMSE} = 100 \frac{\sqrt{\sum_k \|\hat{\mathbf{x}}_k - \mathbf{x}_k\|_2^2}}{\sqrt{\sum_k \|\mathbf{x}_k\|_2^2}}, \quad (31)$$

where $\hat{\mathbf{x}}_k$ is a predicted state vector. As shown in Fig. 4, the proposed PVKO approach outperforms the time-invariant KO for the parameter-varying Lorenz model simulation.

B. Control Performance

The performance of the PVKO-MPC is evaluated using the Van der Pol oscillator model with a time-varying model, given by:

$$\begin{aligned} \dot{x} &= 2y, \\ \dot{y} &= -0.8x + p(y - 2x^2y) + u, \end{aligned} \quad (32)$$

where the control input u and the time-varying parameter p are subject to a random walk model and are constrained to specific value ranges. The proposed PVKO model is identified by using the polynomial function as a lifting functions, given by $\Psi = [x, y, xy, x^2, y^2, x^2y, xy^2, x^3, y^3]^T$, resulting in a dimension of 9. For the PVKO modeling, five working points with $p = 1, 2, 3, 4, 5$ are used. A 1000 s simulation data with 0.01 s sampling time were used for each working point's KO modeling, while 5000 s simulation data were used for conventional KO modeling. Linear interpolation is

used to construct a complete PVKO model. We compared the performance of the PVKO-MPC algorithm with the KO-MPC [10] and nonlinear MPC (NMPC) algorithms. It's worth noting that only the NMPC algorithm utilizes full knowledge of the model. The PVKO-MPC algorithm is compared with the KO-MPC and nonlinear MPC (NMPC) algorithm with full knowledge of the model as follows:

$$\min_{\mathbf{x}(\cdot), \mathbf{u}(\cdot)} \sum_{k=0}^{N-1} (\|\mathbf{x}_{k|t}\|_{CQC^T}^2 + \|\mathbf{u}_{k|t}\|_R^2) + \|\mathbf{x}_{N|t}\|_{CPC^T}^2 \quad (33)$$

$$\begin{aligned} \text{s.t. } \mathbf{x}_{k+1|t} &= f_d(\mathbf{x}_{k|t}, \mathbf{u}_{k|t}, p_{k|t}), \quad k = 0, \dots, N-1, \\ \mathbf{x}_{k|t} &\in \mathbb{X}, \quad k = 0, \dots, N, \\ \mathbf{u}_{k|t} &\in \mathbb{U}, \quad k = 0, \dots, N-1, \end{aligned} \quad (34)$$

where N is the prediction horizon, the weight matrices Q , R , and P are defined as in (24), and matrix C is identified in (12). The function f_d is obtained by discretizing the nonlinear function (5) using the Euler method with a sampling time of $T_s = 0.01$ s. The controller's parameters are provided in Table I.

To compare the performance of two controllers, simulations were conducted using (32) with an initial state of $[x, y] = [3, 0.5]$ and a time-varying parameter is shown in Fig. 5b. The PVKO-MPC problem (24) was solved using the light-weight sparse quadratic programming solver, qpSWIFT [25], while the interior point optimizer, IPOPT [26], with CasADi software [27] in MATLAB was used for NMPC.

Figure 5 shows the result of the three controllers and optimal trajectory obtained by (33) with $N = \infty$. The cumulative cost is calculated as $J_c(k) = \sum_{i=0}^k (\|\mathbf{x}_i\|_{CQC^T}^2 + \|\mathbf{u}_i\|_R^2)$, and the resulting costs are shown in Fig. 5d. As can be seen, the PVKO-MPC spent less cost than KO-MPC in this simulation and almost similar with NMPC, which uses full knowledge of the model. The average computation time and the cumulative cost are summarized in Table. II.

TABLE II: Average computation time, the cumulative cost, and the cost ratio of three controllers (where J_c^* is the cumulative cost of global optimal trajectory)

	NMPC	KO-MPC	PVKO-MPC
Avg. computation time	0.0056	0.0032	0.0033
Cumulative cost J_c	5136.7	5585.5	5246.8
$100 \cdot (J_c - J_c^*)/J_c^*$	0.71%	9.51%	2.87%

V. CONCLUSION

In this paper, we proposed the data-driven PVKO approach for modeling and controlling parametric uncertain nonlinear systems. Our method involved identifying local Koopman operators at each working point and interpolating them to form a complete PVKO. Furthermore, we designed a PVKO-MPC approach with a robust error-compensation controller, derived through linear matrix inequality, and provided recursive feasibility and stability analysis. The efficacy of the proposed approach was demonstrated through simulations, which showed improved accuracy in modeling and performance in controlling for uncertain nonlinear systems.

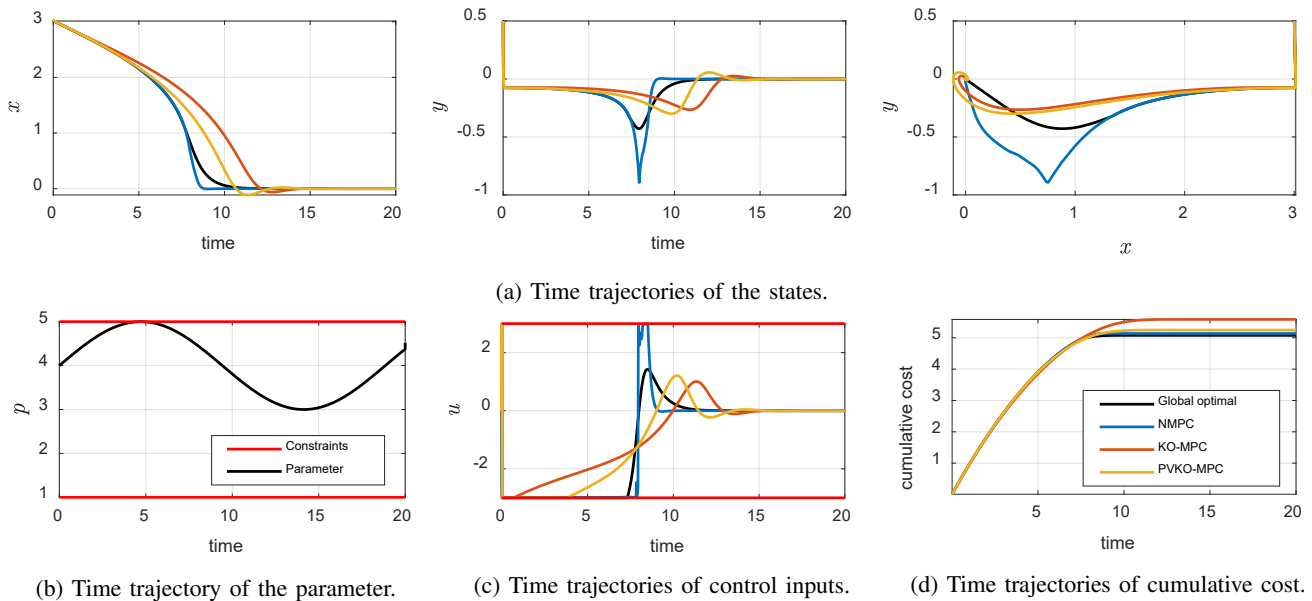


Fig. 5: Simulation results.

REFERENCES

- [1] D. Q. Mayne, "Model predictive control: Recent developments and future promise," *Automatica*, vol. 50, no. 12, pp. 2967–2986, 2014.
- [2] J. Kim, C. Lee, D. Chung, and J. Kim, "Navigable area detection and perception-guided model predictive control for autonomous navigation in narrow waterways," *IEEE Robotics and Automation Letters*, 2023.
- [3] C. Lee, D. Chung, J. Kim, and J. Kim, "Nonlinear model predictive control with obstacle avoidance constraints for autonomous navigation in a canal environment," *arXiv preprint arXiv:2307.09845*, 2023.
- [4] W. Langson, I. Chrysochoos, S. Raković, and D. Q. Mayne, "Robust model predictive control using tubes," *Automatica*, vol. 40, no. 1, pp. 125–133, 2004.
- [5] B. O. Koopman, "Hamiltonian systems and transformation in hilbert space," *Proceedings of the National Academy of Sciences*, vol. 17, no. 5, pp. 315–318, 1931.
- [6] S. L. Brunton, B. W. Brunton, J. L. Proctor, and J. N. Kutz, "Koopman invariant subspaces and finite linear representations of nonlinear dynamical systems for control," *PLoS one*, vol. 11, no. 2, p. e0150171, 2016.
- [7] J. L. Proctor, S. L. Brunton, and J. N. Kutz, "Dynamic mode decomposition with control," *SIAM Journal on Applied Dynamical Systems*, vol. 15, no. 1, pp. 142–161, 2016.
- [8] 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.
- [9] Y. Han, W. Hao, and U. Vaidya, "Deep learning of koopman representation for control," in *2020 59th IEEE Conference on Decision and Control (CDC)*. IEEE, 2020, pp. 1890–1895.
- [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] X. Zhang, W. Pan, R. Scattolini, S. Yu, and X. Xu, "Robust tube-based model predictive control with koopman operators," *Automatica*, vol. 137, p. 110114, 2022.
- [12] G. Mamakoukas, S. Di Cairano, and A. P. Vinod, "Robust model predictive control with data-driven koopman operators," in *2022 American Control Conference (ACC)*. IEEE, 2022, pp. 3885–3892.
- [13] T. I. Fossen, *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons, 2011.
- [14] Z. Xu, J. Zhao, J. Qian, and Y. Zhu, "Nonlinear mpc using an identified lpv model," *Industrial & Engineering Chemistry Research*, vol. 48, no. 6, pp. 3043–3051, 2009.
- [15] J. Mohammadpour and C. W. Scherer, *Control of linear parameter varying systems with applications*. Springer Science & Business Media, 2012.
- [16] S. Mate, H. Kodamana, S. Bhartiya, and P. Nataraj, "A stabilizing sub-optimal model predictive control for quasi-linear parameter varying systems," *IEEE Control Systems Letters*, vol. 4, no. 2, pp. 402–407, 2019.
- [17] M. Lovera, C. Novara, P. L. Dos Santos, and D. Rivera, "Guest editorial special issue on applied lpv modeling and identification," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 1, pp. 1–4, 2010.
- [18] C. Hoffmann and H. Werner, "A survey of linear parameter-varying control applications validated by experiments or high-fidelity simulations," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 2, pp. 416–433, 2014.
- [19] M. M. Morato, J. E. Normey-Rico, and O. Sename, "Model predictive control design for linear parameter varying systems: A survey," *Annual Reviews in Control*, vol. 49, pp. 64–80, 2020.
- [20] Y. Zhu and Z. Xu, "A method of lpv model identification for control," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 5018–5023, 2008.
- [21] J. De Caigny, R. Pintelon, J. F. Camino, and J. Swevers, "Interpolated modeling of lpv systems," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 6, pp. 2232–2246, 2014.
- [22] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*. SIAM, 1994.
- [23] J. Lofberg, "Yalmip: A toolbox for modeling and optimization in matlab," in *2004 IEEE international conference on robotics and automation (IEEE Cat. No. 04CH37508)*. IEEE, 2004, pp. 284–289.
- [24] F. Borrelli, A. Bemporad, and M. Morari, *Predictive control for linear and hybrid systems*. Cambridge University Press, 2017.
- [25] A. G. Pandala, Y. Ding, and H.-W. Park, "qpswift: A real-time sparse quadratic program solver for robotic applications," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3355–3362, 2019.
- [26] L. T. Biegler and V. M. Zavala, "Large-scale nonlinear programming using ipopt: An integrating framework for enterprise-wide dynamic optimization," *Computers & Chemical Engineering*, vol. 33, no. 3, pp. 575–582, 2009.
- [27] J. A. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "Casadi: a software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, pp. 1–36, 2019.