Koopman-LQR Controller for Quadrotor UAVs from Data

Zeyad M. Manaa, Ayman M. Abdallah, Mohammad A. Abido, and Syed S. Azhar Ali

***Abstract*—Quadrotor systems are common and beneficial in many fields, but their intricate behavior often makes it challenging to design effective and optimal control strategies. Some traditional approaches to nonlinear control often rely on local linearizations or complex nonlinear models, which can be inaccurate or computationally expensive. We present an approach based on data to identify the dynamics of a given quadrotor system using Koopman operator theory which offers a linear, but infinite dimensional representation of nonlinear dynamics. This facilitates the use of globally linear models to get an approximation for the nonlinear systems, which can be analyzed and controlled using standard linear optimal control techniques. We leverage the method of extended dynamic mode decomposition (EDMD) to identify the Koopman operator from data. We demonstrate that the identified model can be stabilized and controlled by designing a controller using the linear quadratic regulator (LQR).**

I. Introduction

Linear control theory is particularly pertinent for developing interpretable control frameworks by analyzing the spectral components of the associated dynamical system. For example, system stability may be determined with the use of spectral analysis [1]. Due to the dynamics' inherently nonlinear nature, application to non-linear systems is unsuitable, leading to sub-optimal control applications [2].

Designing efficient control for dynamic systems remains an intricate task when the goal is to achieve optimal performance while abiding by state, actuation, and computing constraints. It is necessary to choose a nonlinear model in order to capture the typical nonlinear behaviors of the majority of dynamical systems. Despite the lack of guaranteed global convergence [3], the identification of a nonlinear model from data typically necessitates solving a nonlinear, non-convex optimization problem [4].

Most of the nonlinear system identification strategies need manual initialization and parameter setting of training parameters, which may have an ambiguous influence on the model created. A neural network can be used to model the nonlinear behavior of a dynamical system [5]; however, the accuracy of the model depends on the number of hidden layers, nodes, the activation function, and the termination condition–all chosen through trial and error until satisfactory results are obtained. Additionally, recent development in optimization techniques have made it possible for methods like non-linear model predicative control in real-time by [6, 7] to be implemented. These advancements still depend on accurate mathematical models and are unable to handle model uncertainty. But, researchers focused on data-driven techniques, and learning-based techniques to identify the underlying model of the system [8–12]. For example [13] employed the sparse identification of nonlinear dynamics algorithm to identify the equations of motion of quadrotors, allowing for the identification of varying parameters in an online framework. Jiahao et al. [14] introduced the use of knowledge-based neural ordinary differential (KNDOE) as a dynamic model employing a method inspired by transfer learning to improve system performance in quadrotor platforms. However, these methods are computationally expensive.

In contrast, linear models, being amenable to identification by simple linear regression methods, do not have the complications faced while using nonlinear techniques. However, because most of dynamical systems exhibit distinctly nonlinear behavior, linear models are ill-equipped to represent this behavior [4]. Data-driven techniques for linear model approximations also have been researched by [15]. However, limitations exist for such data-driven modeling of the dynamical systems. These linear models are locally linearized, and therefore do not capture the important nonlinear physical characteristics of the system.

Recent works have focused to get a *globally linearized dynamics models* via *Koopman theory* [16]. The evolution of the dynamics is governed by the Koopman operator through a set of observable functions A linear, though generally infinite-dimensional, representation of the nonlinear dynamics is produced as a result [17–20]. An approximation for the linear infinite dimensional Koopman operator can be obtained from Dynamic Mode Decomposition (DMD) as in ref. [21]. An extension of the DMD algorithm has been carried out by [22] to deal with controlled systems. Williams et al. [23] further extended the DMD to a version in which they utilized the concept of observable functions to augment the system's original states to obtain an approximation of the Koopman operator to account for complicated nonlinearities. Furthermore, Korda and Mezić [24] introduced a version of the EDMD for controlled dynamical systems which meets linear control methods such as Model Predictive Control. Additionally, noisy and biased measurements can be a crucial

This work was supported by the Interdisciplinary Research Center for Aviation and Space Exploration (IRC-ASE) at King Fahd University of Petroleum and Minerals under research project/grant INAE 2401.

Z. M. Manaa, A. M. Abdallah, and S. S. A. Ali are with the IRC-ASE, and the Department of Aerospace Engineering at King Fahd University for Petroleum and Minerals, Dhahran, 31261, Saudi Arabia.

M. A. Abido is with the Interdisciplinary Research Center for Sustainable Energy Systems (IRC-SES), the SDAIA-KFUPM Joint Research Center for Artificial Intelligence, and the Electrical Engineering Department, King Fahd University of Petroleum & Minerals Dhahran, 31261, Saudi Arabia.

factor on the identified parameters. Recent research efforts have addressed this issue [25–28].

This framework has been widely used in robotics [29–34] with different variations in designing the observable functions and control design, in power grids [35], in fluid dynamics [36], in epidemiology [37], and many other fields. Nevertheless, due to the complexity and the topological nature of quadrotors,extending such applications to quadrotors is challenging.

Folkestad et al. [38] then introduced and used Koopman Eigen function Dynamic Mode Decomposition for such purpose, specially to learn the persistent problem of ground effect to improve the quadrotors landing performance. Additionally, recent work on quadrotors has been tested by [39, 40]. However these methods are complicated, slow, and in many cases, they are applied to low dimensional quadrotors (e.g., planar quadrotor).

Therefore, we aim to introduce a simple LQR-based controller for 6 degree-of freedom quadrotors. We will leverage simulation data to construct a set of observable functions from our background knowledge about the system and combine it with previously proposed set in literature. Afterwards, the set of observable functions is used to learn linear model (globally) of the underlying nonlinear quadrotor dynamics using EDMD to approximate the Koopman operator for LQR design.

II. Methodology

A. Koopman Operator Theory and Dynamic Mode Decomposition

Definition 1 (Koopman Operator): Consider the discrete time dynamical system:

$$x_k^+ = f(x_k, u_k), \tag{1}$$

where $x_k \in \mathbb{R}^n$ is the state vector, $u_k \in \mathbb{R}^l$ is the control input, f is a transition map, and x^+ is the successor state. The Koopman operator $\mathcal{K}_t$ is an infinite-dimensional operator:

$$\mathcal{K}_t \xi = \xi \circ f(x_k, u_k),$$

acting on $\xi \in \mathcal{H} : \mathbb{R}^n \times \mathbb{R}^l \mapsto \mathbb{R}$, where $\circ$ denotes function composition. □

The Koopman operator provides A linear representation of a nonlinear system in infinite-dimensional space is provided by the Koopman operator by its action on the Hilbert space $\mathcal{H}$ of measurement functions ξ. Practically, a finite version of the Koopman operator is sought. However, this transformation trades nonlinear dynamics for linear one, and finite-dimensionality for infinite-dimensionality. An advantage of linear systems in control theory is the convex search space for controllers like LQR and MPC (see Fig. 1).

In order to get around the problems caused by infinite dimensionality, we use data to approximate the Koopman operator. Think of a system whose inputs and states are recorded as

$$\mathcal{D} := \{x_k, u_k : k \in [0, (T-1)T_s] \cap \mathbb{Z}_0\},$$

Assumption 1: The set $\mathcal{D}$ exists. □

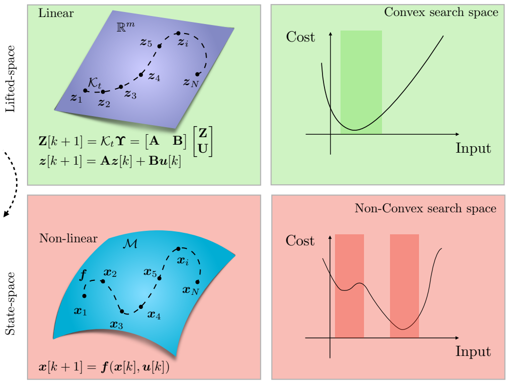

Fig. 1: Illustration of the Koopman Operator: The Koopman Operator ($\mathcal{K}_t$) maps state-space into a lifted linear space governed by ξ. Green panels represent linear space (convex search space for control design), while red panels represent nonlinear state-space (non-convex search space).

Let assumption 1 hold. Define:

$$\Gamma := \begin{bmatrix} u(0) \; u(T_s) \; \dots \; u((T-1)T_s) \end{bmatrix} \in \mathbb{R}^{l \times T}, \tag{2a}$$
$$X := \begin{bmatrix} x(0) \; x(T_s) \; \dots \; x((T-1)T_s) \end{bmatrix} \in \mathbb{R}^{n \times T}, \tag{2b}$$
$$X^+ := \begin{bmatrix} x(1) \; x(T_s) \; \dots \; x((T)T_s) \end{bmatrix} \in \mathbb{R}^{n \times T}. \tag{2c}$$

Assumption 2: Considering $T \geq n + l$, the matrix $\begin{bmatrix} X \\ \Gamma \end{bmatrix}$ has full row rank. □

Definition 2 (Regression using DMD): Consider a dynamical system $x_k^+ = f(x_k, u_k) \approx Ax_k + Bu_k$ and dataset $\mathcal{D}$. The system can be written as:

$$X^+ = AX + B\Gamma \tag{3}$$
$$= \begin{bmatrix} A & B \end{bmatrix} \begin{bmatrix} X \\ \Gamma \end{bmatrix} = \hat{\mathcal{K}}_t \Omega. \tag{4}$$

Assuming 2 holds, solve:

$$\hat{\mathcal{K}}_t = \underset{\hat{\mathcal{K}}_t}{\arg\min} \|X^+ - \hat{\mathcal{K}}_t \Omega\|_{\mathrm{F}} = X^+ \Omega^\dagger, \tag{5}$$

where $\dagger$ denotes the Moore-Penrose pseudo-inverse, and $\|\cdot\|_{\mathrm{F}}$ is the Frobenius norm. □

Remark 1: The Moore-Penrose pseudo-inverse is calculated using SVD. □

The original DMD algorithm was developed for non-controlled systems [21]. In [22], the authors extended it to account for controlled settings. For more on DMD, see refs. [21, 41].

To have an approximation for the Koopman operator, DMD can be extended using EDMD by lifting states into a higher dimensional space with observable functions. The main difference between nominal DMD and EDMD is that in nominal DMD, the observable functions are identity maps. EDMD uses selected observables to approximate the Koopman operator. These observable functions can be found by trial and error or system knowledge. For more on this, see [12, 42–44]. Instead of just system states, consider additional observables:

$\Xi(x) = \begin{bmatrix}\xi_1(x) & \xi_2(x) & \dots & \xi_p(x)\end{bmatrix}^\top$, such that the system (4) becomes

$$\Xi(X^+) = \begin{bmatrix}A & B\end{bmatrix}\begin{bmatrix}\Xi(X)\\ \Gamma\end{bmatrix} = \hat{\mathcal{K}}_t\bar{\Omega}, \tag{6}$$

solved similarly to (5) by

$$\hat{\mathcal{K}}_t = \underset{\hat{\mathcal{K}}_t}{\arg\min}\, \|\Xi(X^+) - \hat{\mathcal{K}}_t\bar{\Omega}\|_{\mathrm{F}}. \tag{7}$$

Remark 2: If the lifting functions are identity, EDMD reduces to nominal DMD. □

The linear lifted approximation of nonlinear dynamics in (1) is

$$\begin{aligned} z_k^+ &= Az_k + Bu_k \\ x_k &= Cz_k. \end{aligned}$$

The implemented technique is summarized in Algorithm 1.

Algorithm 1 Koopman Identification using Total-Least-Squares with L_1 Regularization

```
Require: Dataset $\mathcal{D}$
    Lift data via (11)
    Get the approximated Koopman operator $\hat{\mathcal{K}}_t$ via (7)
    Extract the identified linear matrices $A$, $B$ from $\hat{\mathcal{K}}_t$
    return $\hat{\mathcal{K}}_t$, $A$, and $B$
```

B. Quadcopter dynamics

Quadrotor is assumed to be a rigid body with a 6 degrees of freedom with a mass m and diagonal inertia tensor of $J = \mathrm{diag}(J_{xx}, J_{yy}, J_{zz})$. Our model is built upon quaternion formulation for attitude parameterization and is similar to the standard models found in [45, 46]. For better interpretation of the results quaternions are converted to Euler angles. The state space then is 13-dimensional or 12-dimensional respectively space, based on either quaternion, or Euler angles formulation (respectively). Suppose quaternions are used for attitude formulation, so $x \in \mathbb{R}^{13}$ and is governed by the dynamics

$$\dot{x} = \frac{d}{dt}\begin{bmatrix} p_{WB} \\ \dot{p}_{WB} \\ q_{WB} \\ \omega_B \end{bmatrix} = f(x,u) = \begin{bmatrix} p_W \\ \frac{1}{m}\, q_{WB} \cdot T_B + g_W \\ \frac{1}{2} q_{WB} \otimes \omega_B \\ J^{-1}\left(\tau_B - \omega_B \times J\omega_B\right) \end{bmatrix}, \tag{8}$$

where T_B is the summation of quadrotor's thrust, $g_W = [0, 0, -9.81\,\mathrm{m/s^2}]^\top$ is Earth's gravity, τ_B is the torque, p_{WB}, position vector from world to body frame, and q_{WB} quaternion from world to body frame. Let $\otimes$ denotes the quaternion-vector product by , where, for example, $p \otimes q = qpq^*$, and q^* is the conjugate of the quaternion.

$$T_B = \begin{bmatrix} 0 \\ 0 \\ \sum T_i \end{bmatrix} \quad \text{and} \quad \tau_B = \begin{bmatrix} l(-T_0 - T_1 + T_2 + T_3) \\ l(-T_0 + T_1 + T_2 - T_3) \\ c_\tau(-T_0 + T_1 - T_2 + T_3) \end{bmatrix}, \tag{9}$$

TABLE I: Quadrotor's Parameters

Parameter	Value
Mass m (kg)	0.18
Gravity $g \cdot \hat{k}$ (m/s^2)	9.81
Inertia J (kg·m^2)	$\begin{bmatrix} 0.00025 & 0 & 0 \\ 0 & 0.000232 & 0 \\ 0 & 0 & 0.0003738 \end{bmatrix}$
Moment arm l (m)	0.086

where l is the distance from the center of mass to the motor axis of action and c_τ is a constant for the rotor's drag torque.

The famous Runge-Kutta fourth-order scheme will be considered to integrate the dynamics $\dot{x}$ in a discrete manner with a time step of Δt

$$x_{k+1} = x_k + \int_{k\Delta t}^{(k+1)\Delta t} f(x, u, \Delta t)d\tau \tag{10}$$

C. Data Collection and Koopman Training

A PD controller is designed to track the generated trajectories of the quadrotor. Five helical trajectories are simulated with quadrotor's parameters in table I, providing a diverse set of scenarios for analysis. A time step Δt of 0.01 seconds is set in the simulation to ensure detailed temporal resolution for 30 seconds for each trajectory. Random parameters are utilized to also introduce variability in the trajectories. The helix radius varies randomly between 1 and 5 meters, while the height experiences random fluctuations within the range of 1 to 6 units.

The EDMD model is trained with least squares optimizer dealing with the data using a set of observable functions from our background of the system augmented with another observable functions retrieved from literature (see ref. [47]) defined as

$$\begin{aligned} \Xi(x) = \big[1,\ x,\ p_{WB},\ \dot{p}_{WB},\ \sin(p_{WB}),\ \dots \\ \cos(p_{WB}),\ \mathrm{vec}(R \times \omega_{WB})\big], \end{aligned} \tag{11}$$

where $\mathrm{vec}(\cdot) : \mathbb{R}^{n \times n} \mapsto \mathbb{R}^{n^2 \times 1}$ is an operator that maps a matrix into a vector by concatenating its columns into one vector, and R denotes the rotation matrix representing the quadrotor attitude.

D. Linear Quadratic Regulator Meets Koopman Linearization

Assumption 3: The identified pairs (A, B) and (A, C) are controllable and observable respectively. □

Remark 3: Assumption 3 can be numerically verified for the identified system's matrices. □

Now, consider a quadratic cost function:

$$\mathcal{J} = \underset{u_0,\dots,u_{N-1}}{\text{minimize}} \sum_{0}^{N-1} x^\top(\tau)Qx(\tau) + u^\top(\tau)Ru(\tau)d\tau, \tag{12}$$

where $Q = Q^\top \succeq 0$ is a weight matrix for the cost of deviation of the state x from the reference point, and $R = R^\top \succ 0$ weighs the cost of control action. If a linearized version of (1) is considered, it is possible to have a matrix L, and a control law in the form of $u = -Lx$ such that the cost $\mathcal{J}$ is minimum. However, this controller is linear, and in fact, it will be optimal

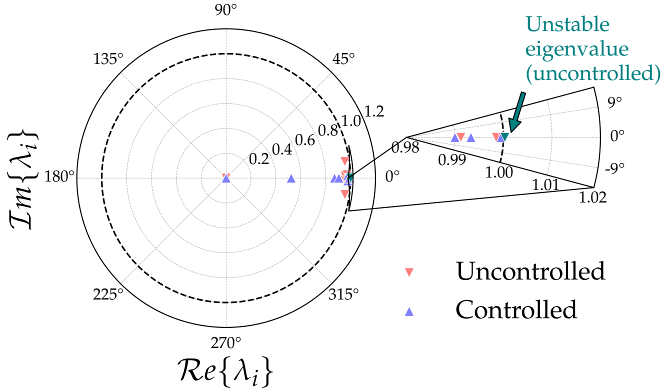

Fig. 2: Eigenvalues of the controlled (light blue) and uncontrolled (light red) Koopman linearized systems. Unlike the controlled system, the uncontrolled system has unstable eigenvalues, which are colored in teal.

only around the neighbourhood of the fixed point at which the linearization took place.

Here, instead of having a local linear model for our system, we will instead use the global linear model derived from Koopman approximation. The cost function still holds but with minor modifications as follows

$$\bar{Q} = \begin{bmatrix} Q_{n\times n} & 0_{p-n\times p-n} \\ 0_{p-n\times p-n} & 0_{p-n\times p-n} \end{bmatrix}_{p\times p}, \quad \bar{R} = R$$

So, the cost function given in (12) will remain the same, while replacing x by z, and Q by $\bar{Q}$.

Remark 4: In this formulation, the matrix R stays as it is as control inputs are remain un-lifted. □

In that sense, an optimal LQR controller can be designed for the Koopman linearization of the system system.

Let $Q = \mathbb{I}_{12\times 12} \times 10^3$, and $R = \mathbb{I}_{4\times 4}$ be chosen for controller design.

The system stability is checked in the lifted space by checking the spectrum of the matrices A and $(A - BK)$ and got the results in Fig. 2. Even in the lifted space we got a confirmation of the system instability with some of the eigenvalues outside the unit disk. The LQR design done on the lifted space rendered the matrix $(A - BK)$ stable with all the eigenvalues contained inside the unit disk.

Furthermore, excellent performance is noted for Koopman control as compared to the nominal PID in the rotational domain of the quadrotor (i.e. the Euler angles and their rates). This can be seen clearly from Table III for the case of Euler angles, as they exhibit the lowest %RMSE. Although the average %RMSE in Table III between the nominal PID and Koopman in terms of angular velocities is quite high, the Koopman controller shows a converging behavior to the reference value.

III. Results and Discussions

In addition to the designed LQR controller, we designed a typical PID controller compiled from several sources such as [48–50] to serve as a typical benchmark with the new method. The controller is implemented in two loops; the inner loop is responsible for attitude control, and the outer loop handles position control. We designed the controller with the parameters presented in table II. The performance of the learned linear system was assessed using the Normalized Root Mean Square Error (NRMSE) as the evaluation metric.

TABLE II: PID controllers parameters.

Parameter	Position			Attitude		
	x	y	z	ϕ	θ	ψ
K_p	30	30	80	200	200	200
K_d	10	10	35	0.1	0.1	0.1
K_i	–	–	–	–	–	–

$$\text{NRMSE} = 100 \times \sqrt{\frac{||x_{\text{pred}} - x_{\text{true}}||_2^2}{||x_{\text{true}}||_2^2}} \tag{13}$$

The average NRMSE of a simulated case that takes place over 1.5 second, are shown in Table III.

The results are within an acceptable range for use in control purposes. The fact that these results are derived from a globally linearized model for a highly nonlinear model such as the quadrotor is remarkable. The predictive capability of the discovered linear mode is also notable. The prediction capability is depicted in Fig. 3 for pre-calculated pd control inputs. The linear model operated in a good way which validates the discovered model.

The problem consists of 39 observable functions. Learning the linearized Koopman operator took 0.1393 seconds on an M2 MacBook Air. The total inference time of the learned system is 0.0035 seconds accumulated over 200 timesteps, with an average of 1.7369×10^{-5} seconds per iteration. These numbers show that the algorithm is not only light and quick in learning but also highly effective during inference, indicating its efficacy for practical settings where both learning and taking the control actions need to be performed quickly.

IV. Conclusion

To sum up, this research offers a data-driven method for creating efficient and ideal control schemes for quadrotor systems. Globally linear models in higher-dimensional space can identify and approximate the dynamics of a quadrotor by utilizing Koopman operator theory and EDMD. By using LQR, the found model enables the design of a stabilizing controller and the prediction of quadrotor dynamics. This strategy provides a more straightforward and understandable framework for control design while overcoming the drawbacks

TABLE III: The average NRMSE of the states between the Koopman and PID based-controllers

states	%NRMSE
p_{WB} – Position	3.2529 ± 2.3216
$\dot{p}_{WB}$ – Velocity	4.8129 ± 3.8608
$\mathcal{E}_{WB}$ – Euler angles	2.4398 ± 2.4263
ω_B – Angular velocity	7.8525 ± 6.6367
Mean	4.5895 ± 3.8114

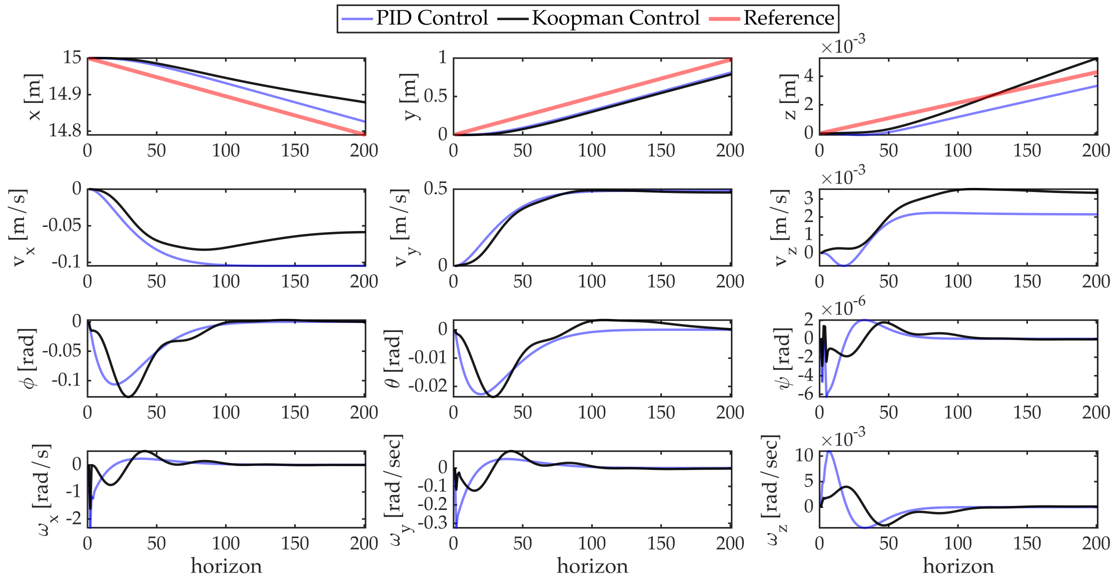

Fig. 3: Illustration of the performance comparison between Nominal PID Control (blue), Koopman Control (black), and the Reference trajectory (red) for a quadrotor. The plots depict the state trajectories over a 200-step prediction horizon, covering the states of the quadrotor including position (x, y, z), velocity ($\dot{x}$, $\dot{y}$, $\dot{z}$), Euler angles (ϕ, θ, ψ), and angular velocities (ω_x, ω_y, ω_z).

of conventional nonlinear control techniques. The suggested approach creates opportunities for open research in modeling quadrotors using global linear models.

Although the algorithm shows promising performance, some limitations we have noticed should be highlighted. First, the system does not scale to different type of trajectories other than the learnt trajectories for our settings limiting its ability to account for general purpose model. When the data is augmented with different types of trajectories, the learning process became harder. Additionally, we have found that the identified model is good on small control horizon (i.e. ~ 150 step, equivalent to ~ 1.5 sec).

Nevertheless, we have laid out one of the initial steps to deal with such problems for complicated system with hard topology like quadrotors. For example, the problem of scalability could be approached by building large datasets with various type of trajectories and conducting ensemble learning on the system. Additionally, the problem of the limited prediction horizon can be tackled by using a recursive estimate of the Koopman operator every N period. Those potentials open the door to myriad of research to validate such concepts.

References

[1] W. Michiels and S.-I. Niculescu, *Stability, control, and computation for time-delay systems: an eigenvalue-based approach*. SIAM, 2014.

[2] N. Komeno, B. Michael, K. Küchler, E. Anarossi, and T. Matsubara, "Deep koopman with control: Spectral analysis of soft robot dynamics," in *2022 61st Annual Conference of the Society of Instrument and Control Engineers (SICE)*, pp. 333–340, IEEE, 2022.

[3] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[4] D. Bruder, C. D. Remy, and R. Vasudevan, "Nonlinear system identification of soft robot dynamics using koopman operator theory," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 6244–6250, IEEE, 2019.

[5] M. T. Gillespie, C. M. Best, E. C. Townsend, D. Wingate, and M. D. Killpack, "Learning nonlinear dynamic models of soft robots for model predictive control with neural networks," in *2018 IEEE International Conference on Soft Robotics (RoboSoft)*, pp. 39–45, IEEE, 2018.

[6] S. Gros, M. Zanon, R. Quirynen, A. Bemporad, and M. Diehl, "From linear to nonlinear mpc: bridging the gap via the real-time iteration," *International Journal of Control*, vol. 93, no. 1, pp. 62–80, 2020.

[7] D. Kouzoupis, G. Frison, A. Zanelli, and M. Diehl, "Recent advances in quadratic programming algorithms for nonlinear model predictive control," *Vietnam Journal of Mathematics*, vol. 46, no. 4, pp. 863–882, 2018.

[8] Z.-S. Hou and Z. Wang, "From model-based control to data-driven control: Survey, classification and perspective," *Information Sciences*, vol. 235, pp. 3–35, 2013.

[9] L. Hewing, J. Kabzan, and M. N. Zeilinger, "Cautious model predictive control using gaussian process regression," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 6, pp. 2736–2743, 2019.

[10] Y. Han, W. Hao, and U. Vaidya, "Deep learning of koopman representation for control," in *2020 59th IEEE Conference on Decision and Control (CDC)*, pp. 1890–1895, IEEE, 2020.

[11] R. Wang, Y. Han, and U. Vaidya, "Deep koopman data-driven control framework for autonomous racing," in *Proc. Int. Conf. Robot. Autom.(ICRA) Workshop Opportunities Challenges Auton. Racing*, pp. 1–6, 2021.

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

[13] Z. M. Manaa, M. R. Elbalshy, and A. M. Abdallah, "Data-driven discovery of the quadrotor equations of motion via sparse

identification of nonlinear dynamics," in *AIAA SCITECH 2024 Forum*, p. 1308, 2024.

[14] T. Z. Jiahao, K. Y. Chee, and M. A. Hsieh, "Online dynamics learning for predictive control with an application to aerial robots," in *Conference on Robot Learning*, pp. 2251–2261, PMLR, 2023.

[15] J.-Q. Huang and F. L. Lewis, "Neural-network predictive control for nonlinear dynamic systems with time-delay," *IEEE Transactions on Neural Networks*, vol. 14, no. 2, pp. 377–389, 2003.

[16] 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.

[17] M. Budišić, R. Mohr, and I. Mezić, "Applied koopmanism," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 22, no. 4, 2012.

[18] S. L. Brunton and J. N. Kutz, *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2022.

[19] Y. Lan and I. Mezić, "Linearization in the large of nonlinear systems and koopman operator spectrum," *Physica D: Nonlinear Phenomena*, vol. 242, no. 1, pp. 42–53, 2013.

[20] 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.

[21] P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," *Journal of fluid mechanics*, vol. 656, pp. 5–28, 2010.

[22] 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.

[23] 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, pp. 1307–1346, 2015.

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

[25] M. S. Hemati, C. W. Rowley, E. A. Deem, and L. N. Cattafesta, "De-biasing the dynamic mode decomposition for applied koopman spectral analysis of noisy datasets," *Theoretical and Computational Fluid Dynamics*, vol. 31, pp. 349–368, 2017.

[26] S. T. Dawson, M. S. Hemati, M. O. Williams, and C. W. Rowley, "Characterizing and correcting for the effect of sensor noise in the dynamic mode decomposition," *Experiments in Fluids*, vol. 57, pp. 1–19, 2016.

[27] T. Askham, P. Zheng, A. Aravkin, and J. N. Kutz, "Robust and scalable methods for the dynamic mode decomposition," *arXiv preprint arXiv:1712.01883*, 2017.

[28] A. H. Abolmasoumi, M. Netto, and L. Mili, "Robust dynamic mode decomposition," *IEEE Access*, vol. 10, pp. 65473–65484, 2022.

[29] D. Bruder, X. Fu, R. B. Gillespie, C. D. Remy, and R. Vasudevan, "Koopman-based control of a soft continuum manipulator under variable loading conditions," *IEEE robotics and automation letters*, vol. 6, no. 4, pp. 6852–6859, 2021.

[30] G. Mamakoukas, M. L. Castano, X. Tan, and T. D. Murphey, "Derivative-based koopman operators for real-time control of robotic systems," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 2173–2192, 2021.

[31] D. A. Haggerty, M. J. Banks, P. C. Curtis, I. Mezić, and E. W. Hawkes, "Modeling, reduction, and control of a helically actuated inertial soft robotic arm via the koopman operator," *arXiv preprint arXiv:2011.07939*, 2020.

[32] M. Han, J. Euler-Rolle, and R. K. Katzschmann, "Desko: Stability-assured robust control with a deep stochastic koopman operator," in *International Conference on Learning Representations*, 2021.

[33] J. Chen, Y. Dang, and J. Han, "Offset-free model predictive control of a soft manipulator using the koopman operator," *Mechatronics*, vol. 86, p. 102871, 2022.

[34] X. Zhu, C. Ding, L. Jia, and Y. Feng, "Koopman operator based model predictive control for trajectory tracking of an omnidirectional mobile manipulator," *Measurement and Control*, vol. 55, no. 9-10, pp. 1067–1077, 2022.

[35] R. R. Hossain, R. Adesunkanmi, and R. Kumar, "Data-driven linear koopman embedding for networked systems: Model-predictive grid control," *IEEE Systems Journal*, 2023.

[36] T. Markmann, M. Straat, and B. Hammer, "Koopman-based surrogate modelling of turbulent rayleigh-b\'enard convection," *arXiv preprint arXiv:2405.06425*, 2024.

[37] I. Mezić, Z. Drmač, N. Črnjarić, S. Maćešić, M. Fonoberova, R. Mohr, A. M. Avila, I. Manojlović, and A. Andrejčuk, "A koopman operator-based prediction algorithm and its application to covid-19 pandemic and influenza cases," *Scientific reports*, vol. 14, no. 1, p. 5788, 2024.

[38] C. Folkestad, Y. Chen, A. D. Ames, and J. W. Burdick, "Data-driven safety-critical control: Synthesizing control barrier functions with koopman operators," *IEEE Control Systems Letters*, vol. 5, no. 6, pp. 2012–2017, 2020.

[39] C. Folkestad, S. X. Wei, and J. W. Burdick, "Quadrotor trajectory tracking with learned dynamics: Joint koopman-based learning of system models and function dictionaries," *arXiv preprint arXiv:2110.10341*, 2021.

[40] C. Folkestad and J. W. Burdick, "Koopman nmpc: Koopman-based learning and nonlinear model predictive control of control-affine systems," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7350–7356, IEEE, 2021.

[41] J. H. Tu, *Dynamic mode decomposition: Theory and applications*. PhD thesis, Princeton University, 2013.

[42] 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: An Interdisciplinary Journal of Nonlinear Science*, vol. 27, no. 10, 2017.

[43] S. E. Otto and C. W. Rowley, "Linearly recurrent autoencoder networks for learning dynamics," *SIAM Journal on Applied Dynamical Systems*, vol. 18, no. 1, pp. 558–593, 2019.

[44] E. Yeung, S. Kundu, and N. Hodas, "Learning deep neural network representations for koopman operators of nonlinear dynamical systems," in *2019 American Control Conference (ACC)*, pp. 4832–4839, IEEE, 2019.

[45] G. Torrente, E. Kaufmann, P. Föhn, and D. Scaramuzza, "Data-driven mpc for quadrotors," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3769–3776, 2021.

[46] J. Carino, H. Abaunza, and P. Castillo, "Quadrotor quaternion control," in *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 825–831, IEEE, 2015.

[47] T. Chen, J. Shan, and H. Wen, "Koopman-operator-based attitude dynamics and control on so (3)," in *Distributed Attitude Consensus of Multiple Flexible Spacecraft*, pp. 177–210, Springer, 2022.

[48] N. A. Johnson, "Control of a folding quadrotor with a slung load using input shaping," *Georgia Institute of Technology*, 2017.

[49] T. Luukkonen, "Modelling and control of quadcopter independent research project in applied mathematics," *Espoo*, vol. 22, p. 22, 2011.

[50] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE robotics & automation magazine*, vol. 19, no. 3, pp. 20–32, 2012.