2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 1-5, 2023. Detroit, USA

Data-Driven Modeling and Experimental Validation of Autonomous Vehicles using Koopman Operator

Ajinkya Joglekar[1], Sarang Sutavani[2], Chinmay Samak[1], Tanmay Samak[1],
Krishna Chaitanya Kosaraju[1], Jonathon Smereka*, David Gorsich*, Umesh Vaidya[2], and Venkat Krovi[1]

2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 978-1-6654-9190-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/IROS55552.2023.10341797

***Abstract*— This paper presents a data-driven framework to discover underlying dynamics on a scaled F1TENTH vehicle using the Koopman operator linear predictor. Traditionally, a range of white, gray, or black-box models are used to develop controllers for vehicle path tracking. However, these models are constrained to either linearized operational domains, unable to handle significant variability or lose explainability through end-2-end operational settings. The Koopman Extended Dynamic Mode Decomposition (EDMD) linear predictor seeks to utilize data-driven model learning whilst providing benefits like explainability, model analysis and the ability to utilize linear model-based control techniques. Consider a trajectory-tracking problem for our scaled vehicle platform. We collect pose measurements of our F1TENTH car undergoing standard vehicle dynamics benchmark maneuvers with an OptiTrack indoor localization system. Utilizing these uniformly spaced temporal snapshots of the states and control inputs, a data-driven Koopman EDMD model is identified. This model serves as a linear predictor for state propagation, upon which an MPC feedback law is designed to enable trajectory tracking. The prediction and control capabilities of our framework are highlighted through real-time deployment on our scaled vehicle.**

I. Introduction

Path tracking is a crucial modality of any autonomous vehicle and its performance can be severely affected by choices and assumptions made in modeling and system identification. Kinematic and geometric path-tracking controllers like Pure-Pursuit [2] are popular choices for low-speed applications owing to their low computational costs and ease of implementation [3]. However, such geometric/kinematic controllers cannot provide adequate path-tracking performance as operational regimes have evolved to include high-speed, non-planar, and rough terrain application spaces.

We will focus on an exemplary case of Ackermann steered Wheeled Mobile Robot (WMR). The critical considerations for modeling and controller design come from the standpoint of chassis, dynamic load transfer, and, most importantly,

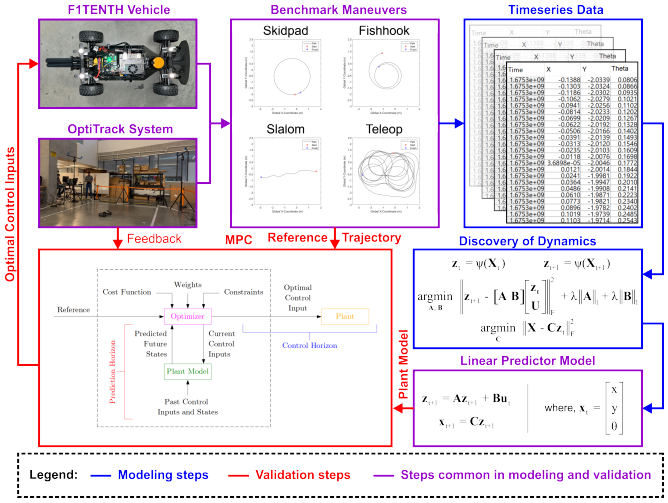


Fig. 1: Proposed approach involving timeseries data collection of benchmark maneuvers, discovery of dynamics using EDMD, identification of linear predictor model using Koopman operator, formulation of linear MPC [1], and experimental validation.

wheel-terrain interaction. In this milieu, there is rich literature highlighting vehicle models and controller design techniques at various levels of fidelity. While approximate linear models allow us to construct optimal and robust control solutions [4] suitable for real-time deployment, the unmodeled dynamics still play a significant factor in performance and safety. These unmodeled fluctuations may arise from multiple sources ranging from inadequate modeling of physics phenomena (e.g. wheel terrain interaction) to variable kinematic and dynamics parameters. High-fidelity non-linear models have sought to capture the dynamics and wheel-terrain interactions effectively but pose a challenge for controller design and added computational complexity for the predictive controllers considering real-time deployment [5]. Additionally, physics-based models do not scale down to small-mid scale WMRs well enough due to factors like tires, suspension and chassis not exhibiting similar mechanical and material properties. This makes developing high-fidelity models across varied robotic platforms even more challenging.

Traditional system identification and adaptive control approaches have been built upon a range of white- gray- or

*This work was supported by Clemson University's Virtual Prototyping of Autonomy Enabled Ground Systems (VIPR-GS), a US Army Center of Excellence for modeling and simulation of ground vehicles, under Cooperative Agreement W56HZV-21-2-0001 with the US Army DEVCOM Ground Vehicle Systems Center (GVSC)

[1]Department of Automotive Engineering, Clemson University International Center for Automotive Research (CU-ICAR), Greenville, SC 29607, USA. {ajoglek, csamak, tsamak, kkosara, vkrovi}@clemson.edu

[2]Department of Mechanical Engineering, Clemson University, Clemson, SC 29634, USA. {ssutava, uvaidya}@clemson.edu

black-box modeling techniques. However, newer approaches like neural network-based implicit modeling of the dynamics and end-to-end black box methods like deep imitation learning and reinforcement learning [6] can implicitly capture the dynamics and control the system behavior. While these methods provide for increased adaptability and have been demonstrated in path tracking applications, they face challenges like explainability, generalizability, and Sim2Real gap.

Hence, data-driven modeling and system identification has gained a lot of importance for merging benefits of leveraging optimization techniques for model discovery while enabling model analysis through their mathematical properties. In this milieu, a prominent paradigm is the Koopman Operator approach, where a finite-dimensional non-linear model can be converted to an infinite-dimensional linear model through the process of lifting. In the application domain, however, we obtain a finite-dimensional linear predictor through the provided data using the Extended Dynamic Mode Decomposition (EDMD) [7] method as highlighted in Section II. While these methodologies were traditionally applied for discovery of dynamics in the fields of chaotic systems and fluid flow analysis [8], studies like [9, 10, 11, 12, 13] highlight the application of the Koopman operator theory for optimal control and stabilization of robotic and dynamic systems. The conventional method is to gather data based on a kinematic/dynamic model-based simulation of the system, obtain a linear model based on the training data, and deploy it using a predictive/robust controller. In such scenarios, there often exists a Sim2Real gap and dependency on the controller and additional constraints to ensure stability. Additionally, the specific trajectories the model has been trained on may not effectively capture the excitation in dynamics.

In this study, we introduce a data-driven approach using the Koopman EDMD algorithm to capture the underlying dynamics of the F1TENTH scaled model vehicle [14]. We conduct a series of maneuvers known for exciting vehicle dynamics, providing insights into the platform's characteristics. To assess the performance of the derived model, we employ a linear Model Predictive Control (MPC) feedback law and compare it against the kinematic bicycle model with Nonlinear Model Predictive Control (NMPC) in a path-tracking problem.

II. Preliminary

This section introduces the mathematical formulation of the Koopman EDMD algorithm for non-linear systems and an optimization sub-routine for obtaining an approximated linear model based on lifted temporal snapshots of states/control measurements.

A. Koopman Operator Theory

Consider a non-linear system

$$\mathbf{x}_{t+1} = \mathbf{f}(\mathbf{x}_t, \mathbf{u}_t) \tag{1}$$

where $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^n$ and $\mathbf{u} \in \mathcal{U} \subseteq \mathbb{R}^m$; $\mathbf{f} : (\mathcal{X},\mathcal{U}) \to \mathcal{X}$ is the evolution operator. To leverage this dynamical system for linear prediction and controls, a higher dimensional lifting of the system can be obtained using a set of functions called the observable ($g(x)$). In this lifted space, the Koopman operator $\mathcal{K}$ is a linear operator governing dynamics evolution.

$$[\mathcal{K}g](x) = g \circ \mathbf{f}(x, u) \tag{2}$$

Theory of the Koopman operator and its spectral properties for uncontrolled and controlled dynamics can be found in [10, 8]. The Koopman operator is typically an infinite dimensional operator, but it captures all the nonlinearities of the underlying system in a linear fashion, which makes it a powerful tool for the analysis of autonomous nonlinear systems. While there are several tools and methods to obtain an infinite dimensional Koopman operator, we will focus on deriving a finite-dimensional linear predictor for time-series data for our study.

B. Koopman EDMD for Finite-dimensional Linear Predictor

Finite dimensional approximation of the Koopman operator has great practical significance as it helps us apply powerful techniques from the linear systems theory to the nonlinear systems. A well-constructed approximation of the Koopman operator can significantly reduce the complexities associated with the controls and synthesis and lend itself to use of proven control techniques like Linear Quadrtic Regulator (LQR) and Linear Model Predictive Controller (MPC). The Koopman EDMD approach allows us to obtain such finite dimensional approximation based on snapshots or measurements of data and control inputs. The linear approximation for a controlled system is given by:

$$\mathbf{\Psi}(x_{t+1}) = \mathbf{A}\mathbf{\Psi}(x_t) + \mathbf{B}u_t \tag{3}$$

Where $\mathbf{\Psi} = [\mathbf{\Psi}_1(x), \ldots, \mathbf{\Psi}_N(x)] \in \mathbb{R}^N$ is the functional basis in the lifted space. $\mathbf{A} \in \mathbb{R}^{N\times N}$ and $\mathbf{B} \in \mathbb{R}^{N\times M}$ are the finite dimensional approximations for the Koopman operator. The following subsection highlights the procedure to obtain these $\mathbf{A}$ and $\mathbf{B}$ matrices given temporal snapshots of state and control inputs.

C. Numerical Optimization for Linear Predictor

Consider the matrices $\mathbf{X}_t = [\mathbf{x}_0, \ldots, \mathbf{x}_{t-1}] \in \mathbb{R}^{n\times t}$ and $\mathbf{X}_{t+1} = [\mathbf{x}_1, \ldots, \mathbf{x}_t] \in \mathbb{R}^{n\times t}$ with columns as the snapshots of the measured state x at times $[0, \ldots, t-1]$ and $[1, \ldots, t]$ respectively, where $t \in \mathcal{Z}_+$. The $\mathbf{X}_{t+1}$ matrix is one time-step forward progression of the $\mathbf{X}_t$ matrix. Similarly, denote the controls matrix by $\mathbf{U} = [\mathbf{u}_0, \cdots, \mathbf{u}_{t-1}] \in \mathbb{R}^{m\times t}$, where each column denotes the respective control input at time $[0, \ldots, t]$.

Let $\mathbf{z}_t = \mathbf{\Psi}(\mathbf{X}_t)$ and $\mathbf{z}_{t+1} = \mathbf{\Psi}(\mathbf{X}_{t+1})$ denote the lifted states where $\mathbf{z}_t, \mathbf{z}_{t+1} \in \mathbb{R}^{N\times t}$ where $N >> n$. From Eq. (3), we have:

$$\mathbf{z}_{t+1} = \mathbf{A}\mathbf{z}_t + \mathbf{B}\mathbf{u}_t \tag{4}$$

The unknown matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ can be estimated numerically using the measured data as follows [10].

1) The matrices $\mathbf{A}$ and $\mathbf{B}$ are computed by solving the convex optimization problem

$$\underset{\mathbf{A},\mathbf{B}}{\arg\min} \left\| \mathbf{z}_{t+1} - [\mathbf{A}\ \mathbf{B}] \begin{bmatrix} \mathbf{z}_t \\ \mathbf{U} \end{bmatrix} \right\|_F^2 + \lambda_1 \left\| \mathbf{A} \right\|_1 + \lambda_2 \left\| \mathbf{B} \right\|_1 \tag{5}$$

where λ_1 and λ_2 are the regularization hyperparameter for imposing L_1 penalty on $\mathbf{A}$ and $\mathbf{B}$ matrices for promoting sparsity (LASSO regression).

2) Then, the matrix $\mathbf{C}$, representing the linear mapping from the lifted space to the original state space, is obtained as the projection of $\mathbf{\Psi}(\mathbf{x})$ onto $\mathbf{x}$ with least-squares optimization of the following form.

$$\underset{\mathbf{C}}{\arg\min} \left\| \mathbf{X} - \mathbf{C}\mathbf{z}_t \right\|_F^2 \tag{6}$$

III. Motivation and Problem Statement

This study presents an experimental framework for data-driven model identification and subsequent controls for trajectory tracking of a WMR. Traditionally, data driven discovery relies on simulation data which may not be truly representative of the underlying dynamics or real-world data, which in the case of a full-scale autonomous vehicle is prohibitively expensive.

Our proposed framework is implemented on the F1TENTH system, a scaled robotics platform widely used as a surrogate model of a full-scale autonomous vehicle [14]. This platform is equipped with an Ackermann-steered drive and NVIDIA ARM-based compute, designed for the development of autonomous driving capabilities. The F1TENTH scaled vehicle system, shown in Fig. 2a, utilizes a radio-controlled hobby car chassis capable of performing highly dynamic maneuvers for its scale. To capture the underlying dynamics, the data must record the dynamic excitation of the system. The following maneuvers, recognized benchmarks in the vehicle dynamics community, are employed to excite the dynamics and comprehend the system's characteristics.

- **Skidpad:** The skidpad is seemingly the simplest maneuver to follow, with a constant longitudinal velocity and steering angle. Still, it showcases the underlying dynamic behavior of the vehicle through the effect of lateral acceleration and body roll causing under/oversteer.
- **Fishhook:** The fishhook maneuver is used to test out the roll stability of the vehicle. It indicates the vehicle response to gradual changes in steering.
- **Slalom:** The slalom run (or the double lane change) tests the lateral stability of the vehicle and roll dynamics.
- **Teleop:** The teleop runs randomly excite the dynamics with unordered command inputs. This dataset is solely used for testing to validate the generalizability of the model and performance of the MPC.

Furthermore, OptiTrack motion capture (mo-cap) system consisting of 12 camera array with 0.2 mm measurement accuracy was exploited to stream and record the vehicle state data along with the vehicle control input data over a mutual ROS [15] interface between the vehicle's on-board computer and the OptiTrack computer in an indoor lab setting (refer Fig. 2b). Datasets were collected by throttling the topics for vehicle states and inputs at 60 Hz frequency to ensure synchronized recording of state-input pairs (refer Table I).

In this paper, we propose a framework to leverage the Koopman EDMD algorithm over this temporal data (snapshots) and identify a linear approximation of the Koopman operator. Furthermore, a feedback control law is implemented using the linear MPC for optimal path tracking.

IV. Methodology

This section highlights process flow for our data-driven model identification and control pipeline. An overview of this framework is highlighted in Fig. 1.

A. Data Collection

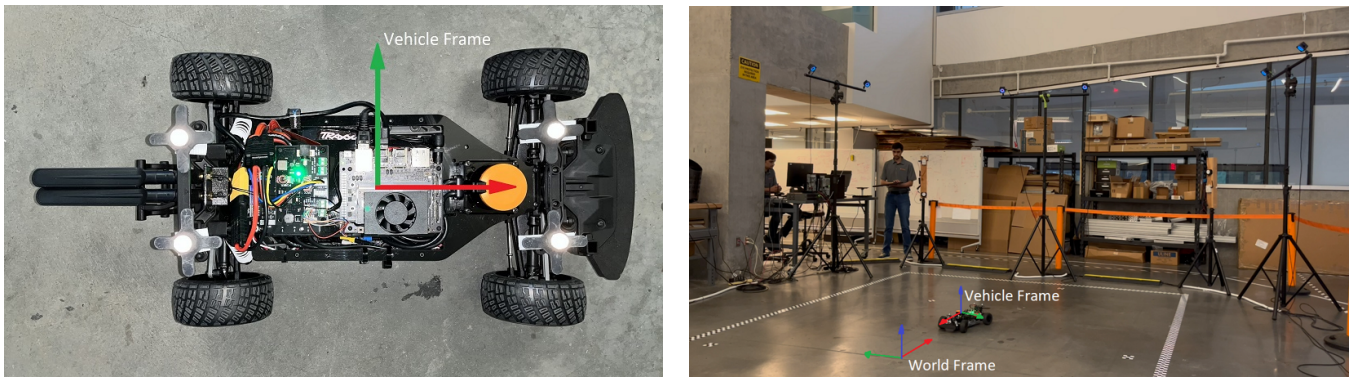
Intuitively, the first step for data-driven discovery of dynamics is collection of relevant experimental data indicative of the underlying vehicle dynamics. Consider the F1TENTH vehicle performing vehicle dynamics maneuvers inside the OptiTrack arena. The measured vehicle states include Cartesian coordinates for position x, y (in meters) and the vehicle's heading angle θ (in radians) with respect to static world frame. The control inputs applied to the vehicle on the other hand included linear velocity v (in meters per second) and steering angle δ (in radians), which is a common practice followed across various vehicle dynamics experiments for Ackermann-steered vehicles.

The variations in trajectory profiles of the vehicle were achieved using two techniques:

- **Nature of the maneuver:** Skidpad test drives the vehicle in a constant-radius circular trajectory by applying constant velocity and steering control inputs. Fishhook test drives the vehicle in a spiral trajectory, which can be achieved in one of the following ways: (i) keeping linear velocity constant and gradually increasing/decreasing the steering angle, or (ii) keeping constant steering angle and gradually decreasing/increasing the linear velocity; we used the former approach. Slalom test drives the vehicle in a sinusoidal trajectory by applying constant velocity and square-wave shaped steering control inputs.
- **Magnitude of control inputs:** Each of the maneuvers described earlier are characterized by the magnitude of control inputs being applied. The velocity and steering commands were therefore discretely varied in each case, as described in Table I.

B. Data Driven Model Discovery

1) Data Preprocessing: The training data from Section IV-A consists of 4 state and 2 input measurements. We have used randomly sampled trajectories from the skidpad, fishhook, slalom and maneuvers to train the model. Considering a total of t samples, the training data matrix can be represented as $\mathbf{X}_t = [\mathbf{x}_0, \ldots, \mathbf{x}_{t-1}] \in \mathbb{R}^{4\times t}$, $\mathbf{X}_{t+1} = [\mathbf{x}_1, \ldots, \mathbf{x}_t] \in \mathbb{R}^{4\times t}$ and $\mathbf{U}_t = [\mathbf{u}_0, \ldots, \mathbf{u}_{t-1}] \in \mathbb{R}^{2\times t}$ as highlighted in Section II-C.



(a) F1TENTH vehicle with 4 X-base passive reflective markers.

(b) F1TENTH vehicle performing slalom maneuver in mo-cap area.

Fig. 2: Experimental setup used for collection of time-synchronized state-input pairs of the physical F1TENTH vehicle using three distinct open-loop controllers and deployment of linear MPC on the vehicle using the Koopman operator based lifted model. The standard frame axes convention {X = red, Y = green, Z = blue} applies.

TABLE I: Parameter variation for collection of time-synchronized state-input pairs under a variety of trajectory profiles.

Maneuver	Velocity (m/s)	Steering Angle (rad)	Sampling Frequency (Hz)	Sync Tolerance (s)	Data Samples (#)
Skidpad	{0.5, 1.0, 1.5, 2.0, 2.5}	{0.312, 0.416, 0.520}	60	5e-5	5830
Slalom	{0.5, 1.0, 1.5, 2.0, 2.5}	{0.312, 0.416, 0.520}	60	1e-4	3033
Fishhook	{0.5, 1.0, 1.5, 2.0, 2.5}	{0.104, 0.208, 0.312, 0.416, 0.520}	60	4e-5	2412

2) Candidate Basis Function: The lifted dynamics, as shown in Eq. (3), are obtained through analytical construction of the basis function $\mathbf{\Psi}$ using the Dubins car model formulation. The seminal work on the Dubin's car model [16] and its extensions incorporating controls, as presented by Lamiraux et.al [17], offer a formulation for smooth paths between two points for a car-like robot. The robot model, utilizing the Dubin's car model with control input, can be expressed as:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{v} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} v\cos\theta \\ v\sin\theta \\ a \\ \omega \end{bmatrix} \tag{7}$$

Considering Eq. (7) and our observed state measurements $\mathbf{x}_t$, a 17 function candidate library is selected consisting product of $\sin\theta$ and $\cos\theta$ with the velocity v estimated at each data point based on x and y coordinates. The additional lifted functions are of the form $v^n\cos\theta$, $v^n\sin\theta$ for $n = 1, \cdots, 3$. We define $\mathbf{z}_t := \mathbf{\Psi}(\mathbf{x}_t)$ as the state vector in the lifted space of aforementioned library functions, i.e., $\mathbf{z}_t = \begin{pmatrix} 1 & x & y & v & v\cos\theta & v\sin\theta & \dots & v^3\cos\theta & v^3\sin\theta \end{pmatrix}^\top$.

3) Linear Approximation using Koopman EDMD: The lifted state measurements $(\mathbf{z}_t, \mathbf{z}_{t+1})$ and control inputs $(\mathbf{u}_t)$ are plugged into Eq. (5) and a convex optimization subroutine yields the $\mathbf{A}$ and $\mathbf{B}$ matrix. These are the finite dimension Koopman approximation for propagation of lifted state dynamics. This EDMD model is further used for prediction and design of MPC based control law.

C. Linear MPC for Trajectory Tracking

Our objective is to attain a control law enabling accurate trajectory tracking using a Quadratic Programming solver for fast evaluation of control inputs. Considering our Koopman lifted predictor as our discrete dynamical system:

$$\begin{aligned} \mathbf{z}_{t+1} &= \mathbf{A}\mathbf{z}_t + \mathbf{B}\mathbf{u}_t \\ \mathbf{x}_{t+1} &= \mathcal{C}\mathbf{z}_{t+1} \end{aligned} \tag{8}$$

Where original coordinate space of the dynamics as described in Eq. (7), the lifted space in $\mathbf{z_t}$ manifests the spatial transformation defined by the lifting function $\Psi(x_t)$. The proposed controller solves a closed-loop quadratic optimization at each time given by:

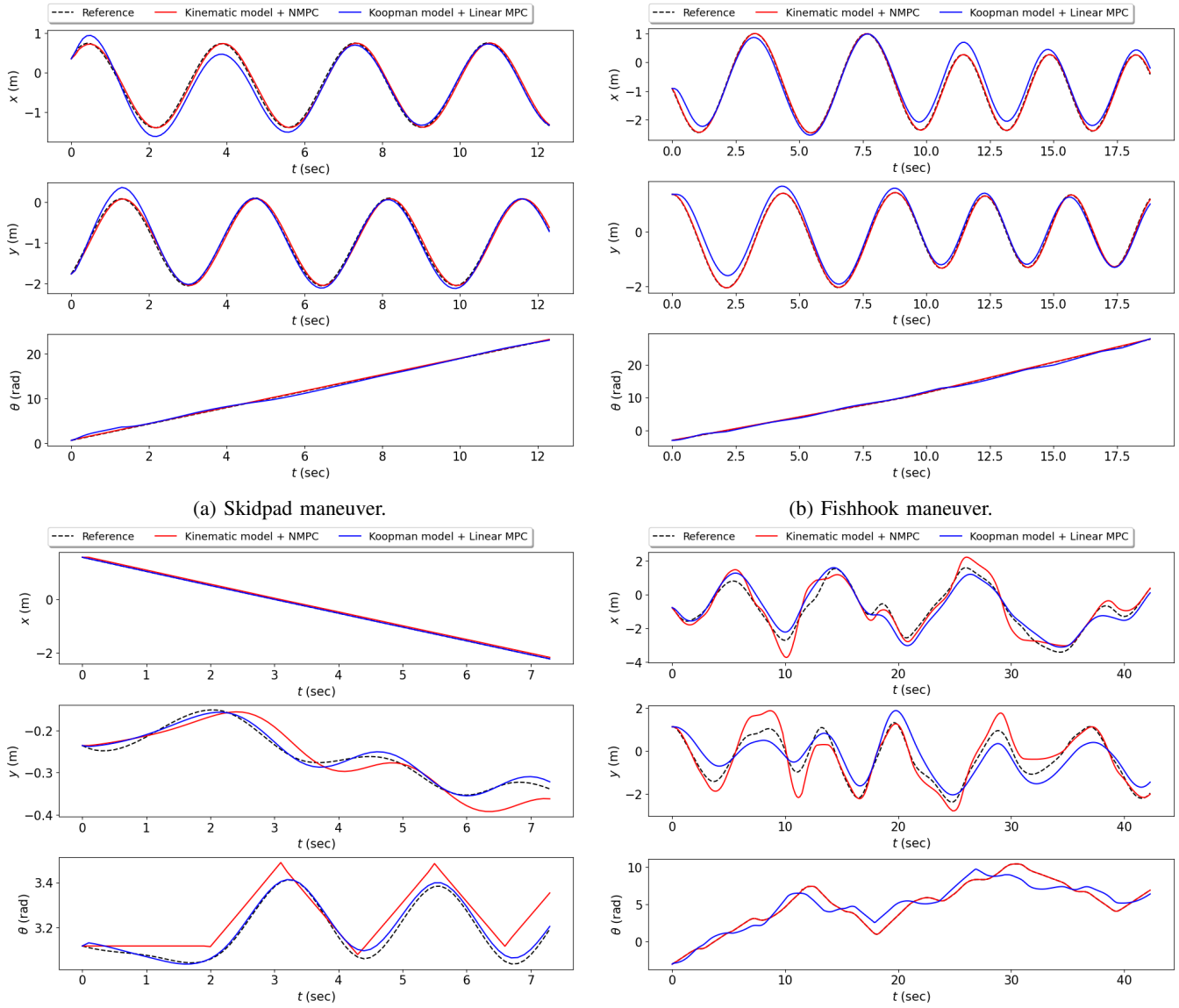
$$\begin{aligned} \underset{u_t, z_t}{min} \sum_{t=1}^{t=N_p} & \left(\mathcal{C}\mathbf{z_t} - \mathcal{C}\mathbf{z_t^{ref}}\right)^\top Q \left(\mathcal{C}\mathbf{z_t} - \mathcal{C}\mathbf{z_t^{ref}}\right) \\ & + \left(\frac{\mathbf{u_t} - \mathbf{u_{t-1}}}{\Delta t}\right)^\top P \left(\frac{\mathbf{u_t} - \mathbf{u_{t-1}}}{\Delta t}\right) \end{aligned} \tag{9a}$$

subject to

$$\mathbf{u_{min}} \leq \mathbf{u_t} \leq \mathbf{u_{max}} \ \forall t \tag{9b}$$

$$\mathcal{C}\mathbf{z}_0 = [x_0, y_0, v_0, \psi_0]^\top \tag{9c}$$

The objective function is designed to penalize the position error relative to the trajectory window, represented as $([\mathcal{C}z_{ref} - \mathcal{C}z_{pred}])$, with $(\mathcal{C}z_{pred})$ denoting the predicted position of the robot. Moreover, to mitigate rapid changes in control inputs that may cause instability, the rate of change of the control input $\mathbf{u(t)}$ is also penalized, considering actuator inertia. The MPC hyperparameters can be found in [18].



(a) Skidpad maneuver.

(b) Fishhook maneuver.

(c) Slalom maneuver.

(d) Teleoperation maneuver.

Fig. 3: Experimental validation for a variety of trajectory profiles: (a) skidpad, (b) fishhook, (c) slalom, and (d) teleoperation.

V. Results and Discussion

This section presents a comprehensive path-tracking comparison between two models: a benchmark kinematic bicycle model with a non-linear model predictive controller (NMPC) and the derived Koopman linear model with the designed linear model predictive controller (MPC). The evaluation is performed using a test data set comprising randomly sampled trajectories from skidpad, fishhook, slalom and teleoperation maneuvers. Notably, these trajectories were not previously utilized during the training process, making this evaluation crucial for assessing the generalizability of our linear predictor and the efficacy of the MPC feedback law in achieving accurate tracking. The path-tracking performance of both controllers is illustrated in Fig. 3, depicting their responses along the mentioned trajectories. Table II presents a comprehensive summary of the tracking errors. The research findings reveal that during the Skidpad and Fishhook maneuvers, there is limited excitation in the roll and yaw plane dynamics, leading to steady-state conditions, especially at slower speeds. Consequently, both approaches achieve similar tracking performance, as the vehicle's motion can be effectively captured using the kinematic model. However, in the case of the Slalom maneuver, the continuous excitation of system dynamics reveals limitations in tracking performance when relying on the kinematic model with NMPC feedback controller. In contrast, the Koopman EDMD model combined with linear MPC demonstrates markedly superior performance, indicating its enhanced capability to accurately capture the underlying dynamics more effectively. This phenomenon is further observed during the teleop maneuver, which serves as a crucial test to assess the generalizability of our proposed approach. The presence of random dynamic

excitation in this scenario can potentially lead to specific instabilities in the system behavior, highlighting the importance of evaluating the robustness and adaptability of our method under diverse and unpredictable conditions. These results underscore the potential advantages of utilizing the Koopman EDMD model and linear MPC for enhanced path-tracking control in challenging maneuvers.

TABLE II: Comparison of tracking performance for various test maneuvers.

Maneuver	Control	x-MSE (m)	y-MSE (m)	θ-MSE (rad)
Skidpad	Koopman + MPC	0.019	0.007	0.09
	Kinematic model + NMPC	0.0036	0.0035	0.008
Slalom	Koopman + MPC	2.4e-6	7.5e-5	3.3e-4
	Kinematic model + NMPC	2.5e-3	7.0e-4	5.7e-3
Fishhook	Koopman + MPC	0.046	0.01	0.11
	Kinematic model + NMPC	0.019	0.014	0.006
Teleop	Koopman + MPC	0.15	0.26	1.14
	Kinematic model + NMPC	0.166	0.38	7.97e-5

VI. Conclusion

The primary contribution of our study is the data-driven framework for the discovery of underlying vehicle dynamics in form of a linear predictor model. As the scaled vehicle undergoes maneuvers like skidpad, fishhook, and slalom, the excitation of dynamics is captured under an indoor localization system with temporal snapshots of state and action sequences. Based on these snapshots, a linear model of the system is identified using the Koopman EDMD approximation. The realized linear model can now leverage the widely accepted linear controller design methodologies like MPC while being computationally efficient for edge-device deployment. The performance of the model and controller design is highlighted using the tracking capabilities across testing data.

References

[1] Chinmay Vilas Samak, Tanmay Vilas Samak, and Sivanathan Kandhasamy. "Control strategies for autonomous vehicles". In: *Autonomous Driving and Advanced Driver-Assistance Systems (ADAS)*. CRC Press, 2021, pp. 37–86.

[2] R. Craig Coulter. *Implementation of the Pure Pursuit Path Tracking Algorithm*. Tech. rep. CMU-RI-TR-92-01. Pittsburgh, PA: Carnegie Mellon University, Jan. 1992.

[3] James C Alexander and John H Maddocks. "On the kinematics of wheeled mobile robots". In: *The International Journal of Robotics Research* 8.5 (1989), pp. 15–27.

[4] Felipe Kühne, Walter Fetter Lages, and João Manoel Gomes da Silva. "Model Predictive Control of a Mobile Robot Using Linearization". In.

[5] Tiago P Nascimento, Carlos Eduardo Trabuco Dórea, and Luiz Marcos G Gonçalves. "Nonlinear model predictive control for trajectory tracking of nonholonomic mobile robots: A modified approach". In: *International Journal of Advanced Robotic Systems* 15.1 (2018), p. 1729881418760461.

[6] Ajinkya Joglekar et al. "Hybrid Reinforcement Learning based controller for autonomous navigation". In: *2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring)*. 2022, pp. 1–6.

[7] Matthew O Williams, Ioannis G Kevrekidis, and Clarence W Rowley. "A data–driven approximation of the koopman operator: Extending dynamic mode decomposition". In: *Journal of Nonlinear Science* 25 (2015), pp. 1307–1346.

[8] Igor Mezić. "Analysis of fluid flows via spectral properties of the Koopman operator". In: *Annual Review of Fluid Mechanics* 45 (2013), pp. 357–378.

[9] Ian Abraham, Gerardo De La Torre, and Todd D. Murphey. "Model-based control using koopman operators". In: *Robotics*. Ed. by Nancy Amato et al. Robotics: Science and Systems. United States: MIT Press Journals.

[10] Milan Korda and Igor Mezić. "Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control". In: *Automatica* 93 (2018), pp. 149–160. ISSN: 0005-1098.

[11] Yiqiang Han, Wenjian Hao, and Umesh Vaidya. "Deep Learning of Koopman Representation for Control". In: Dec. 2020, pp. 1890–1895.

[12] Umesh Vaidya. "Spectral Analysis of Koopman Operator and Hamilton Jacobi Equation". In: *2022 61st IEEE Conference on Decision and Control, in press*. IEEE. 2022, pp. 1–6.

[13] Bowen Huang and Umesh Vaidya. "A convex approach to data-driven optimal control via perron-frobenius and koopman operators". In: *IEEE Transactions on Automatic Control* (2022).

[14] Matthew O'Kelly et al. "F1/10: An Open-Source Autonomous Cyber-Physical Platform". In: (2019).

[15] *ROS: an open-source Robot Operating System*. Vol. 3. Jan. 2009.

[16] Lester E. Dubins. "On Curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents". In: *American Journal of Mathematics* 79 (1957), p. 497.

[17] Florent Lamiraux and Jean-Paul Laumond. "Smooth motion planning for car-like vehicles". In: *Robotics and Automation, IEEE Transactions* 17 (Sept. 2001), pp. 498–501.

[18] Codebase for Data-Driven Modeling and Experimental Validation of Autonomous Vehicles using Koopman Operator. `https://github.com/ajinkya-joglekar/F1Tenth_KEDMD_Dubin_VIPR`.