Adaptive Centralized State Prediction for Multi-Agent Systems via Sliding-Window Dynamic Mode Decomposition

Alp Merzi

***Abstract*— Predicting the evolution of a Multi-Agent System (MAS) is a notoriously complex task due to the non-linear coupling and local interaction protocols governing the agents. This work explores a purely data-driven adaptive predictive framework based on Dynamic Mode Decomposition with Control (DMDc) applied at the macroscopic swarm level. Rather than modeling individual agents and their localized interaction rules in a static manner, the proposed approach aggregates the entire swarm into a single global state and continuously identifies a local linear surrogate model over a sliding historical window. By doing so, the sliding-window DMDc algorithm implicitly extracts a time-varying linear surrogate model that captures the dominant cross-couplings and emergent spatiotemporal patterns of the collective motion. This paper outlines how such a linear adaptive predictive engine can be naturally embedded within a centralized Model Predictive Control (MPC) architecture, tightly coupling the evaluation horizon to the MPC prediction horizon. Finally, experimental results involving a swarm of wheeled robots successfully validate the open-loop predictive capabilities of the identified sliding-window model, showing a drastic performance increase over isolated agent identification.**

I. Introduction

The deployment of Multi-Agent Systems (MASs) has seen exponential growth in robotics, driven by applications ranging from logistics and smart agriculture to cooperative surveillance [1]–[7]. The defining characteristic of a MAS is that complex, macroscopic collective behaviors, such as synchronization, cohesion, and obstacle avoidance, emerge from relatively simple, localized interaction rules among individual agents [8].

While these emergent properties make MASs highly robust and adaptable, they simultaneously render the mathematical prediction of the swarm's future state extremely challenging. The underlying differential equations governing the multi-agent network are inherently high-dimensional, non-linear, and heavily coupled. Consequently, designing high-level task controllers, such as Model Predictive Control (MPC) algorithms aimed at guiding the swarm along prescribed waypoints, becomes computationally prohibitive if the optimization engine must solve the full non-linear interaction model at each sampling instant [9].

The Data-Driven paradigm offers a powerful alternative to explicitly defining and solving these non-linear interaction equations [10]. Rooted in Koopman operator theory [11], algorithms such as Dynamic Mode Decomposition (DMD) [12], [13] allow for the extraction of linear, reduced-order dynamic models directly from time-series measurements of the system.

The core contribution of this paper is the conceptualization and experimental validation of an adaptive macroscopic, swarm-level DMD predictive framework based on a sliding window. We argue that applying DMD to the aggregated global state of the MAS is vastly superior to applying it to individual agents. While single-agent dynamics appear highly complex due to unmodeled external influences from neighbors, the swarm as a cohesive whole evolves on a low-dimensional manifold dictated by the emergent behavior. By operating on the aggregated state over a rolling window, DMD seamlessly captures these implicit time-varying cross-couplings. The paper outlines the formulation of an MPC strategy built upon this surrogate model, rigorously examining both the computational strengths of the resulting linear formulation and the architectural weaknesses tied to its centralized nature. Finally, an experimental campaign is conducted strictly to validate the predictive accuracy of the data-driven model under the new multi-metric configuration.

II. The Advantage of Macroscopic Swarm Identification

Consider a multi-agent system comprising N agents. Let $z_i(t) \in \mathbb{R}^n$ represent the measured spatial coordinates of the i-th agent, and $u_i(t) \in \mathbb{R}^m$ its control input. In a standard coordinated swarm, the time evolution of any single agent is a non-linear function of its own state, its input, and the states of its neighboring agents within a sensing radius:

$$\dot{z}_i(t) = f_i(z_i(t), z_j(t), u_i(t)), \quad \forall j \in \mathcal{N}_i \tag{1}$$

where $\mathcal{N}_i$ denotes the neighbor set of agent i, and $f_i(\cdot)$ encapsulates the unknown or highly complex interaction logic (e.g., potential fields for collision avoidance and cohesion).

A. Individual vs. Collective Identification

A naive data-driven approach might attempt to identify a predictive model for each agent independently. Applying DMD to the isolated state history of agent i seeks to find a local matrix $\mathbf{A}_i$ such that $z_{i,k+1} \approx \mathbf{A}_i z_{i,k} + \mathbf{B}_i u_{i,k}$. However, this approach is fundamentally flawed in a MAS context. Because agent i's motion is continuously perturbed by the state of agent j, the isolated trajectory appears quasi-stochastic or highly non-linear, leading to extremely poor linear fits and catastrophic prediction divergence.

A. Merzi is with the University of Calabria, Dept. of Informatics, Modeling, Electronics and Systems Eng., 87036 Rende, Italy (e-mail: alp.merzi@dimes.unical.it). ORCID: 0000-0003-4282-2486.

This study was carried out within the Space It Up project funded by the Italian Space Agency (ASI) and the Ministry of University and Research (MUR), under contract no. 2024-5-E.0 – CUP no. I53D24000060005.

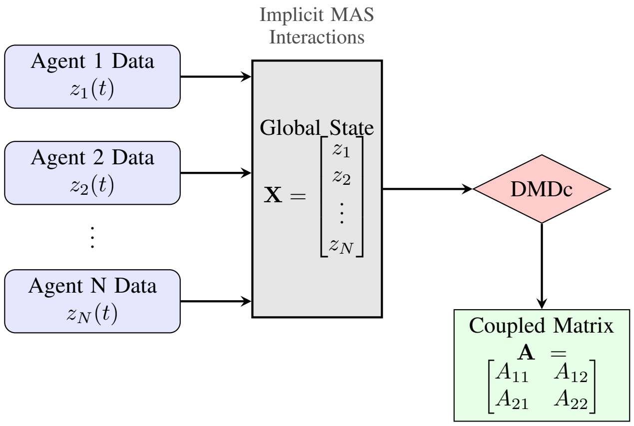

Fig. 1. The DMD+MAS paradigm: Stacking agent states into a global vector allows the DMDc algorithm to explicitly quantify emergent cross-couplings (captured in the off-diagonal blocks A_{ij}) that would be invisible if agents were modeled independently.

Instead, the proposed methodology aggregates the entire swarm into a single global state vector $\mathbf{x}_k \in \mathbb{R}^{nN}$:

$$\mathbf{x}_k := \begin{bmatrix} z_1(kT_s) \\ z_2(kT_s) \\ \vdots \\ z_N(kT_s) \end{bmatrix}, \quad \mathbf{u}_k := \begin{bmatrix} u_1(kT_s) \\ u_2(kT_s) \\ \vdots \\ u_N(kT_s) \end{bmatrix} \tag{2}$$

By collecting data at the macroscopic level, we exploit the fundamental nature of swarm robotics: *emergent order*. However, forming the global snapshot accurately relies on the assumption that all local state updates are acquired synchronously. In distributed hardware, preserving synchronization accuracy and managing trigger realignments is a fundamental prerequisite [14], [15] before data aggregation can occur. While individual interactions $f_i(\cdot)$ are non-linear, the collective goal of the protocol (e.g., maintaining a formation or synchronizing velocities) forces the overall system $\mathbf{x}_k$ to collapse onto a highly structured, low-dimensional attractor. Applying DMD to the global state $\mathbf{x}_k$ yields a global operator $\mathbf{A} \in \mathbb{R}^{nN\times nN}$. The off-diagonal block matrices within $\mathbf{A}$ naturally capture the cross-agent dependencies without requiring any explicit knowledge of the topology $\mathcal{N}_i$ or the non-linear interaction rules. The conceptual advantage of this approach is illustrated in Fig. 1.

III. Adaptive Sliding-Window Dynamic Mode Decomposition with Control

To capture the time-varying nature of the non-linear coupling and localized agent interactions during complex maneuvers, we introduce an adaptive sliding-window formulation of Dynamic Mode Decomposition with Control (DMDc) [16]. Rather than identifying a static global model over the entire operational history, a local linear surrogate is continuously updated using a rolling window of recent historical data. As illustrated in Fig. 2, at each discrete time step k, we define a historical training window of length N_w. The corresponding sequential snapshots of the

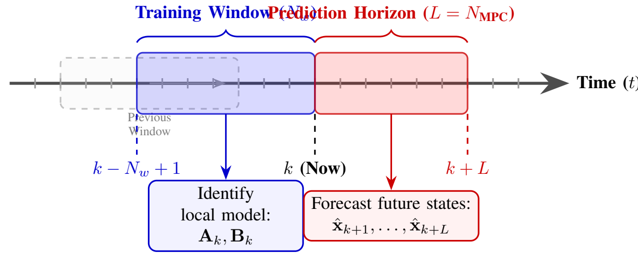

Fig. 2. Adaptive Sliding-Window DMDc paradigm: at each sampling instant k, a localized linear model is identified over a rolling window N_w and used to predict the future states over the validation horizon L.

aggregated swarm states and control inputs are organized into the following local data matrices:

$$\begin{cases} \mathbf{X}_k = [\mathbf{x}_{k-N_w+1}, \mathbf{x}_{k-N_w+2}, \dots, \mathbf{x}_{k-1}] \\ \mathbf{U}_k = [\mathbf{u}_{k-N_w+1}, \mathbf{u}_{k-N_w+2}, \dots, \mathbf{u}_{k-1}] \\ \mathbf{X}'_k = [\mathbf{x}_{k-N_w+2}, \mathbf{x}_{k-N_w+3}, \dots, \mathbf{x}_k] \end{cases} \tag{3}$$

The objective at time step k is to identify the instantaneous best-fit linear operators $\mathbf{A}_k \in \mathbb{R}^{nN\times nN}$ and $\mathbf{B}_k \in \mathbb{R}^{nN\times mN}$ that minimize the multi-step residual error, satisfying:

$$\mathbf{X}'_k \approx \mathbf{A}_k\mathbf{X}_k + \mathbf{B}_k\mathbf{U}_k = \begin{bmatrix} \mathbf{A}_k & \mathbf{B}_k \end{bmatrix} \boldsymbol{\Omega}_k \tag{4}$$

where $\boldsymbol{\Omega}_k := \begin{bmatrix} \mathbf{X}_k \\ \mathbf{U}_k \end{bmatrix}$.

The local identification process leverages Singular Value Decomposition (SVD) truncated to a chosen rank r to project the snapshot data onto a low-dimensional dominant manifold, filtering out high-frequency measurement noise. The computation steps executed at each time step k are as follows:

1) Compute the SVD of the augmented matrix $\boldsymbol{\Omega}_k \approx \tilde{\mathbf{U}}\tilde{\boldsymbol{\Sigma}}\tilde{\mathbf{V}}^*$, truncating the expansion to rank r. The left singular vector matrix $\tilde{\mathbf{U}} \in \mathbb{R}^{(nN+mN)\times r}$ is conformably partitioned as:

$$\tilde{\mathbf{U}} = \begin{bmatrix} \tilde{\mathbf{U}}_1 \\ \tilde{\mathbf{U}}_2 \end{bmatrix} \tag{5}$$

where $\tilde{\mathbf{U}}_1 \in \mathbb{R}^{nN\times r}$ and $\tilde{\mathbf{U}}_2 \in \mathbb{R}^{mN\times r}$ match the state and input dimensions, respectively.

2) Directly extract the high-dimensional, adaptive system operators utilizing the pseudo-inverse of the singular values:

$$\mathbf{A}_k = \mathbf{X}'_k\tilde{\mathbf{V}}\tilde{\boldsymbol{\Sigma}}^{-1}\tilde{\mathbf{U}}_1^*, \quad \mathbf{B}_k = \mathbf{X}'_k\tilde{\mathbf{V}}\tilde{\boldsymbol{\Sigma}}^{-1}\tilde{\mathbf{U}}_2^* \tag{6}$$

This yields an active, data-driven linear surrogate capable of forecasting the state trajectory into a future horizon based on the immediate past correlations of the collective motion.

IV. MPC Application: Strengths and Weaknesses

The primary motivation for identifying a linear, discrete-time model of a MAS is to employ it within an optimal control framework. By substituting the non-linear dynamics (1) with the online adaptive DMDc surrogate, a centralized MPC outer-loop can be formulated. At each time step k, the MPC minimizes a quadratic cost function over a finite optimization horizon N_{MPC}:

$$\min_{\mathbf{U}_k} \sum_{l=1}^{N_{\text{MPC}}} \left(\|\hat{\mathbf{x}}_{k+l} - \mathbf{x}_{\text{ref}}\|_{\mathbf{Q}}^2 + \|\mathbf{u}_{k+l-1}\|_{\mathbf{R}}^2 \right) \tag{7}$$
$$\text{subject to} \quad \hat{\mathbf{x}}_{k+l} = \mathbf{A}_k \hat{\mathbf{x}}_{k+l-1} + \mathbf{B}_k \mathbf{u}_{k+l-1},$$

where $\mathbf{x}_{\text{ref}}$ is the desired waypoint configuration, $\mathbf{Q}, \mathbf{R}$ are standard weight matrices, and $\hat{\mathbf{x}}$ denotes the predicted state trajectory. Crucially, the prediction horizon L utilized to assess the open-loop model validation accuracy in the subsequent experimental section is selected to strictly mirror this control horizon ($L \equiv N_{\text{MPC}}$), guaranteeing that model performance is evaluated precisely over the operational look-ahead window of the optimizer.

A. *Methodological Strengths*

The primary strength of this DMD+MPC approach is its **mathematical simplicity and computational speed**. Solving a standard MPC with non-linear constraints requires Non-Linear Programming (NLP) solvers, which suffer from local minima and high convergence times, making real-time MAS control nearly impossible. By utilizing the DMD surrogate, the MPC constraint becomes strictly linear, reducing the optimization to a highly convex Quadratic Programming (QP) problem. The optimization solver is agnostic to the complex collision-avoidance or cohesion algorithms running locally on the agents; it simply observes and exploits the resulting linear correlations.

B. *Methodological Weaknesses*

Conversely, the reliance on a global state vector $\mathbf{x}_k$ introduces significant architectural limitations. The DMD+MAS paradigm is inherently **centralized**. It requires a central computation hub capable of receiving the state of every agent, computing the global optimal input $\mathbf{U}_k$, and transmitting the commands back to the swarm. 1. *Scalability and Communication:* As N grows, the communication overhead scales drastically. Furthermore, transmitting individual states to a centralized hub introduces critical communication delays, requiring robust clock synchronization protocols across the multi-agent network (see, e.g., [**?**]) to ensure the aggregated global state is evaluated correctly. 2. *Topological Fragility:* The matrix $\mathbf{A}$ implicitly encodes the spatial relationship of the agents. If the swarm splits due to an obstacle, or if an agent suffers a hardware failure, the previously identified correlations in $\mathbf{A}$ become invalid, requiring immediate re-identification. Distributing the DMDc algorithm to allow agents to compute local predictive models using only neighboring data remains a complex open challenge in operator theory.

V. Experimental Validation of Predictions

It is crucial to clarify that the current experimental phase focuses exclusively on validating the *open-loop predictive capabilities* of the identified DMDc surrogate. The closed-loop execution of the MPC detailed in Section IV is deferred to future work. The objective here is to prove that the centralized DMD model can accurately forecast the non-linear MAS trajectory over a finite horizon.

Three Elisa-3 autonomous wheeled robots [18] were deployed in a controlled laboratory arena. Position tracking is managed centrally via an overhead RGB camera and a video processing module. During the experiment, the 3 agents evolved according to the multi-agent synchronization law defined in [8]. The collected dataset consists of 402 discrete samples. For the sliding-window DMDc implementation, the training window was set to $N_w = 100$ samples. The SVD truncation rank was chosen as $r = 8$ for the centralized global model (which processes the full 9 state variables collectively), and $r = 3$ for the individual decentralized models (which identify the 3 state variables of each agent independently). The complete experimental setup is illustrated in Fig. 3.

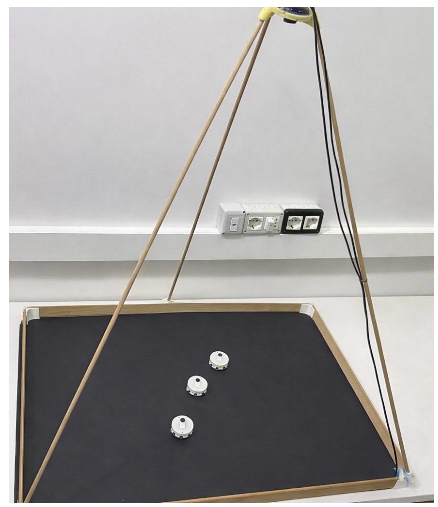

Fig. 3. Laboratory environment.

To rigorously assess the model's accuracy during the rolling testing phase without masking local variations, the prediction evaluation is decoupled between the translational spatial coordinates ($\mathbf{p}$) and the rotational orientations ($\boldsymbol{\theta}$). Let $\mathcal{I}_{\text{pos}}$ and $\mathcal{I}_\theta$ denote the sets of indices corresponding to the position states (x, y) and heading angles (θ) within the global state vector, respectively.

Instead of taking the norm of the entire trajectory matrix globally, the error is computed column-by-column (i.e., at each individual future step within the horizon) to evaluate the instantaneous relative error, and then averaged over the prediction horizon L. Furthermore, the orientational metric explicitly incorporates a wrapping operation to correctly handle the 2π angular periodicity and prevent artificial error inflation near the phase boundaries. The normalized percentage prediction errors at step k are defined as:

$$E_{\text{pos}}(k) = \frac{1}{L} \sum_{j=1}^{L} \frac{\|\mathbf{p}_{k+j} - \hat{\mathbf{p}}_{k+j,k}\|_2}{\|\mathbf{p}_{k+j}\|_2 + \epsilon} \times 100\% \tag{8}$$

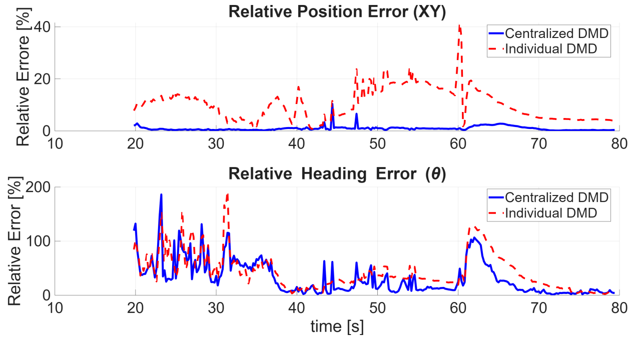

Fig. 4. Comparison of the Relative Position Error (top) and Relative Heading Error (bottom) over time. The centralized sliding-window DMDc (solid blue) effectively prevents cumulative drift, severely outperforming the individual DMDc (dashed red).

$$E_\theta(k) = \frac{1}{L}\sum_{j=1}^{L} \frac{\|\,\mathrm{mod}(\boldsymbol{\theta}_{k+j} - \hat{\boldsymbol{\theta}}_{k+j,k} + \pi, 2\pi) - \pi\|_2}{\|\boldsymbol{\theta}_{k+j}\|_2 + \epsilon} \times 100\% \tag{9}$$

where $\mathbf{p}$ and $\boldsymbol{\theta}$ represent the true states extracted via $\mathcal{I}_{\text{pos}}$ and $\mathcal{I}_\theta$, $\hat{\mathbf{p}}_{k+j,k}$ and $\hat{\boldsymbol{\theta}}_{k+j,k}$ represent the j-step ahead open-loop forecasts generated by recursively applying the local operators $(\mathbf{A}_k, \mathbf{B}_k)$ starting from the real state at time k, and ϵ is a small regularization constant added to prevent numerical instability during zero-crossings.

Centralized vs. Individual Performance

The comparative results of the predictive performance are shown in Fig. 4. As hypothesized, aggregating the swarm into a single state vector allows the DMDc to accurately track the collective motion, whereas the individual model suffers significantly. The top plot of Fig. 4 demonstrates a drastic reduction in the Relative Position Error (E_{pos}) for the centralized model, which remains consistently flat and near zero (around $1-2\%$). In stark contrast, when the DMDc algorithm is applied individually (agent-by-agent), the model fails to account for unmodeled neighboring perturbations. This results in significant cumulative drift, with the individual positional error heavily fluctuating and exhibiting severe spikes approaching 50%. The bottom plot illustrates the Relative Heading Error (E_θ). It is important to note that relative angular metrics inherently exhibit transient spikes—a known numerical artifact when calculating relative errors for near-zero angular values during fast maneuvers or phase wraps. Despite this noise, the centralized approach clearly maintains a lower overall error profile compared to the individual baseline. These results confirm that the coupled off-diagonal elements of the identified global matrix $\mathbf{A}_k$ successfully approximate the non-linear interaction forces keeping the swarm cohesive, preventing the catastrophic divergence seen in the decentralized approach.

VI. Concluding Remarks

This work demonstrates that applying Dynamic Mode Decomposition at the macroscopic level of a Multi-Agent System provides a highly accurate, data-driven predictive engine. By treating the swarm as a single dynamical entity, the DMD algorithm circumvents the need to explicitly model non-linear local interactions, instead capturing the emergent low-dimensional behaviors directly from position data. While this approach perfectly facilitates a computationally lightweight, centralized MPC formulation, its reliance on a global state vector renders it vulnerable to communication bottlenecks and topological changes. Future research must focus on decentralizing the DMDc identification process to preserve the scalability inherent to swarm robotics.

References

[1] L. D'Alfonso, F. Giannini, G. Franzé, G. Fedele, F. Pupo, and G. Fortino, "Autonomous vehicle platoons in urban road networks: A joint distributed reinforcement learning and model predictive control approach," in *IEEE/CAA journal of automatica sinica*, Vol. 11, No. 1, Pp. 141-156, 2024.

[2] A. Vitale and F. Lamonaca, "Enhancing GeoAI land cover classification via hyperparameter tuning and cross-validation: A case study in Ravenna, Italy," *Measurement: Journal of the International Measurement Confederation*, vol. 257, 2026, doi: 10.1016/j.measurement.2025.118662.

[3] R. F. Carpio, L. Di Giulio, E. Garone, G. Ulivi, and A. Gasparri, "A distributed swarm aggregation algorithm for bar shaped multi-agent systems," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Pp. 4303-4308, 2018.

[4] G. Fedele and L. D'Alfonso, "A model for swarm formation with reference tracking," in *Proceedings of the 56th IEEE Conference on Decision and Control (CDC)*, Melbourne, Australia, 2017, pp. 381–386.

[5] G. Fedele, L. D'Alfonso, A. Bono, and V. Gazi, "Swarm trajectories generation for target capturing with uncertain information," *IEEE Transactions on Control of Network Systems*, vol. 10, no. 4, pp. 1986–1996, 2023.

[6] M. A. Merzi, V. Gazi, G. Fedele, L. D'Alfonso, and A. Bono, "Target capturing in an ellipsoidal region with a swarm of quadrotor UAVs," in *Proceedings of the 2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE, 2022, pp. 972–981.

[7] G. Fedele, L. D'Alfonso, and A. Bono, "Vortex formation in a swarm of agents with a coordinates mixing matrix-based model," *IEEE Control Systems Letters*, vol. 4, no. 2, pp. 420–425, 2019.

[8] L. D'Alfonso and G. Fedele, "An Emergent Synchronization Property in a Set of Non-Competing Multi-Agent Systems," *IEEE Control Systems Letters*, vol. 8, pp. 598–603, 2024.

[9] F. Borrelli, A. Bemporad, and M. Morari, "Predictive control for linear and hybrid systems," *Cambridge University Press*, 2017.

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

[11] 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.

[12] P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," *J. Fluid Mech.*, vol. 656, pp. 5–28, 2010.

[13] S. L. Brunton, B. R. Noack, and P. Koumoutsakos, "Machine Learning for Fluid Mechanics," *Annu. Rev. Fluid Mech.*, vol. 52, pp. 477–508, 2020.

[14] F. Lamonaca, D. L. Carni, M. Riccio, D. Grimaldi, and G. Andria, "Preserving Synchronization Accuracy from the Plug-in of NonSynchronized Nodes in a Wireless Sensor Network," *IEEE Transactions on Instrumentation and Measurement*, vol. 66, no. 5, pp. 1058–1066, 2017, 10.1109/TIM.2017.2664422.

[15] F. Lamonaca and D. Grimaldi, "Trigger realignment by networking synchronized embedded hardware," *IEEE Transactions on Instrumentation and Measurement*, vol. 62, no. 1, pp. 38–49, 2013, doi: 10.1109/TIM.2012.2218058.

[16] 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.

[17] E. Garone *et al.*, "Clock synchronization for wireless sensor network with communication delay," *Proceedings of the American Control Conference*, pp. 771–776, 2013, doi: 10.1109/ACC.2013.6579929.

[18] GCTronic, "Elisa-3 Documentation," *Online*, `http://www.gctronic.com/doc/index.php/Elisa-3`.