

Real-Time Learning of Predictive Dynamic Obstacle Models for Robotic Motion Planning

Stella Kombo¹, Masih Haseli¹, Skylar X. Wei², and Joel W. Burdick¹

Abstract—Autonomous systems often must predict the motions of nearby agents from partial and noisy data. This paper asks and answers the question: "can we learn, in real-time, a nonlinear predictive model of another agent's motions?" Our online framework denoises and forecasts such dynamics using a modified sliding-window Hankel Dynamic Mode Decomposition (Hankel-DMD). Partial noisy measurements are embedded into a Hankel matrix, while an associated Page matrix enables singular-value hard thresholding (SVHT) to denoise the Hankel matrix and estimate its rank. A Cadzow projection enforces structured low-rank consistency, yielding a denoised trajectory and local noise variance estimates. From this representation, a time-varying Hankel-DMD *lifted* linear predictor is constructed for multi-step forecasts. The residual analysis provides variance-tracking signals that can support downstream estimators and risk-aware planning. We validate the approach in simulation under Gaussian and heavy-tailed noise, and experimentally on a dynamic crane testbed. Results show that the method achieves stable variance-aware denoising and short-horizon prediction suitable for integration into real-time control frameworks.

I. INTRODUCTION

Autonomous robotic systems often operate in the presence of other dynamic and non-coordinated agents. For example, autonomous cars must navigate around vehicles, pedestrians, and cyclists [1], [2]. In autonomous drone racing, drones must avoid crashing into other drones on the racecourse [3], [4] and in maritime robotics, a shipboard robotic arm or autonomous crane must plan for the payload's motion while compensating for the ship's sea-induced oscillations. In such settings, the other agent's dynamics and intentions are typically unknown, while onboard sensing provides noisy, partial observations of the agent motions. Safe, efficient behavior therefore hinges on accurate short-horizon prediction of agent motion under uncertainty to enable collision-free planning and real-time control. Thus, this paper introduces a data-driven framework for real-time learning and short-horizon prediction from noisy, partial observations.

Related Work: Different approaches have been developed for integrating prediction and planning in dynamic environments where other agents' dynamics and intentions are unknown. Classical geometric planners such as the Velocity-Obstacle (VO) family [5], Reciprocal VO (RVO), Optimal Reciprocal Collision Avoidance (ORCA) and Acceleration Velocity Obstacles extensions (AVO)) [6]–[9], embed simple

relative-motion estimates to predict collisions and select controls outside forbidden sets. These methods are efficient but assume simplified behavior models and near-perfect state estimates. However, real agents often violate these assumptions. For instance, some thrown objects may be roughly ballistic [10], but aerodynamically complex bodies, e.g., frisbees, do not follow a simple ballistic model. Additionally, behavior-driven agents, e.g., pedestrians, require data-driven intent models to capture variability beyond constant-velocity assumptions [11]. Lastly, the future trajectories of constraint-driven agents, e.g., autonomous vehicles, may be restricted by road geometry and traffic rules. Across all these examples, VO-style methods assume perfect knowledge of the moving agents' states and neglect measurement noise or sensing delays [12]. This reliance on idealized assumptions highlights the gap between geometric formulations and the stochastic, nonlinear realities of robotics applications.

When other agents' dynamics are unknown, it may be desirable to learn predictive models online from streaming observations. In practice, only partial and noisy measurements are available, while key latent states are unobserved, degrading forecasts and the safety of plans. Prior methods assume full-state observability for simplicity [13]. This includes frameworks that learn dynamics directly from data *offline*, e.g., classical Autoregressive Moving Average (ARMA) models [14], modern sequence learners such as Recurrent Neural Networks (RNNs), transformers, and tokenized representations [15]. These off-line methods capture complex behavior, but require large datasets and retraining [16] and adapt poorly to distribution shift [17]. *Online* variants (e.g., Kalman filtering, recursive least squares, and online Gaussian processes [18]) update models during deployment, but presume known or structured noise distributions [19].

Koopman theory is a complimentary alternative to learning these dynamic models. Formally, the Koopman operator models a nonlinear dynamical system via a *linear operator* on a vector space of functions [20]. Since the operator acts on an infinite dimensional vector space, real-time analysis and prediction is intractable. A practical approach is to *approximate* the operator's action on a finite-dimensional subspace (which is typically of greater dimension than the state-space of the nonlinear system), enabling the use of well-developed methods from linear algebra and linear system theory. These approximations are often performed via orthogonal projections on the subspace of choice, also referred to as truncations of the operator's action. Dynamic Mode Decomposition (DMD) [21] and its variant Extended-DMD (EDMD) [22] are well-known examples of such projection-

*This work was supported by the Defense Advanced Research Projects Agency (DARPA) under the LINC program.

¹S. Kombo, M. Haseli, and J. W. Burdick are with the Division of Engineering and Applied Science, California Institute of Technology, Pasadena, CA 91125, USA. {skombo, mhaseli, jburdick}@caltech.edu

²S. X. Wei is with Applied Intuition, Mountain View, CA 94041, USA. skylar@applied.com

based algorithms. Hankel-DMD uses time-delay embedding to estimate an effective state-space dimension [23], [24]. Such truncated Koopman-based models offer computationally feasible approximations of nonlinear dynamics with proven utility in robotics for prediction and control [25], [26]. However, sensor measurements may be noisy, and since the fidelity of real-time Koopman predictors depends on data quality, noise degrades these learned models.

Existing denoising frameworks to resolve noisy measurements prior to model fitting include classical low-rank denoising techniques i.e., truncated Singular Value Decomposition (tSVD), Principal Component Analysis (PCA) [27] and Proper Orthogonal Decomposition (POD) [28]. They provide optimal projections but rely on *ad hoc* rank selection. Structured Hankel denoising [29] enforces temporal consistency but requires tuning and degrades under non-Gaussian noise. Scalable variants like randomized SVD [30] improve efficiency but depend on problem-specific thresholds, while Eigensystem Realization Algorithm (ERA) [31] and Singular Spectrum Analysis (SSA) [32] remain sensitive to noise.

Contribution: We propose an adaptive, denoising, sliding-window Hankel-DMD framework for real-time prediction from noisy partial measurements. At each time step, a finite data buffer fits a local model, balancing responsiveness to non-stationary behavior with enough samples for reliability. We first denoise via Cadzow’s low-rank projection, with rank \hat{r} chosen by Singular Value Hard Thresholding (SVHT) on a Page matrix embedding. We prove that, under mild conditions, Page and Hankel matrices have the same finite-sample rank, so the SVHT rank transfers to the Hankel matrix for Cadzow denoising. The resulting denoised Hankel matrices yield a sequence of local models and noise-variance estimates for uncertainty-aware planning. We validate the framework in both simulation and hardware experiments under Gaussian and non-Gaussian correlated heavy-tailed noise. We show that the framework offers robust denoised predictions suitable for real-time control and planning tasks.

Notation: \mathbb{R} and \mathbb{N} denote real and natural numbers. I_n and $0_{m \times n}$ denote the $n \times n$ identity and $m \times n$ zero matrices, respectively. For matrix $A \in \mathbb{R}^{m \times n}$, A^\top , A^\dagger , and $\text{rank}(A)$ denote its transpose, Moore–Penrose pseudo-inverse, and rank respectively. We denote by $\text{diag}(v)$, the $n \times n$ diagonal matrix with elements of $v \in \mathbb{R}^n$ on its main diagonal. All linear combinations of vectors $\{v_1, \dots, v_k\} \subset \mathbb{R}^n$ is denoted by $\text{span}\{v_1, \dots, v_k\}$. We denote the variance of random variable X by σ_X^2 and drop the subscript when the context is clear. For functions f and g with matching domain and co-domain, $f \circ g(x) := f(g(x))$ denotes their composition.

II. PROBLEM DEFINITION

Consider a robot that operates in the vicinity of another moving entity \mathcal{O} whose behavior and dynamics are unknown. The robot’s onboard sensors detect \mathcal{O} and provide noisy, partial measurements of the obstacle state, denoted by $x_t \in \mathbb{R}^{n_x}$, which are sampled at a uniform interval $\Delta t \geq 0$, and capture kinematic quantities such as velocities or angular rates. While x_t may not directly encode full

Cartesian position, it carries sufficient temporal information to enable short-horizon motion prediction. The motion of \mathcal{O} is governed by a discrete-time dynamical system with latent (unobserved) variables and unmeasured control inputs in the form: $z_{t+1} = \hat{f}(z_t, u_t)$ where z_t denotes the state, which may include latent (unobserved) variables and u_t an input from an unknown policy $u_t = c(z_t)$. Under these assumptions, we have an autonomous system

$$\begin{aligned} z_{t+1} &= f(z_t), \\ x_t &= C z_t + \eta_t, \end{aligned} \quad (1)$$

where measurement noise $\eta_t \in \mathbb{R}^{n_x}$ has an unknown distribution, and C is the state-output map. This discrete time model generally arises from discretization of a continuous-time system. We maintain a sliding buffer of N recent measurements $\mathcal{B}_t = \{x_{t-N+1}, \dots, x_t\}$ for online processing.

Problem Statement: Given streaming noisy observations $\{x_t\}$ from a single trajectory and a sliding data buffer of fixed length N , construct, in real time, a tractable and adaptive local model. This model generates multi-step forecasts $\{\bar{x}_{t+1}, \dots, \bar{x}_{t+N_h}\}$ over a prediction horizon $N_h \in \mathbb{N}$, suitable for downstream planning and control.

III. PRELIMINARIES

Here, we define key concepts that are leveraged in our method such as Hankel and Page embeddings, Hankel-DMD, Cadzow denoising, and Singular Value Hard Thresholding.

A. Hankel and Page Matrices

Guided by Takens’ delay-embedding theorem (the dynamics of a system can be reconstructed from delay-coordinate maps of a single observable when the embedding dimension is sufficiently large [33]), we apply delay embedding to the obstacle measurements to extract informative information about its dynamics. Given a sequence of N measurements $\{x_{i-N+1}, \dots, x_i\} \subset \mathbb{R}^{n_x}$, we denote the associated (block) Hankel matrix $H_{i-N+1:i}^L \in \mathbb{R}^{(Ln_x) \times (N-L+1)}$ by:

$$H_{i-N+1:i}^L = \begin{bmatrix} x_{i-N+1} & x_{i-N+2} & \cdots & x_{i-L+1} \\ x_{i-N+2} & x_{i-N+3} & \cdots & x_{i-L+2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{i-N+L} & x_{i-N+L+1} & \cdots & x_i \end{bmatrix}. \quad (2)$$

The (block) Page¹ matrix $P_{i-N+1:i}^L \in \mathbb{R}^{(Ln_x) \times m}$ partitions the same sequence into m non-overlapping blocks of length L (with $m = N/L$):

$$P_{i-N+1:i}^L = \begin{bmatrix} x_{i-N+1} & x_{i-N+L+1} & \cdots & x_{i-L+1} \\ x_{i-N+2} & x_{i-N+L+2} & \cdots & x_{i-L+2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{i-N+L} & x_{i-N+2L} & \cdots & x_i \end{bmatrix}. \quad (3)$$

¹For $n_x = 1$, the matrices $H_{i-N+1:i}^L$ and $P_{i-N+1:i}^L$ take Hankel and Page forms, respectively. For $n_x > 1$, they become block Hankel and block Page. For simplicity, we refer to both cases as Hankel and Page structures.

²Note that, unlike the Hankel matrix, in the Page matrix, the buffer length N must be a multiple of the embedding window length L .

B. Hankel-DMD

We review a variant of Hankel-DMD (cf. [23], [24]) suitable for our problem. It uses delay embedding on the measurements of obstacle's observables (1).

Given a measurement sequence $\{x_0, x_1, \dots, x_n\}$, form the one step shifted Hankel matrices $H_{0:n-1}^L$ and $H_{1:n}^L$ according to the Eq. (2). Note that the columns of $H_{1:n}^L$ are shifted one time step forward from the columns of $H_{0:n-1}^L$. To estimate the one-step dynamic prediction propagator, A^* , Hankel-DMD relies on the following least Frobenius-norm problem,

$$A^* = \arg \min_A \|H_{1:n}^L - AH_{0:n-1}^L\|_F. \quad (4)$$

The closed form solution for A^* is

$$A^* = H_{1:n}^L (H_{0:n-1}^L)^\dagger. \quad (5)$$

Under standard assumptions of ergodicity and noise-free observables, Hankel-DMD can be used as a finite-dimensional approximation of the Koopman operator via delay embeddings [23]. Our goal is to tackle the real-world cases where these assumptions do not hold.

C. Cadzow Algorithm for Denoising

Here, we review a variant of Cadzow's algorithm for denoising Hankel matrices [34]. In the approximately linear *lifted* dynamics (e.g., Koopman-based approximations), noise-free data resides on a low-dimensional vector space. Hence, for a sufficiently large delay embedding window L , the noise-free Hankel trajectory matrix becomes low rank. However, with measurement noise, the observed Hankel matrix is generally full rank. Let Hankel matrix $H^L \in (\mathbb{R}^{n_x}) \times (N-L+1)$ be constructed from N consecutive noisy measurements with embedding window L . Decompose H^L as $H^L = \hat{H}^L + \Delta$ with

$$\text{rank}(\hat{H}^L) = r < \min\{Ln_x, N-L+1\} \quad (6)$$

where Δ denotes measurement noise and \hat{H}^L is the noise-free Hankel matrix. The goal is to recover the (unknown) noise-free, low-rank Hankel matrix \hat{H}^L from H^L . Cadzow's algorithm aims for this objective by alternating projections of H^L on the set of matrices with rank of at most r with projections of the result back onto the set of Hankel matrices.

Given a target rank r , define the projection map Π_r onto the set of rank- $\leq r$ matrices as

$$\Pi_r(H) := \arg \min_{\text{rank}(Y) \leq r} \|H - Y\|_F^2 = U \Sigma_r V^\top \quad (7)$$

where $H = U \Sigma V^\top$ is the SVD and Σ_r is created by keeping the top r singular values in Σ while setting the rest to zero. The closed-form solution of (7) is a direct consequence of Eckart–Young–Mirsky theorem [35]. Similarly, we define the projection map Π_H onto the set of Hankel matrices as

$$\Pi_H(M) := \arg \min_{Z \in \text{Hankel}} \|M - Z\|_F^2. \quad (8)$$

Interestingly, the optimization problem (8) has a closed-form solution which can be calculated by replacing the members

of each anti-diagonal of M with their average. The procedure is as follows: decompose M into L by $N - L + 1$ blocks similarly to the Hankel matrix³. And denote by $M_{i,j}$ the ij th block. For each anti-diagonal offset $d = 0, \dots, N-1$, define:

$$\mathcal{I}_d = \{(\ell, m) : 1 \leq \ell \leq L, 1 \leq m \leq N-L+1, \ell+m-2 = d\}$$

where $s_d = |\mathcal{I}_d|$ is number of elements in \mathcal{I}_d . Then, the anti-diagonal averaging can be computed as

$$(\Pi_H(M))_{\ell,m} = \frac{1}{s_{\ell+m-2}} \sum_{(i,j) \in \mathcal{I}_{\ell+m-2}} M_{i,j},$$

for $1 \leq \ell \leq L, 1 \leq m \leq N-L+1$. The Cadzow algorithm successively applies the projections Π_r and Π_H so that the sequence converges to a Hankel matrix, \hat{H}^L , with approximate rank r . After n iterations the Cadzow algorithm yields $[\Pi_H \circ \Pi_r]^n(H^L)$ where $[\Pi_H \circ \Pi_r]^n$ denotes the map created by composing $\Pi_H \circ \Pi_r$, n -times with itself.

Notably, since projections Π_r and Π_H can be computed in closed form, the Cadzow algorithm is an attractive choice for real-time denoising, as only a few iterations are sufficient to yield a reasonable signal-to-noise ratio.

D. Singular Value Hard Thresholding (SVHT)

Here we review a variant of Gavish and Donoho's result [36]. Consider a sequence (indexed by a) of noisy matrices $Y_a = X_a + Z_a/\sqrt{n_a}$ where $Y_a \in \mathbb{R}^{m_a \times n_a}$ denotes a noisy matrix, X_a the "true" low-rank matrix, and Z_a denotes the noise matrix whose entries are i.i.d. zero-mean, unit-variance entries with finite fourth moment. As $a \rightarrow \infty$, the aspect ratio m_a/n_a converge to $\beta \in (0, 1]^4$.

Given the SVD $Y_a = U \Sigma V^\top$ and a threshold τ , we approximate the true matrix X_a by $\hat{X}_a = U \Sigma_{\geq \tau} V^\top$ where $\Sigma_{\geq \tau}$ is formed by zeroing the subthreshold elements of Σ .

Gavish and Donoho showed that the Asymptotic Mean Squared Error (AMSE)-optimal threshold for the data singular values depends only on the matrix aspect ratio β via a constant $\lambda^*(\beta)$:

$$\lambda^*(\beta) = \sqrt{2(\beta+1) + \frac{8\beta}{(\beta+1) + \sqrt{\beta^2 + 14\beta + 1}}}. \quad (9)$$

For the noise standard deviation σ , the optimal threshold is:

$$\tau^* = \lambda^*(\beta) \sigma \sqrt{n_a}. \quad (10)$$

When σ is unknown, the optimal threshold can be estimated using a data-driven approach through the Marchenko–Pastur (MP) law [37], which gives the limiting eigenvalue distribution of $(1/n_a) Z_a Z_a^\top$ as $m_a/n_a \rightarrow \beta$:

$$p_\beta(\delta) = \frac{\sqrt{(\delta_+ - \delta)(\delta - \delta_-)}}{2\pi\beta\delta} \mathbf{1}_{[\delta_-, \delta_+] }(\delta),$$

with $p_\beta(\delta) = 0$ outside $[\delta_-, \delta_+]$. The median of the MP distribution, denoted by μ_β , depends only on β and provides

³Note that each block is a column vector with n_x rows similarly to Eq. (2)

⁴Following [36], a indexes a growing problem sequence with $Y_a \in \mathbb{R}^{m_a \times n_a}$, $m_a/n_a \rightarrow \beta$, and $Y_a = X_a + Z_a/\sqrt{n_a}$. Their AMSE-optimality results are asymptotic as $a \rightarrow \infty$.

a normalization for estimating the noise level σ . Taking ς_{med} as the median of the observed singular values, the data-driven AMSE-optimal threshold can thus be estimated as:

$$\tau^*(\beta) = \frac{\lambda^*(\beta)}{\sqrt{\mu_\beta}} \varsigma_{\text{med}}. \quad (11)$$

which remains asymptotically optimal under general white noise (i.i.d., zero mean, unit variance, finite fourth moment) with the same risk as in the Gaussian case [36, Sec. VI].

IV. ADAPTIVE HANKEL-DMD FOR NOISY DATA

We now address the problem outlined in Sec. II. To enable real-time obstacle dynamics learning and motion prediction, we consider the use of Hankel-DMD (cf. Sec. III-B). However, applying Hankel-DMD in real-time robotics applications presents two key challenges:

- **Sensor noise:** sensor signals are typically noisy, which contaminates matrices $H_{0:n-1}^L$ and $H_{1:n}^L$. In least-squares schemes such as (4), noise in both data matrices leads to biased or inconsistent estimators [38].
- **Ergodicity and long trajectory data requirements:** as noted in Sec. III-B, one must assume ergodicity and long data sequences to connect Hankel-DMD and the Koopman operator. Such assumptions rarely hold in robotics applications.

To tackle the noise issue, we introduce a Hankel matrix denoising scheme. We then propose an adaptive sliding-window variation of Hankel-DMD that continually updates the model. This approach ensures that the predictive model remains accurate as the object's state evolves over time.

A. Denoising Hankel Matrices

Consider obstacle output measurements $\mathcal{B}_t = \{x_{t-N+1}, \dots, x_t\}$ for buffer size N , collected up to time t . We construct a Hankel matrix $H_{t-N+1:t}^L$ from \mathcal{B} according to Eq. (2). Since the measurements are noisy, we seek to denoise the Hankel matrix $H_{t-N+1:t}^L$ via Cadzow's algorithm, as described in Sec. III-C. A key step in this algorithm is the low-rank projection Π_r in Eq. (7), which requires knowledge of the effective rank r of the noise-free Hankel matrix. In practice, this typically unknown rank must be estimated. An rank estimat error can cause over-smoothing (if underestimated) or noise amplification (if overestimated). To avoid these problems, we adopt a principled, data-driven rank selection strategy based on Singular Value Hard Thresholding (SVHT) (Sec. III-D). However, the SVHT framework assumes i.i.d. noise, an assumption that is violated in Hankel matrices due to repeated entries. To address this, we employ Page matrices (Eq. (3)), whose partition of the measurement buffer into non-overlapping blocks avoids correlation.

1) *Step I: Page-Hankel Rank Transfer:* As a first step toward estimating the rank of the noise-free Hankel matrix using SVHT on Page matrices, we show that, under mild conditions, the Page and Hankel embeddings of the *same* noise-free measurement buffer share the same rank.

Lemma 1 (Page–Hankel Rank Equivalence). *Consider a locally valid linear output model*

$$z_{k+1} = Az_k, \quad x_k = Cz_k,$$

with $z_k \in \mathbb{R}^{n_z}$, $A \in \mathbb{R}^{n_z \times n_z}$, and $C \in \mathbb{R}^{n_x \times n_z}$. Let $\{x_0 \dots, x_{N-1}\} \subset \mathbb{R}^{n_x}$ be the noise-free measurements generated by the system above from the initial condition z_0 . Let $L \geq n_z$ be the embedding window and let $N = dL$, with $d \geq L$. Define⁵ $B = (A)^L$ and let P^L and H^L be the block Page and Hankel matrices constructed via the data. Then, $\text{rank}(P^L) = \text{rank}(H^L)$ if

$$\text{span}\{z_0, Bz_0, \dots, (B)^{d-1}z_0\} = \mathbb{R}^{n_z}$$
⁶

Lemma 1 provides a convenient way to connect the rank of Page and Hankel embedding of noise-free measurement buffers. Before proceeding to rank estimation, we briefly explain that the conditions of Lemma 1 are mild. First, it is worth noting that with a sufficiently high sampling frequency, the system states associated with a given measurement buffer cluster within a small region of the state space, where a local linear model can accurately approximate the system's behavior. In addition, the condition $\text{span}\{z_0, Bz_0, \dots, (B)^{d-1}z_0\} = \mathbb{R}^{n_z}$ is generic⁷ and holds for almost all matrices B and vectors z_0 . Now, with Lemma 1 at our disposal, we can estimate the rank of the noise-free Hankel matrix associated with $H_{t-N+1:t}^L$ by applying SVHT on the Page matrix constructed with the same data sequence.

2) *Rank Estimation using SVHT:* To apply the SVHT (cf. Sec. III-D) on the Page matrix, the number of matrix rows should not exceed the number of columns. Therefore, for the rest of the paper, we consider the buffer length $N = mLn_x$, with $m \geq Ln_x$. Note that this choice already satisfies the size condition in Lemma 1. Now, construct the Page matrix $P_{t-N+1:t}^L \in \mathbb{R}^{Ln_x \times m}$ from the same data sequence used in $H_{t-N+1:t}^L$. By applying the SVHT formula in Eq.(11), on the Page matrix, we can estimate the singular value threshold τ^* and approximate the effective rank of $P_{t-N+1:t}^L$ as

$$\hat{r} = \#\{i \in \{1, \dots, Ln_x\} : \varsigma_i \geq \tau^*\} \quad (12)$$

where $\{\varsigma_i\}_{i=1}^{Ln_x}$ are the singular values of $P_{t-N+1:t}^L$.

It is important to note that the rank \hat{r} estimated via Page SVHT is not fixed but evolves as the sliding window advances. As the buffer shifts, the local data distribution changes, and the singular spectrum captures the instantaneous richness of the underlying dynamics. Consequently, small fluctuations or occasional variations in \hat{r} are to be expected. Far from being a drawback, this adaptivity ensures that the thresholding remains sensitive to regime shifts and transient behaviors, ultimately producing more reliable models when the system exhibits greater dynamical complexity.

Sec. III-D also yields a conservative local noise–variance estimate. Singular values below the cutoff in Eq. (11) are

⁵To avoid confusion with superscript L , we use $(A)^L$ to denote the matrix created by raising A to the power of L .

⁶The proof is deferred to the extended version [39]

⁷If the elements of A (with $B = (A)^L$) and z_0 in Lemma 1 are drawn from a reasonable distribution, the condition holds with probability one.

assumed to arise from noise, while those above the cutoff define the rank via Eq. (12). Let ς_{med} be the median singular value of the *Page* matrix with aspect ratio β and m columns. Let μ_β denote the median of the Marchenko–Pastur law [37] at aspect ratio β . The noise variance $\hat{\sigma}^2$ can be conservatively estimated as $\hat{\sigma}^2 \approx \varsigma_{\text{med}}^2 / \mu_\beta m$. Under the assumption of i.i.d. sensor noise, we approximate the delay–state noise covariance as $\hat{\nu} \approx \hat{\sigma}^2 I_{Ln_x}$. This approximation treats each delayed state as being perturbed by independent, isotropic noise, which may underestimate correlations in more structured noise processes. Uses for such variance information within a risk-aware framework are provided in [40].

3) *Cadzow Algorithm*: Using the rank estimate \hat{r} in Eq.(12) and Lemma 1, we apply Cadzow’s algorithm (Sec. III-C) on Hankel matrix $H_{t-N+1:t}^L$ with rank \hat{r} until we approximate a denoised Hankel matrix, denoted by $\hat{H}_{t-N+1:t}^L$.

B. Online Identification and Multi-Step Prediction

With the denoised Hankel matrix $\hat{H}_{t-N+1:t}^L$ obtained, we next model the system dynamics using Hankel-DMD. However, as mentioned earlier, to rigorously connect the dynamics learned by Hankel-DMD in Eqs.(4)-(5) to the system’s global behavior via a Koopman operator, one typically requires assumptions of system ergodicity and access to very long trajectories. Because these assumptions rarely hold in robotics applications, we apply an adaptive sliding window strategy to build a linear model for each buffer and update it as we receive new data. Let $\hat{H}_{t-N+1:t-1}^L$ and $\hat{H}_{t-N+2:t}^L$ be the Hankel matrices constructed from the first and last $N-L$ columns of $\hat{H}_{t-N+1:t}^L$ respectively. We then solve a least Frobenius-norm problem similar to Eq. (4) as:

$$\hat{A}_t = \arg \min_{A_t} \|\hat{H}_{t-N+2:t}^L - A_t \hat{H}_{t-N+1:t-1}^L\|_F \quad (13)$$

with the closed-form solution

$$\hat{A}_t = \hat{H}_{t-N+2:t}^L (\hat{H}_{t-N+1:t-1}^L)^\dagger. \quad (14)$$

We then define the local predictor dynamics as

$$\psi_{m+1} = \hat{A}_t \psi_m, \quad m \in \{t, t+1, \dots\}. \quad (15)$$

If we initialize the system (15) by setting ψ_t to be equal to the last column of $\hat{H}_{t-N+2:t}^L$ (which corresponds to the denoised version of the lifted state at the current time), we can forecast the future outputs of system (1) by running the predictor (15) and extracting the last block of the delay embedded state as

$$\bar{x}_{t+j} = D (\hat{A}_t)^j \psi_t^* \quad j \in 1, \dots, N_h, \quad (16)$$

where ψ_t^* is set to be the last column of $\hat{H}_{t-N+2:t}^L$ and matrix $D \in^{n_x \times Ln_x}$ is defined as $D = [0_{n_x \times (L-1)n_x}, I_{n_x}]$. At the next time step, we advance the buffer to $\mathcal{B}_{t+1} = \{x_{t-N+2}, \dots, x_{t+1}\}$ and repeat this procedure to produce a time-varying sequence of linear predictors $\{\hat{A}_t\}_{t \geq N}$. These predictors are well suited for Model Predictive Control (MPC) by providing linear predictions over the planning horizon. Algorithm 1 compiles the entire procedure.

Algorithm 1: Adaptive Sliding-Window Page–Hankel DMD Predictor

Input: Embedding window L

Buffer Length $N = mLn_x$ with $m \geq Ln_x$

Buffer $\mathcal{B}_t = \{x_{t-N+1}, \dots, x_t\}$ with $x_t \in \mathbb{R}^{n_x}$

Prediction horizon N_h

Output: Denoised Hankel $\hat{H}_{t-N+1:t}^L$

Model \hat{A}_t and Forecasts $\{\bar{x}_{t+1}, \dots, \bar{x}_{t+N_h}\}$

Noise covariance estimate $\hat{\nu}$

(1) **Page & SVHT:** Form $P_{t-N+1:t}^L$ per Eq. (3)

Compute SVD $P_{t-N+1:t}^L = U \text{diag}(\varsigma_i) V^\top$.

Set $\beta = (Ln_x)/m$ and $\varsigma_{\text{med}} = \text{median}\{\varsigma_i\}$

Set cutoff $\tau^* = \lambda^*(\beta)/\sqrt{\mu_\beta}$ via Eq. (11)

$\hat{r} \leftarrow \#\{i : \varsigma_i \geq \tau^*\}$ per Eq. (12)

(2) **Cadzow Algorithm for J Iterations:**

$\hat{H}_{t-N+1:t}^L \leftarrow [\Pi_H \circ \Pi_r]^J (H_{t-N+1:t}^L)$

(3) **Hankel Partitions and Predictor at t :**

$\hat{H}_{t-N+1:t-1}^L, \hat{H}_{t-N+2:t}^L$

$\hat{A}_t = \hat{H}_{t-N+2:t}^L (\hat{H}_{t-N+1:t-1}^L)^\dagger$ as per Eq. (14)

(4) **Prediction (N_h steps):** Let ψ_t^* be the last

column of $\hat{H}_{t-N+2:t}^L$. For $j = 1, 2, \dots, N_h$,

$\bar{x}_{t+j} = D (\hat{A}_t)^j \psi_t^*$ per Eq. (16)

(5) **Noise Variance Estimation:** With m Page

columns and MP median μ_β , set $\hat{\sigma}^2 = \varsigma_{\text{med}}^2 / (\mu_\beta m)$

and $\hat{\nu} = \hat{\sigma}^2 I_{Ln_x}$.

(6) **Slide Window:** $\mathcal{B}_{t+1} \leftarrow \{x_{t-N+2}, \dots, x_{t+1}\}$ and repeat (1)–(5).

C. Comparison with Alternative Online Prediction Methods

Classical online noise filtering approaches, such as the Kalman Filter and its nonlinear variants [41], assume parametric dynamics and structured noise models. While online Gaussian processes [42], [43] and neural sequence models [44], [45] provide strong predictive capability, they can require significant offline training and careful tuning for online deployment. In contrast, DMD [46] and EDMD [22] are operator-theoretic system identification methods that approximate the Koopman operator from data. These formulations do not account for measurement noise and can exhibit bias under finite, noisy datasets [47], [48], particularly when nonlinear lifting is used [49]. Although noise-aware formulations and regularized variants exist [50], these methods mainly improve dynamical consistency rather than provide a framework for denoising streaming measurements in finite-data regimes. Our proposed method explicitly performs structured low-rank denoising prior to short-horizon prediction. Designed for finite, streaming data with limited computational budgets, the framework combines adaptive rank selection with structured matrix embeddings to enable stable real-time performance.

V. SIMULATIONS AND EXPERIMENTS

We apply Algorithm 1 to validate our framework in simulations and real-world experiments. Our evaluation examines

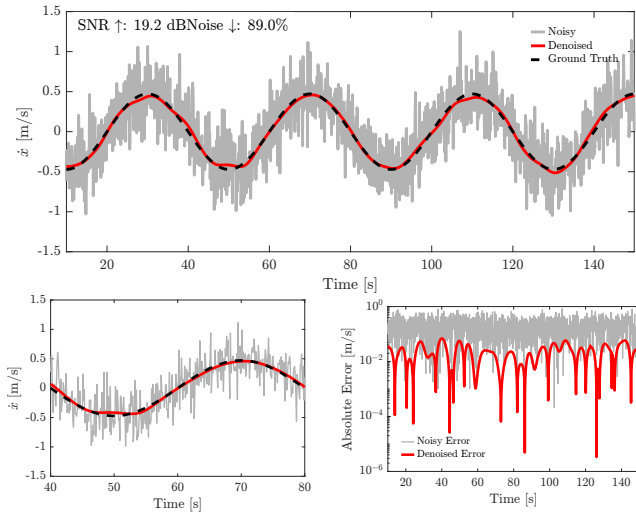


Fig. 1. Gaussian noise: Top: noisy (gray), denoised (red) and ground truth (black dashed); call-outs show SNR improvement and average noise reduction. Bottom: zoomed segment and log-scale error.

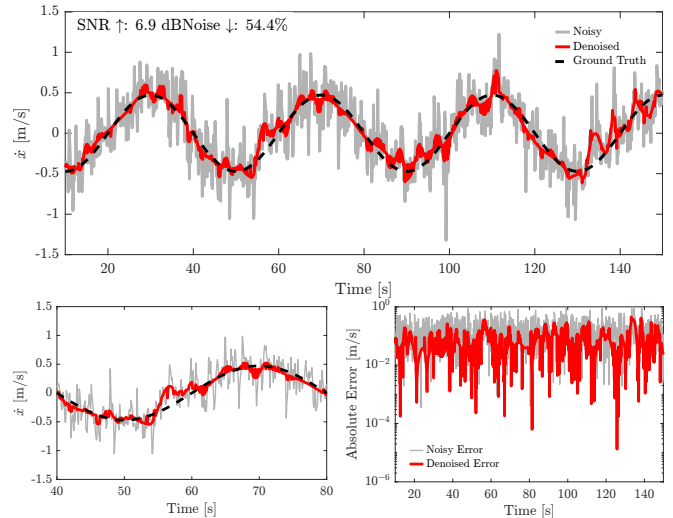


Fig. 2. AR(1)-Laplace noise: The denoising algorithm remains effective under heavy-tailed, correlated disturbances: call-outs: SNR +6.9 dB and 54.4% noise reduction.

(i) signal-to-noise (SNR) separability, (ii) noise variance estimation, and (iii) multi-step prediction accuracy. Experiments consider open-loop, short-horizon prediction to assess the accuracy, stability and noise robustness of the learned dynamics model. These properties are critical for downstream planning and control frameworks [51].

A. Simulation: Noisy Unicycle

Consider a reference unicycle model $\dot{x} = u_1 \cos \theta, \dot{y} = u_1 \sin \theta, \dot{\theta} = u_2$, [52], [53], which serves as a dynamic obstacle moving along a figure-eight trajectory of amplitude $a = 3$ m and period $T = 40$ s. An ego agent seeks to reach its goal while avoiding collisions with the unicycle, but has access to only noisy velocity measurements of the unicycle rather than the full state. Sampling the reference trajectory at $\Delta t = 0.02$ s yields the forward velocity profile $u_1^*(t)$. We inject sensor noise from two distributions: (i) i.i.d. Gaussian, $\eta_k \sim \mathcal{N}(0, \sigma_x^2)$ with $\sigma_x = 0.25$ m/s and (ii) correlated heavy-tailed AR(1)-Laplace, $\eta_t = \rho \eta_{t-1} + w_t$, $w_t \sim \text{Laplace}(0, b)$, $|\rho| < 1$. For fair comparison, b is chosen so that the stationary variance matches the Gaussian case, $\text{Var}[\eta_t] = \sigma_x^2$, i.e. $b = \sqrt{(1 - \rho^2)\sigma_x^2/2}$. In both cases, the noisy measurement stream is $v_t = u_1^*(t) + \eta_t$.

Denoising results (Gaussian): Fig. 1 demonstrates that our Page-Hankel SVHT framework achieves stable recovery of the ground-truth velocity profile under Gaussian sensor noise. The denoised estimate (red, top panel) tracks the ground truth (dashed black) without visible phase lag while suppressing high-frequency artifacts that can contaminate raw measurements. Over the 160 s trajectory, the method delivered an SNR gain of 19.2 dB and an average noise reduction of 89.0%, corresponding to nearly an order-of-magnitude improvement. The zoomed view (bottom-left) highlights that turning points and low-curvature segments, often degraded by conventional low-pass filters, were preserved, while the log-scale absolute error confirms a 10-100 \times reduction in residual magnitude. These results show that

the method can denoise aggressively without compromising structural features critical for downstream control.

Denoising results (Heavy tailed, correlated noise): To probe robustness under non-Gaussian disturbances, we injected temporally correlated AR(1)-Laplace noise (Fig. 2). Despite heavy tails and temporal correlation, we observed an SNR gain of 6.9 dB and an average noise reduction of 54.4%, with no visible phase lag. While performance is lower than under Gaussian noise, as to be expected, the degradation remains gradual. Crucially, the low-rank Hankel structure proved distribution-agnostic i.e., SVHT adaptively trims singular values inflated by heavy-tailed noise requiring explicit Gaussian assumptions on the noise statistics. For reference, we also evaluated an Extended Kalman Filter (EKF) as a denoising baseline. A detailed comparison and discussion are provided in [39].

B. System Hardware and Experiments

This section experimentally validates the complete method in a setup (Fig. 3) motivated by a naval application. Currently, payloads are loaded into their silos by a ship-board crane while the ship docks in a port, where no ocean swells affect the safety of the loading process. It would be desirable to load these payloads while the ship is underway, potentially in heavy seas. In active sea states, the ship sways in response to ocean swells. Anticipating deck motion via a learned predictive model would enable the crane controller to compensate during payload delivery.

The testbed is a scaled crane mounted on an *E2M eMove eM6-300-1500* five-DoF Stewart platform that can reproduce wave-induced ship deck motions. A VN-100 IMU measures orientation (quaternion) and angular velocity of the simulated moving base at 30 Hz. These measurements capture only the platform (deck) motion, modeled as a time-varying base input to the crane-payload system. From the IMU data, the framework generates a 31-step open-loop forecast (≈ 1.0 s) of the base motion, while crane and payload dynamics evolve downstream in response to the predicted input. This setup



Fig. 3. Stewart-platform testbed. **Left:** Moving-base crane with target, obstacle, payload, and arm/tip cameras. **Right:** Base-mounted VectorNav VN-100 IMU supplying orientation and angular rates to the sliding window Hankel-DMD predictor.

enables two evaluations: (i) whether the denoising observed in simulation persists under real IMU noise and (ii) whether short-horizon sliding window Hankel-DMD predictions are suitable for control integration. Fig. 4 (top) shows the deck x -coordinate over an $N = 250$ sample buffer (gray) which drives our predictive system to generate 31-step forecasts (red) aligned with the ground truth (dashed black). The open-loop predictor achieves an RMSE of 0.012 m/s. Given a fixed error-tolerance $\varepsilon = 0.04^8$, we define a violation-duration metric on the prediction error e_t as

$$J_t = \Delta t \sum_{t=1}^{N_h} \mathbf{1}(e_t \geq \varepsilon), \quad \text{with } e_t := \|\bar{x}_t - x_t\|_2$$

where N_h is the prediction horizon, \bar{x}_t the prediction forecast, x_t the ground truth, Δt the sample period, and $\mathbf{1}(\cdot)$ an indicator (1 if the condition holds, 0 otherwise). In Fig. 4 (bottom), $J_t = 89.0$ s (1.6% of horizon⁹), certifying that prediction errors are below tolerance for 98.4% of the time.

Further investigation of Fig. 4 (top) reveals occasional transient spikes in the predicted trajectory, after which the forecasts quickly return to a nominal accuracy. These anomalies arise when the local embedding window captures a system shift or abrupt disturbance, briefly mis-aligning the rank estimate. Importantly, such events are short-lived and self-correct as the sliding buffer refreshes, with the subsequent predictions re-stabilizing around the future measurements. This highlights both the adaptivity and the finite-sample sensitivity of the method, i.e., while transient outliers can occur, the framework recovers without persistent drift.

Fig. 5 illustrates the evolution of the eigenvalues of the learned predictors across sliding windows. As the system behavior changes along the trajectory, the models adapt accordingly, and the eigenvalues shift smoothly while consistently remaining within the unit circle, ensuring Schur stability. The gradual evolution of the spectrum highlights

⁸Hardware experiments show reliable insertion for errors up to 0.08 m. We therefore conservatively set $\varepsilon = 0.04$.

⁹The fraction of the horizon spent above threshold, %violating = $100 \times \frac{J_t}{T_{\text{hor}}}$, $T_{\text{hor}} := N_h \Delta t$.

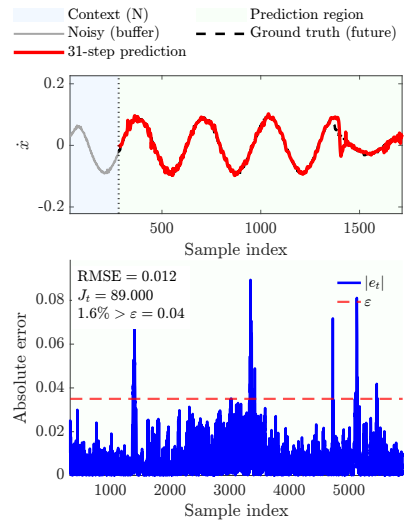


Fig. 4. Top: context buffer ($N = 250$, gray) and 31-step forecasts (red) overlaid with ground truth (black dashed). Bottom: absolute prediction error $|e_t|$ (blue) compared to threshold ε (red dashed).

the stability and time-varying nature of the local Hankel-DMD models throughout the experiment.

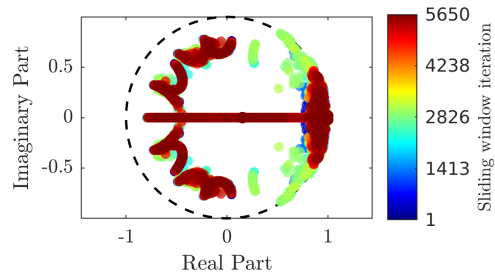


Fig. 5. Evolution of eigenvalue across sliding windows: full spectra with iteration-colored progression.

Together, these results demonstrate that our framework not only denoises real sensor streams but also yields stable and bounded predictions that are suitable for downstream planning and control applications. To further characterize the robustness of our approach, an ablation study revealed an accuracy-computation trade-off. Moderate Cadzow iterations ($J \approx 20$) and intermediate embedding lengths ($L \approx 10$ – 12) achieve near-optimal RMSE while maintaining bounded runtime, with diminishing returns beyond these regimes. A detailed analysis of hyper-parameter sensitivity and computational scaling is in the extended version [39].

VI. CONCLUSION AND FUTURE WORK

We introduced a Page–Hankel SVHT framework for learning short-horizon predictive models of moving obstacles from noisy data streams. Simulations (Gaussian and heavy-tailed noise) showed good variance-stable, distribution-agnostic performance, and hardware tests with IMU data produce reliable forecasts that can support downstream control. Future work will explore online adaptation, tensorized delayed embeddings for multi-axis consistency, and integration within an MPC-based prediction framework.

REFERENCES

- [1] M. Gulzar, Y. Muhammad, and N. Muhammad, "A survey on motion prediction of pedestrians and vehicles for autonomous driving," *IEEE Access*, vol. 9, pp. 137 957–137 969, 2021.
- [2] I. Yaqoob, L. U. Khan, and S. M. A. e. a. Kazmi, "Autonomous driving cars in smart cities: Recent advances, requirements, and challenges," *IEEE Network*, vol. 34, no. 1, pp. 174–181, 2019.
- [3] R. Spica, D. Falanga, E. Cristofalo, E. Montijano, D. Scaramuzza, and M. Schwager, "A real-time Game Theoretic planner for autonomous two-player drone racing," in *Robotics: Science and Systems*, 2018.
- [4] D. Hanover, A. Loquercio, L. Bauersfeld, A. Romero, R. Penicka, Y. Song, G. Cioffi, E. Kaufmann, and D. Scaramuzza, "Autonomous drone racing: A survey," 2023.
- [5] P. Fiorini and Z. Shiller, "Motion Planning in dynamic environments using Velocity Obstacles," *Int. J. Rob. Res.*, vol. 17, no. 7, pp. 760–772, 1998.
- [6] J. van den Berg, S. J. Guy, M. C. Lin, and D. Manocha, "Reciprocal n-body Collision Avoidance," in *Robotics Research (ISRR)*, ser. Springer Tracts Adv. Robot. Springer, 2011, vol. 70, pp. 3–19.
- [7] J. van den Berg, M. C. Lin, and D. Manocha, "Reciprocal Velocity Obstacles for Real-Time Multi-Agent Navigation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2008, pp. 1928–1935.
- [8] S. J. Guy, J. Chhugani, C. Kim, N. Satish, M. C. Lin, D. Manocha, and P. Dubey, "Clearpath: Highly parallel collision avoidance for multi-agent simulation," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation (SCA)*, 2009, pp. 177–187.
- [9] J. van den Berg, J. Snape, S. J. Guy, and D. Manocha, "Reciprocal Collision Avoidance with Acceleration-Velocity Obstacles," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2011, pp. 3475–3482.
- [10] H. Goldstein, C. Poole, and J. Safko, *Classical Mechanics*, 3rd ed. Addison-Wesley, 2002.
- [11] F. Camara, N. Bellotto, S. Cosar, F. Weber, D. Nathanael, M. Althoff, J. Wu, J. Ruenz, A. Dietrich, G. Markkula, A. Schieben, F. Tango, N. Merat, and C. Fox, "Pedestrian models for autonomous driving part ii: High-level models of human behavior," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 9, pp. 5453–5472, Sep. 2021.
- [12] D. Hennes, D. Claes, W. Meeussen, and K. Tuyls, "Multi-robot collision avoidance with Localization Uncertainty," in *Proc. Int. Conf. Auton. Agents and Multiagent Systems*, 2012, pp. 147–154.
- [13] N. E. D. Toit and J. W. Burdick, "Robotic Motion Planning in Dynamic, Cluttered, Uncertain environments," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2010, pp. 966–973.
- [14] P. Huang, W. Liu, L. Ma, W. Liang, and Z. Liu, "Auto Regressive Moving Average (arma) modeling method for Gyro random noise using a Robust Kalman Filter," *Sensors*, 2015.
- [15] S. J. Talukder, Y. Yue, and G. Gkioxari, "Totem: Tokenized Time Series Embeddings for General Time Series Analysis," *Trans. Mach. Learn. Res.*, 2024.
- [16] V. Saxena, M. Bronars, N. R. Arachchige, K. Wang, W. C. Shin, S. Nasiriany, A. Mandekar, and D. Xu, "What matters in learning from Large-Scale datasets for Robot Manipulation?" in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2025.
- [17] H. A. Pierson and M. S. Gashler, "Deep learning in robotics: A review of recent research," *Adv. Robot.*, Jul 2017.
- [18] J. Ko and D. Fox, "GP-Bayesfilters: Bayesian Filtering using Gaussian Process Prediction and Observation Models," in *Proc. Robotics: Sci. Syst. (RSS)*, 2007.
- [19] C. D. McKinnon and A. P. Schoellig, "Context-aware cost shaping to reduce the impact of model error in safe, Receding Horizon Control," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2020, pp. 2386–2392.
- [20] B. O. Koopman, "Hamiltonian Systems and Transformation in Hilbert Space," *Proc. Natl. Acad. Sci. U.S.A.*, 1931.
- [21] P. J. Schmid and J. Sesterhenn, "Dynamic Mode Decomposition of numerical and experimental data," in *APS Fluid Dyn. Meet.*, 2008.
- [22] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A Data-Driven approximation of the Koopman Operator: Extending Dynamic Mode Decomposition," *J. Nonlinear Sci.*, 2015.
- [23] H. Arbabi and I. Mezić, "Ergodic Theory, Dynamic Mode Decomposition, and Computation of Spectral Properties of the Koopman Operator," *J. Nonlinear Sci.*, 2017.
- [24] M. Kamb, E. Kaiser, S. L. Brunton, and J. N. Kutz, "Time-Delay Observables for Koopman: Theory and applications," *SIAM J. Appl. Dyn. Syst.*, 2020.
- [25] M. Korda and I. Mezić, "Linear predictors for nonlinear dynamical systems: Koopman Operator meets model predictive control," *Automatica*, 2018.
- [26] C. Folkestad, D. Pastor, and J. W. Burdick, "Episodic Koopman learning of nonlinear robot dynamics with application to fast multirotor landing," in *IEEE Int. Conf. Robot. Autom.*, 2020, pp. 9216–9222.
- [27] H. Abdi and L. J. Williams, "Principal Component Analysis," *Wiley Interdiscip. Rev. Comput. Stat.*, 2010.
- [28] G. Berkooz, P. Holmes, and J. L. Lumley, "The Proper Orthogonal Decomposition in the analysis of turbulent flows," *Annu. Rev. Fluid Mech.*, 1993.
- [29] M. Yin and R. S. Smith, "On Low-Rank Hankel matrix denoising," *IFAC PapersOnLine*, 2021.
- [30] N. Halko, P.-G. Martinsson, and J. A. Tropp, "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions," *SIAM Rev.*, 2011.
- [31] J.-N. Juang and R. S. Pappa, "An Eigensystem Realization Algorithm for Modal Parameter Identification and Model Reduction," *J. Guid. Control Dyn.*, 1985.
- [32] N. Golyandina, V. Nekrutkin, and A. Zhigljavsky, *Analysis of Time Series Structure: SSA and Related Techniques*. Chapman & Hall, 2001.
- [33] F. Takens, "Detecting Strange Attractors in Turbulence," in *Dyn. Syst. Turbulence*, ser. Lect. Notes Math. Springer, 1981, pp. 366–381.
- [34] J. A. Cadzow, "Signal Enhancement – a composite property mapping algorithm," *IEEE Trans. Acoust., Speech, Signal Process.*, 1988.
- [35] C. Eckart and G. Young, "The Approximation of One Matrix by Another of Lower Rank," *Psychometrika*, 1936.
- [36] M. Gavish and D. L. Donoho, "The Optimal Hard Threshold for Singular Values is $4/\sqrt{3}$," *IEEE Trans. Inf. Theory*, 2014.
- [37] V. A. Marčenko and L. A. Pastur, "Distribution of Eigenvalues for Some Sets of Random Matrices," *Math. USSR-Sb.*, 1967.
- [38] L. J. Gleser, "Estimation in a Multivariate Errors in Variables Regression Model: Large Sample Results," *Ann. Stat.*, 1981.
- [39] S. Kombo, M. Haseli, S. X. Wei, and J. W. Burdick, "Real-Time Learning of Predictive Dynamic Obstacle Models for Robotic Motion Planning," 2025.
- [40] S. X. Wei, "Data-Driven Safety-Critical Autonomy in Unknown, Unstructured, and Dynamic Environments," Ph.D. dissertation, California Inst. Technol., 2024.
- [41] P. S. Maybeck, *Stochastic Models, Estimation, and Control*. New York: Academic Press, 1979.
- [42] L. Csato and Manfred Opper, "Sparse Online Gaussian Processes," *Neural Computation*, vol. 14, pp. 641–668, 2002.
- [43] S. Julier and J. Uhlmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, 2004.
- [44] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [45] M. P. Deisenroth, D. Fox, and C. E. Rasmussen, "Gaussian Processes for Data-Efficient Learning in Robotics and Control," *IEEE Trans. on Pattern Analysis and Mach. Int.*, vol. 37, no. 2, pp. 408–423, 2015.
- [46] H. Zhang, C. W. Rowley, E. A. Deem, and L. N. Cattafesta, "Online Dynamic Mode Decomposition for Time-Varying Systems," *SIAM J. Applied Dynamical Systems*, vol. 18, no. 3, pp. 1586–1609, 2019.
- [47] M. S. Hemati, M. O. Williams, and C. W. Rowley, "Dynamic Mode Decomposition for large and noisy datasets," *Phys. Fluids*, 2014.
- [48] S. T. M. Dawson, M. S. Hemati, M. O. Williams, and C. W. Rowley, "Characterizing and correcting for the effect of sensor noise in dynamic mode decomposition," *Exp. Fluids*, vol. 57, no. 3, p. 42, 2016.
- [49] M. Haseli and J. Cortés, "Approximating the Koopman Operator Using Noisy Data: Noise-Resilient Extended Dynamic Mode Decomposition," in *Amer. Control Conf.*, Philadelphia, PA, 2019, pp. 5499–5504.
- [50] L. Shi, M. Haseli, G. Mamakoukas, D. Bruder, I. Abraham, T. Murphey, J. Cortés, and K. Karydis, "Koopman Operators in Robot Learning," *IEEE Trans. on Robot.*, pp. 1–20, 2026.
- [51] E. Das, W. A. Welch, and a. Spieler, Patrick et, "Safe Payload Transfer with Ship-Mounted Cranes: A Robust Model Predictive Control Approach," *arXiv preprint arXiv:2510.16953*, 2026.
- [52] G. Klancar, A. Zdesar, S. Blazic, and I. Skrjanc, *Wheeled Mobile Robotics: From Fundamentals Towards Autonomous Systems*. Butterworth-Heinemann, 2017.
- [53] M. Aicardi, G. Casalino, A. Bicchi, and A. Balestrino, "Closed-loop steering of unicycle-like vehicles via Lyapunov techniques," *IEEE Robot. Autom. Mag.*, 1995.