Real-Time Estimation of Drag-Based Parameters in LEO via Adaptive Control and Sparse Identification

Morokot Sakal* and Camilo Riano-Rios.† and Madhur Tiwari‡
Florida Institute of Technology, Melbourne, Florida, 32901, United States

Estimating drag-related parameters in Low Earth Orbit (LEO) remains a challenge due to uncertainties in atmospheric models and spacecraft physical properties. This paper takes inspiration from the Sparse Identification of Nonlinear Dynamics (SINDy) technique from Machine Learning (ML) to propose a Concurrent Learning-based adaptation law, modified with a Bregman divergence-derived term, that learns the experienced differential drag when modeled as a large collection of nonlinear functions with unknown coefficients, while ensuring tracking of desired relative orbital states between a chaser and a target spacecraft. The stability of the controller is assessed via Lyapunov-based stability analysis and performance evaluated through numerical simulations. The numerical results show that the proposed controller can simultaneously ensure relative states tracking and identify a sparse representation of the differential drag acceleration in the LVLH frame. This work combines traditional adaptive control with an ML-based strategy to enable real-time identification with convergence guarantees, making the framework well-suited for safety-critical aerospace applications where large datasets are often unavailable.

I. Introduction

As the number of space missions involving multiple satellites increases, so does the need to make spacecraft more autonomous. In Low Earth Orbit (LEO), the most populated orbital regime with an increasing number of valuable assets, a spacecraft's ability to adapt and learn from its environment becomes critically important.

A promising technique for autonomous and cost-effective maneuvering in LEO is Differential Drag (DD), which exploits differences in atmospheric drag experienced by spacecraft to generate controlled relative motion. DD-based maneuvering has evolved from linearized, open-loop guidance strategies [1, 2], to adaptive [3] and optimal [4] control methods, with demonstrated success in on-orbit station-keeping operations by commercial satellite constellations as a propellant saving technique [5, 6]. Similar efforts have been made to adopt a higher fidelity relative-motion models and exploit additional aerodynamic effects (e.g., lift) to increase control authority [7–11]. However, most DD control strategies rely on accurate knowledge (or reliable online estimates) of drag-related parameters to allocate the differential drag required by control laws on the involved spacecraft, which can include the effect of atmospheric density and ballistic/drag coefficients, both highly uncertain quantities.

Estimation of drag-related parameters is difficult because atmospheric density is highly uncertain. Several semi-empirical atmospheric models, such as Harris-Priester [12] and NRLMSISE-00 [13], and neural network-based [14] exist but still show prediction error due to changing space-weather conditions. Similarly, drag/ballistic coefficients depend on spacecraft geometry, attitude, and surface properties, and are often only approximately known [15]. These issues motivate online learning strategies that can improve drag-acceleration reconstruction directly from onboard measurements, without relying on large datasets or computationally expensive identification routines or models.

Lyapunov-based adaptive control techniques have been explored to compensate for uncertainties in these drag-related parameters [3]. In prior work, we introduced an adaptive control strategy to simultaneously perform DD-based maneuvers and estimate key drag-related parameters under formal convergence guarantees [16–18]. The control strategy incorporated the Integral Concurrent Learning (ICL) adaptation technique [19], which allows online collection of input-output data and ensures convergence of estimated model parameters to their true values, provided a verifiable Finite Excitation (FE) condition is satisfied. While the results in [16] were promising, they required expressing the spacecraft relative dynamics and the atmospheric density model in simplified forms to enable the use of ICL, ultimately limiting the applicability of the estimation in more or complex scenarios.

*Ph.D. Student, Aerospace, Physics and Space Sciences Department, 150 W. University Blvd., Melbourne, FL, 32901.
†Assistant Professor, Aerospace, Physics and Space Sciences Department, 150 W. University Blvd., Melbourne, FL, 32901.
‡Assistant Professor, Aerospace, Physics and Space Sciences Department, 150 W. University Blvd., Melbourne, FL, 32901.

In this paper we consider a two-spacecraft scenario where a chaser maneuvers using actuation mechanisms other than DD. The focus is not on executing DD-based maneuvers, but on developing a more flexible and general framework for learning portions of the spacecraft dynamics, e.g., the experienced differential drag, using only measurable state data and a large library of nonlinear functions. The application of the learning strategy to this scenario has the potential to enable its use for control allocation of DD-based control laws, and illustrates its applicability on a wide range of space applications where dynamics are not accurately known.

We take inspiration from the ML technique known as Sparse Identification of Nonlinear Dynamics (SINDy) [20], which formulates a sparse regression problem using batches of data to learn all or part of the uncertain Equations of Motion (EoMs) describing a dynamic system. The data consists of system state measurements, their time-derivatives, and applied control inputs[21]. SINDy leverages the fact that most physically realizable systems have dynamics governed by a few dominant terms, making their EoMs sparse in a high-dimensional nonlinear function space. This method has been successfully used to discover EoMs of multiple systems from data [20], and in control contexts when combined with Model Predictive Control (MPC) [22, 23]. However, it relies on collecting large amounts of data and executing iterative algorithms to enforce sparsity, features that limit its applicability for onboard spacecraft operations.

To address these limitations, we recognize its similarities with Concurrent Learning (CL) in terms of the input-output data required for estimation and the formulation of the underlying regression problem. We propose a modified CL-inspired online adaptation law with sparsity enforcement. This approach enables us to express the uncertain terms in the EoMs as a linear parameterization over an arbitrarily large library of nonlinear functions, each associated with an unknown coefficient. We then enforce sparsity in the coefficients matrix, i.e., the EoMs are represented using a relatively small number of active terms.

To ensure state tracking and sparse identification, we explore the use of a Lyapunov-Bregman function as part of the adaptive controller design. This function incorporates a Bregman divergence term [24], a non-Euclidean "distance" measure designed in this case to encourage sparsity in the learned coefficients. This approach is motivated by the growing interest in connecting adaptive control with ML concepts [25–27], and recent developments based on incorporating Bregman divergence into candidate Lyapunov functions [28–30].

This paper is structured as follows: Section II introduces the relative dynamics and differential drag modeling, Section III presents the control design, Section IV provides the proof of stability analysis, Section V and VI explain the simulation setup and the numerical simulation results, and Section VII concludes the paper and outline future work.

II. Dynamics Modeling

Consider a pair of satellites, a target t and a chaser c in a circular Low Earth Orbit (LEO), as illustrated in Fig. 1. The target is assumed to follow a circular orbit, while the chaser performs relative maneuvers using thrusters, and both spacecraft are subjected to unknown time-varying differential drag. This section describes the relative dynamics and the differential drag model in a manner that enables the application of SINDy-inspired parameterization and estimation using an adaptive concurrent learning law. Note that the relative dynamics can be modeled using Clohessy-Wiltshire [31], Schweighart-Sedwick equations [7], or any other valid representation, as the specific choice does not affect the development of the estimation strategy.

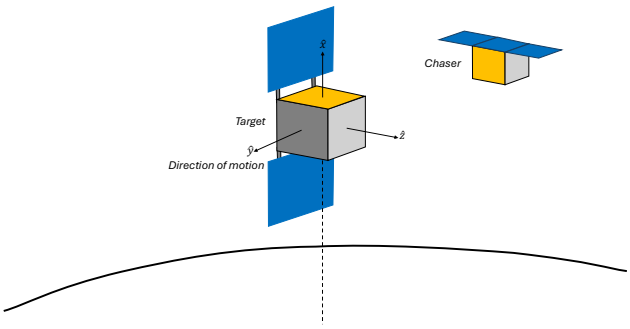


Fig. 1 Chaser-target satellites in the LVLH frame

.

A. Spacecraft Relative Dynamics

Let $\mathcal{I}$ denote the Earth-Centered Initial (ECI) frame with origin locates at the Earth's center of mass. $\hat{\mathbf{i}}_z$ points to the North Celestial Pole, $\hat{\mathbf{i}}_x$ points toward the vernal equinox, and $\hat{\mathbf{i}}_y$ completes the right-handed triad [32]. Let $\mathcal{L}$ denote the Local–Vertical/Local–Horizontal (LVLH) frame attached to the target. The origin of the LVLH frame is at the center of the target where $\hat{\mathbf{x}}$ axis is pointing from the Earth toward the target, $\hat{\mathbf{z}}$ axis is pointing to the orbital angular momentum, and $\hat{\mathbf{y}}$ axis complete the right-handed coordinate system[7, 31].

The chaser-to-target relative position expressed in $\mathcal{L}$ is defined as

$$\mathbf{x} \triangleq \begin{bmatrix} x & y & z \end{bmatrix}^\top \in \mathbb{R}^3, \tag{1}$$

In this paper, the relative dynamics is modeled using the Clohessy–Wiltshire (CW) equations for the circular orbit with mean motion n[31]. The equation of motion based on the CW dynamics can be written in the compact form as

$$\ddot{\mathbf{x}} = C\,\dot{\mathbf{x}} + G\,\mathbf{x} + \mathbf{u} + \Delta\mathbf{a}_D^L, \tag{2}$$

where $\dot{\mathbf{x}}$ is the relative velocity and $\ddot{\mathbf{x}}$ is relative acceleration, $\Delta\mathbf{a}_D^L \in \mathbb{R}^3$ is the differential acceleration expressed in the LVLH frame, and

$$C = \begin{bmatrix} 0 & 2n & 0 \\ -2n & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad G = \operatorname{diag}\big(3n^2,\, 0,\, -n^2\big). \tag{3}$$

The unknown disturbance is the differential drag acceleration, which we model in the next subsection.

B. Atmospheric Drag Model

For each spacecraft $i \in \{c, t\}$, its position and velocity in the ECI frame $\mathcal{I}$ is $\mathbf{r}_i^I$, $\mathbf{v}_i^I$ respectively. We express the velocity of the spacecraft i relative to the Earth atmosphere as

$$\mathbf{v}_{\mathrm{rel},i}^I = \mathbf{v}_i^I - \boldsymbol{\omega}_E \times \mathbf{r}_i^I, \tag{4}$$

where $\boldsymbol{\omega}_E$ denotes the Earth rotation vector. We define,

$$v_{\mathrm{rel},i} \triangleq \big\|\mathbf{v}_{\mathrm{rel},i}^I\big\|, \qquad \hat{\mathbf{v}}_{\mathrm{rel},i}^I \triangleq \frac{\mathbf{v}_{\mathrm{rel},i}^I}{v_{\mathrm{rel},i}}, \tag{5}$$

as the relative speed and its unit direction, respectively.

The drag acceleration of the spacecraft i in the ECI is modeled as

$$\mathbf{a}_{D,i}^I(t) = -\,B_i\,\rho_{\mathrm{atm},i}(t)\,v_{\mathrm{rel},i}^2\,\hat{\mathbf{v}}_{\mathrm{rel},i}^I, \tag{6}$$

where

$$B_i \triangleq \frac{1}{2}\,\frac{C_{D,i}A_i}{m_i} \quad [\mathrm{m}^2/\mathrm{kg}] \tag{7}$$

is the ballistic coefficient based on the drag coefficient, effective drag area A_i, and mass m_i of the spacecraft i.

To model the atmosphere density, we adopted the simplified density model used in [10] and [16]. For each spacecraft $i \in \{c, t\}$, the atmosphere density is defined as

$$\rho_{\mathrm{atm},i}(t) = D_{i,1} + D_{i,2}\sin(nt) + D_{i,3}\cos(nt) = s^\top(t)\,D_i, \tag{8}$$

where

$$s(t) \triangleq \begin{bmatrix} 1 & \sin(nt) & \cos(nt) \end{bmatrix}^\top, \qquad D_i \triangleq \begin{bmatrix} D_{i,1} & D_{i,2} & D_{i,3} \end{bmatrix}^\top \in \mathbb{R}^3 \tag{9}$$

where n is the mean motion. By substituting the (8) into (6), we obtain

$$\mathbf{a}_{D,i}^I(t) = \Big(-\,B_i\,v_{\mathrm{rel},i}^2\,\hat{\mathbf{v}}_{\mathrm{rel},i}^I\,s^\top(t)\Big)D_i, \tag{10}$$

C. SINDy-Inspired Representation of Differential Drag

Applied to this problem, the standard (offline) SINDy algorithm would collect $m \in \mathbb{Z}_{>0}$ samples of $\mathbf{x}$, $\mathbf{x}$, and $\mathbf{x}$, and the collected data matrix is formed as

$$X = \begin{bmatrix} \boldsymbol{x}(t_1) & \boldsymbol{x}(t_2) & \cdots & \boldsymbol{x}(t_m) \end{bmatrix}^T \in \mathbb{R}^{m\times 3}, \tag{11}$$

with $\dot{\boldsymbol{X}}$ and $\ddot{\boldsymbol{X}}$ containing the corresponding time derivatives. A library $\Gamma(X,\ \dot{X}) \in \mathbb{R}^{m\times p}$ of $p \in \mathbb{Z}_{>0}$ candidate nonlinear functions of X, is constructed as

$$\Gamma(\boldsymbol{X},\ \dot{\boldsymbol{X}}) = \begin{bmatrix} | & | & | & & | & | \\ 1 & f_1(\boldsymbol{x},\dot{\boldsymbol{x}},t_i) & f_2(\boldsymbol{x},\dot{\boldsymbol{x}},t_i) & \cdots & f_{p-2}(\boldsymbol{x},\dot{\boldsymbol{x}},t_i) & f_{p-1}(\boldsymbol{x},\dot{\boldsymbol{x}},t_i) \\ | & | & | & & | & | \end{bmatrix}, \text{ for } i = 1, \cdots, m \tag{12}$$

where $f_k(\cdot) : \mathbb{R}^6 \to \mathbb{R}$ denotes an user-defined nonlinear function in $(\cdot)$, and $k = 1,\ 2,\ \cdots,\ p-1$. The problem is then presented as a sparse regression to find the matrix $\Xi = [\boldsymbol{\xi_1}, \boldsymbol{\xi_2}, \boldsymbol{\xi_3}] \in \mathbb{R}^{p\times 3}$ in

$$\ddot{\boldsymbol{X}} = F(\boldsymbol{X},\ \dot{\boldsymbol{X}}) + \Gamma(\boldsymbol{X},\ \dot{\boldsymbol{X}})\Xi + U \tag{13}$$

where $\boldsymbol{\xi_j} \in \mathbb{R}^p$ is the sparse vector of coefficients determining which nonlinearities are active for the j^{th} state dynamics, and $F(X,\ \dot{X}) \in \mathbb{R}^{m\times 3}$, and $U \in \mathbb{R}^{m\times 3}$ are matrices with data samples collected from the known portion of the system's dynamics and the control inputs , respectively. SINDy then requires m data samples of the states, their time derivatives, known dynamics and control inputs to run a sparse regression algorithm (e.g., Lasso regression [33]).

Motivated by this structure, we express the unknown differential drag term using the SINDy-inspired representation over a rich library of functions as

$$\Delta\mathbf{a}_D^L(\mathbf{x},\dot{\mathbf{x}},t) = Y(\mathbf{x},\dot{\mathbf{x}},t)\,\boldsymbol{\theta}, \tag{14}$$

where $Y(\mathbf{x},\dot{\mathbf{x}},t) \in \mathbb{R}^{3\times 3p}$ is a regressor built from a library $\Gamma(\mathbf{x},\dot{\mathbf{x}},t) \in \mathbb{R}^{1\times p}$ and $\boldsymbol{\theta} \in \mathbb{R}^{3p}$ contains the corresponding (unknown) coefficients. In this work, the library $\Gamma(\mathbf{x},\dot{\mathbf{x}},t)$ is built using the LVLH components of the velocities relative to the atmosphere of both spacecraft, and the atmospheric density model.

To build the library, first, we introduce

$$\mathbf{v}_{\text{rel},i}^L = R_{I2L}(t)\,\mathbf{v}_{\text{rel},i}^I = \begin{bmatrix} v_{\text{rel},ix} & v_{\text{rel},iy} & v_{\text{rel},iz} \end{bmatrix}^\top, \qquad i \in \{c,t\}, \tag{15}$$

as the components of the relative velocity expressed in LVLH. We define the following term that depends on the mean motion n of the target as

$$S(t) \triangleq \sin(nt), \qquad C(t) \triangleq \cos(nt). \tag{16}$$

We build the library for the chaser as

$$\begin{aligned}\Phi_c(t) = \big[&v_{\text{rel},cx},\ v_{\text{rel},cx}S,\ v_{\text{rel},cx}C,\ v_{\text{rel},cy},\ v_{\text{rel},cy}S,\ v_{\text{rel},cy}C,\\ &v_{\text{rel},cz},\ v_{\text{rel},cz}S,\ v_{\text{rel},cz}C\big] \in \mathbb{R}^{1\times 9},\end{aligned} \tag{17}$$

and similarly, the library for the target as

$$\begin{aligned}\Phi_t(t) = \big[&v_{\text{rel},tx},\ v_{\text{rel},tx}S,\ v_{\text{rel},tx}C,\ v_{\text{rel},ty},\ v_{\text{rel},ty}S,\ v_{\text{rel},ty}C,\\ &v_{\text{rel},tz},\ v_{\text{rel},tz}S,\ v_{\text{rel},tz}C\big] \in \mathbb{R}^{1\times 9}.\end{aligned} \tag{18}$$

We then add the scalar factors, including the ballistic coefficients and squared velocities into the library

$$K_c(t) = -B_c\,v_{\text{rel},c}^2, \qquad K_t(t) = +B_t\,v_{\text{rel},t}^2, \tag{19}$$

and by stacking all the library of functions of the chaser and target, we obtain

$$\Gamma(\mathbf{x},\dot{\mathbf{x}},t) = \begin{bmatrix} K_c(t)\,\Phi_c(t) & K_t(t)\,\Phi_t(t) \end{bmatrix} \in \mathbb{R}^{1\times 18}. \tag{20}$$

The regressor $Y(\mathbf{x},\dot{\mathbf{x}},t) \in \mathbb{R}^{3\times 54}$ can be built using the Kronecker product

$$Y(\mathbf{x}, \dot{\mathbf{x}}, t) = \Gamma(\mathbf{x}, \dot{\mathbf{x}}, t) \otimes I_3, \tag{21}$$

and the differential drag can be expressed using this newly defined term as

$$\Delta \mathbf{a}_D^L = Y(\mathbf{x}, \dot{\mathbf{x}}, t)\, \boldsymbol{\theta} + \boldsymbol{\epsilon} = (\Gamma(\mathbf{x}, \dot{\mathbf{x}}, t) \otimes I_3)\, \boldsymbol{\theta} + \boldsymbol{\epsilon}, \tag{22}$$

where $\boldsymbol{\epsilon} \in \mathbb{R}^3$ represents an approximation error whose norm can be upper bounded by a constant, and $\boldsymbol{\theta} \in \mathbb{R}^{54}$. Finally, we substitute the (14) and (21) into the CW dynamics (2) to obtain the equation that is used in the subsequent control design section as

$$\ddot{\mathbf{x}} = C\dot{\mathbf{x}} + G\mathbf{x} + \mathbf{u} + Y(\mathbf{x}, \dot{\mathbf{x}}, t)\, \boldsymbol{\theta} + \boldsymbol{\epsilon}. \tag{23}$$

In this form, all the uncertainty is captured into the unknown coefficient vector $\boldsymbol{\theta}$. Although this differential drag model could be represented in a more compact form using additional knowledge about the system, as done in our previous work [16], here we allow the model to calibrate a significantly larger set of coefficients to test the online sparse identification approach we propose.

III. Control Design

A. Control objective

We introduce $\mathbf{x}_d(t)$ as the desired relative trajectory and define the position and velocity tracking errors as

$$\mathbf{e} \triangleq \mathbf{x} - \mathbf{x}_d, \qquad \dot{\mathbf{e}} \triangleq \dot{\mathbf{x}} - \dot{\mathbf{x}}_d, \tag{24}$$

and introduce the auxiliary state error

$$\mathbf{r} \triangleq \dot{\mathbf{e}} + \alpha\, \mathbf{e}, \tag{25}$$

where $\alpha \in \mathbb{R}^{3\times3}$ is a symmetric positive definite control gain. The control objective is to design an adaptive controller that ensures convergence of the error states and identification of a sparse vector of coefficients $\boldsymbol{\theta}(t)$. The control objective can then be set to

$$\|\boldsymbol{e}\| \to 0, \quad \|\boldsymbol{r}\| \to 0, \tag{26}$$

and

$$\|\tilde{\boldsymbol{\theta}}_{\boldsymbol{s}}\| \to 0 \tag{27}$$

where $\tilde{\boldsymbol{\theta}}_{\boldsymbol{s}} \in \mathbb{R}^q = \boldsymbol{\theta}_{\boldsymbol{s}} - \hat{\boldsymbol{\theta}}_s$, and $\boldsymbol{\theta}_s \in \mathbb{R}^q$ is a vector containing the $q \in \mathbb{Z}_{>0}$ most relevant coefficients in $\boldsymbol{\theta}$.

B. Control Development

Let us propose a two-stage approach as follows: A “warm-up" stage where the controller is focused on ensuring stability of the tracking errors $\boldsymbol{e}$ and $\boldsymbol{r}$, bounded estimates $\boldsymbol{\theta}$, and determining which are the most relevant coefficients $\boldsymbol{\theta}_{\boldsymbol{s}}$; and an “identification" stage where the controller aims to ensure convergence of $\boldsymbol{e}$, $\boldsymbol{r}$ and $\tilde{\boldsymbol{\theta}}_{\boldsymbol{s}}$.

This two-stage approach is inspired by the SINDy assumption that any dynamical system can be approximated by small number of active terms from a large library of functions. This assumption can be summarized in our context as follows:

Assumption 1: The dynamics in (23) can be represented using a subset of active nonlinear functions as

$$\ddot{\mathbf{x}} = C\dot{\mathbf{x}} + G\mathbf{x} + \mathbf{u} + Y_s(\mathbf{x}, \dot{\mathbf{x}}, t)\, \boldsymbol{\theta}_{\boldsymbol{s}} + \boldsymbol{\epsilon}_1, \tag{28}$$

where $Y_s(\mathbf{x}, \dot{\mathbf{x}}, t) \in \mathbb{R}^{3\times q}$ is a reduced regressor containing only the columns associated to the most relevant functions, and $\boldsymbol{\epsilon}_1 \in \mathbb{R}^3$ is the new approximation error whose norm can be upper bounded by a constant.

During the warm-up stage, no information about the most relevant functions is available. Therefore, the adaptive controller must be designed using the full dynamics in (23). Differentiating the equation (25) and plugging in the equation of motion (23), we obtain

$$\dot{\mathbf{r}} = \ddot{\mathbf{e}} + \alpha\,\dot{\mathbf{e}} = \ddot{\mathbf{x}} - \ddot{\mathbf{x}}_d + \alpha\,\dot{\mathbf{e}} \tag{29}$$

$$= C\,\dot{\mathbf{x}} + G\,\mathbf{x} + \mathbf{u} + Y(\mathbf{x}, \dot{\mathbf{x}}, t)\,\boldsymbol{\theta} + \boldsymbol{\epsilon} - \ddot{\mathbf{x}}_d + \alpha\,\dot{\mathbf{e}}. \tag{30}$$

Let us propose the following control law

$$\mathbf{u} = -C\,\dot{\mathbf{x}} - G\,\mathbf{x} - \alpha\,\dot{\mathbf{e}} - K_r\,\mathbf{r} - k_e\,\mathbf{e} + \ddot{\mathbf{x}}_d - Y(\mathbf{x}, \dot{\mathbf{x}}, t)\,\hat{\boldsymbol{\theta}}, \tag{31}$$

where $K_r \in \mathbb{R}^{3\times3}$ is a symmetric positive definite control gain, $k_e \in \mathbb{R}_{>0}$ is a scalar control gain, and $\hat{\boldsymbol{\theta}} \in \mathbb{R}^{54}$ is an estimate of $\boldsymbol{\theta}$. Plugging the proposed control law into the (30) to obtain the closed-loop error dynamics as

$$\dot{\mathbf{r}} = -K_r\,\mathbf{r} - k_e\,\mathbf{e} + Y(\mathbf{x}, \dot{\mathbf{x}}, t)\,\tilde{\boldsymbol{\theta}} + \boldsymbol{\epsilon}, \tag{32}$$

To address the uncertainties in differential drag, let us propose the following gradient-based adaptation law

$$\dot{\hat{\boldsymbol{\theta}}} = \text{proj}\left\{\gamma\left[\nabla^2\psi(\hat{\boldsymbol{\theta}})\right]^{-1} Y(\boldsymbol{x})^T\boldsymbol{r}\right\}. \tag{33}$$

where $\gamma \in \mathbb{R}_{>0}$ is a positive adaptation gain, $\text{proj}\{\cdot\}$ denotes a smooth projection algorithm such as the one in [34] that keeps the estimates within known user-defined bounds, and $\nabla^2\psi\left(\hat{\boldsymbol{\theta}}\right) \in \mathbb{R}^{54\times54}$ is the Hessian of the convex function $\psi\left(\hat{\boldsymbol{\theta}}\right) \in \mathbb{R}_{>0}$ defined as

$$\psi\left(\hat{\boldsymbol{\theta}}\right) = \sum_{i=1}^{54}\sqrt{\hat{\theta}_i^2 + \delta^2}, \quad \delta << 1, \tag{34}$$

whose motivation will be explained in the next section.

During the identification stage, we assume that the most relevant coefficients can be determined, enabling the representation of the open-loop error system as

$$\dot{\mathbf{r}} = C\,\dot{\mathbf{x}} + G\,\mathbf{x} + \mathbf{u} + Y_s(\mathbf{x}, \dot{\mathbf{x}}, t)\,\boldsymbol{\theta}_s + \boldsymbol{\epsilon}_1 - \ddot{\mathbf{x}}_d + \alpha\,\dot{\mathbf{e}}. \tag{35}$$

The control law can then be modified as

$$\mathbf{u} = -C\,\dot{\mathbf{x}} - G\,\mathbf{x} - \alpha\,\dot{\mathbf{e}} - K_r\,\mathbf{r} - k_e\,\mathbf{e} + \ddot{\mathbf{x}}_d - Y_s(\mathbf{x}, \dot{\mathbf{x}}, t)\,\hat{\boldsymbol{\theta}}_s, \tag{36}$$

resulting in the closed-loop error dynamics

$$\dot{\mathbf{r}} = -K_r\,\mathbf{r} - k_e\,\mathbf{e} + Y_s(\mathbf{x}, \dot{\mathbf{x}}, t)\,\tilde{\boldsymbol{\theta}}_s + \boldsymbol{\epsilon}_1, \tag{37}$$

and let us propose a CL-based adaptation law as

$$\dot{\hat{\boldsymbol{\theta}}}_{\boldsymbol{s}} = \text{proj}\left\{\gamma\left[\nabla^2\psi(\hat{\boldsymbol{\theta}}_{\boldsymbol{s}})\right]^{-1} Y_s(\boldsymbol{x}_i)^T\boldsymbol{r}_i + \gamma\left[\nabla^2\psi(\hat{\boldsymbol{\theta}}_s)\right]^{-1} K_{CL}\sum_{i=1}^{N_s} Y_{s_i}^T\left(\ddot{\boldsymbol{x}}_i - C\dot{\boldsymbol{x}}_i - G\boldsymbol{x}_i - \boldsymbol{u}_i - Y_{s_i}\hat{\boldsymbol{\theta}}_s\right)\right\} \tag{38}$$

where $N_s \in \mathbb{Z}_{>0}$ denotes a finite number of samples and $(\cdot)_i$ denotes the i^{th} sample of $(\cdot)$. This adaptation law "collects" input-output data through the summation to help identify, provided a Finite Excitation (FE) condition is satisfied, the real values of the coefficients in $\boldsymbol{\theta}_s$.

IV. Stability Analysis

A. Warm-up stage

To facilitate the subsequent stability analysis let $\boldsymbol{\eta} \in \mathbb{R}^6 = \left[\boldsymbol{e}^T,\ \boldsymbol{r}^T\right]^T$ be an augmented error state vector, $\chi = \min\left\{\lambda_{min}\left\{k_e\alpha\right\},\ \lambda_{min}\left\{K_r\right\} - (1/2)\right\},\ \lambda_{min}\{\cdot\}$ denote the minimum eigenvalue of $\{\cdot\}$, $c = (1/2)\bar{\epsilon}^2$, $\|\boldsymbol{\epsilon}\| \leq \bar{\epsilon}$, and $\underline{\kappa_w},\ \bar{\kappa}_w, \bar{\beta}_w \in \mathbb{R}_{>0}$ be known constants that bound the subsequent Lyapunov function $V : \mathbb{R}^{60} \to \mathbb{R}_{>0}$ as

$$\underline{\kappa_w}\|\boldsymbol{\eta}\|^2 \leq V \leq \bar{\kappa}_w\|\boldsymbol{\eta}\|^2 + \bar{\beta}_w, \tag{39}$$

where $\bar{\beta}_w$ can be used due to the projection algorithm in the adaptation law, Eq. (33).

Assumption 2: There exists a finite warm-up time $t_w \in \mathbb{R}_{>0}$ after which the time-varying adaptation gain $\left[\nabla^2\psi(\hat{\boldsymbol{\theta}})\right]^{-1}$ has adjusted its diagonal entries so that the q largest are associated to the q most relevant parameters.

Theorem 1. *Given the spacecraft relative dynamics in Eq.* (2)*, the control and adaptation laws in Eqs.* (31) *and* (33)*, respectively, yield a globally ultimately bounded result for the error states in* $\boldsymbol{\eta}$ *such that*

$$\|\boldsymbol{\eta}\| \leq \sqrt{\frac{\bar{\kappa}_w}{\underline{\kappa_w}}}\|\boldsymbol{\eta}(\boldsymbol{t_0})\|\exp\left(-\frac{\chi}{2\bar{\kappa}_w}(t - t_0)\right) + \sqrt{\frac{\bar{\beta}_w}{\underline{\kappa_w}}} + \sqrt{\frac{c\bar{\kappa}_w}{\chi\underline{\kappa_w}}}, \tag{40}$$

while the estimation error $\tilde{\boldsymbol{\theta}}$ *remains bounded. Moreover, under Assumption 2, the q most relevant s can be determined from* $\left[\nabla^2\psi(\hat{\boldsymbol{\theta}})\right]^{-1}$ *after* $t = t_w$.

Proof. Consider the following Lyapunov candidate function

$$V = \frac{k_e}{2}\boldsymbol{e}^T\boldsymbol{e} + \frac{1}{2}\boldsymbol{r}^T\boldsymbol{r} + \frac{1}{\gamma}d_\psi\left(\boldsymbol{\theta} \parallel \hat{\boldsymbol{\theta}}\right). \tag{41}$$

Here, we introduced the Bregman divergence term $d_\psi(\boldsymbol{\theta}\|\hat{\boldsymbol{\theta}}) \in \mathbb{R}_{>0}$, a non-Euclidean "distance" measure defined as

$$d_\psi\left(\boldsymbol{\theta} \parallel \hat{\boldsymbol{\theta}}\right) = \psi\left(\boldsymbol{\theta}\right) - \psi\left(\hat{\boldsymbol{\theta}}\right) - \left(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}\right)^T\nabla\psi\left(\hat{\boldsymbol{\theta}}\right), \tag{42}$$

where $\psi\left(\hat{\boldsymbol{\theta}}\right)$ is an arbitrary strongly convex function, in this case defined as in Eq. (34). This divergence is guaranteed to be non-negative for strongly convex functions, making it suitable for use in the Lyapunov function, Eq. (41).

It can be shown via Hadamard's Lemma that the time derivative of the Bregman divergence can be expressed in terms of the Hessian of $\psi\left(\hat{\boldsymbol{\theta}}\right)$, i.e., $\nabla^2\psi(\hat{\boldsymbol{\theta}})$, as [29]

$$\frac{d}{dt}d_\psi\left(\boldsymbol{\theta} \parallel \hat{\boldsymbol{\theta}}\right) = -\tilde{\boldsymbol{\theta}}^T\nabla^2\psi\left(\hat{\boldsymbol{\theta}}\right)\dot{\hat{\boldsymbol{\theta}}}, \tag{43}$$

where $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta} - \hat{\boldsymbol{\theta}}$. The Hessian $\nabla^2\psi(\hat{\boldsymbol{\theta}})$ for our convex function is

$$\nabla^2\psi(\hat{\boldsymbol{\theta}}) = \text{diag}\left(\frac{\delta^2}{(\hat{\theta}_1^2 + \delta^2)^{3/2}}, \cdots, \frac{\delta^2}{(\hat{\theta}_{3p}^2 + \delta^2)^{3/2}}\right). \tag{44}$$

The diagonal structure of $\left[\nabla^2\psi(\hat{\boldsymbol{\theta}})\right]^{-1}$ acts as a varying adaptation gain that "freezes adaptation" for a small $\hat{\theta}_i$ parameter in $\hat{\boldsymbol{\theta}}$ (i.e., small rate of change for small, near-zero entries) and allows stronger adaptation for larger and more relevant parameters, thereby promoting sparsity. This property can be exploited to determine the most relevant functions during adaptation, i.e., those associated to the largest q gains in $\left[\nabla^2\psi(\hat{\boldsymbol{\theta}})\right]^{-1}$ after an user-defined warm-up time $t = t_w$, as described in Assumption 2.

Taking the time derivative of Eq. (41), plugging in Eq. (32), and using the definition in Eq. (25) yields

$$\dot{V} = k_e\boldsymbol{e}^T\left(\mathbf{r} - \alpha\mathbf{e}\right) + \boldsymbol{r}^T\left(-K_r\,\mathbf{r} - k_e\,\mathbf{e} + Y(\mathbf{x}, \dot{\mathbf{x}}, t)\,\tilde{\boldsymbol{\theta}} + \boldsymbol{\epsilon}\right) - \frac{1}{\gamma}\tilde{\boldsymbol{\theta}}^T\nabla^2\psi\left(\hat{\boldsymbol{\theta}}\right)\dot{\hat{\boldsymbol{\theta}}}, \tag{45}$$

and further substituting the adaptation law in Eq. (33) yields

$$\dot{V} = -k_e \boldsymbol{e}^T \alpha \mathbf{e} - \mathbf{r}^T K_r \mathbf{r} + \mathbf{r}^T \boldsymbol{\epsilon}. \tag{46}$$

Applying Cauchy-Schwarz and Young's inequalities such that $\boldsymbol{r}^T \boldsymbol{\epsilon} \leq \frac{1}{2}\|r\|^2 + c$, the above expression can be rewritten as

$$\dot{V} \leq -\chi \|\boldsymbol{\eta}\|^2 + c. \tag{47}$$

Using the bounds in Eq. (39) yields

$$\dot{V} \leq -\frac{\chi}{\bar{\kappa}_w} V + \left(\frac{\chi \bar{\beta}_w}{\bar{\kappa}_w} + c \right), \tag{48}$$

and by Comparison lemma we obtain

$$V \leq \left(V(t_0) - \bar{\beta}_w - \frac{c\bar{\kappa}_w}{\chi} \right) \exp\left(-\frac{\chi}{\bar{\kappa}_w}(t - t_0) \right) + \bar{\beta}_w + \frac{c\bar{\kappa}_w}{\chi}, \tag{49}$$

using the bounds of V again on both sides yields the expression in Eq. (40). This results in the error states, i.e., $\|\boldsymbol{\eta}\|$ converging exponentially up to an ultimate bound of size $\sqrt{\bar{\beta}_w / \underline{\kappa_w}} + \sqrt{c\bar{\kappa}_w / \chi \underline{\kappa_w}}$.

□

B. Identification Stage

To facilitate the subsequent stability analysis let $\underline{\kappa}$, $\bar{\kappa}$, $\bar{\beta} \in \mathbb{R}_{>0}$ be known constants that bound the subsequent Lyapunov function $V_1 : \mathbb{R}^{6+q} \to \mathbb{R}_{>0}$ as

$$\underline{\kappa}\|\boldsymbol{\eta}\|^2 \leq V_1 \leq \bar{\kappa}\|\boldsymbol{\eta}\|^2 + \bar{\beta}, \tag{50}$$

where $\bar{\beta}$ can be used due to the projection algorithm in the adaptation law, Eq. (38).

Assumption 3: There exists some finite time $T \in \mathbb{R}_{>0} > t_w$ such that the following FE condition is satisfied

$$\lambda_{min} \left\{ \sum_{i=1}^{N_s} Y_{s_i}^T Y_{s_i} \right\} \geq \bar{\lambda}, \quad \forall\, t > T \tag{51}$$

where $\bar{\lambda} \in \mathbb{R}_{>0}$ is some user-defined constant.

Consider also the alternative bounds for the Lyapunov function

$$\underline{\zeta}\|z\|^2 \leq V_1 \leq \bar{\zeta}\|z\|^2, \tag{52}$$

where $z = \left[\boldsymbol{e}^T, \boldsymbol{r}^T, \tilde{\boldsymbol{\theta}}_s\right]^T$, and the fact that estimates $\hat{\boldsymbol{\theta}}_s$ remain in a compact set, enforced by the projection algorithm, was used to bound d_ψ with quadratic expressions of $\left\|\tilde{\boldsymbol{\theta}}_s\right\|$.

The two standard theorems for CL-based adaptive controllers are subsequently presented, Theorem 2 covers the period $t_w < t \leq T$ (i.e., before the FE condition is satisfied), and Theorem 3 presents the stability result for $t > T$

Theorem 2. *Given the spacecraft relative dynamics in Eq.* (2)*, the control and adaptation laws in Eqs.* (36) *and* (38)*, respectively, yield a globally ultimately bounded result for the error states in* $\boldsymbol{\eta}$ *such that*

$$\|\boldsymbol{\eta}\| \leq \sqrt{\frac{\bar{\kappa}}{\underline{\kappa}}} \|\boldsymbol{\eta}(\boldsymbol{t_w})\| \exp\left(-\frac{\chi}{2\bar{\kappa}}(t - t_w) \right) + \sqrt{\frac{\bar{\beta}}{\underline{\kappa}}} + \sqrt{\frac{c_1 \bar{\kappa}}{\chi \underline{\kappa}}}, \quad \forall\, t_w < t \leq T \tag{53}$$

while the estimation error $\tilde{\boldsymbol{\theta}}_{\boldsymbol{s}}$ *remains bounded.*

Proof. Consider the modified Candidate Lyapunov function that accounts only for the most relevant subset of $\tilde{\boldsymbol{\theta}}$, namely $\tilde{\boldsymbol{\theta}}_{\boldsymbol{s}}$, which contains the q most relevant functions from the library

$$V_1 = \frac{k_e}{2}\boldsymbol{e}^T\boldsymbol{e} + \frac{1}{2}\boldsymbol{r}^T\boldsymbol{r} + \frac{1}{\gamma}d_\psi\left(\boldsymbol{\theta_s} \parallel \hat{\boldsymbol{\theta}}_s\right). \tag{54}$$

Taking the time derivative, plugging in the closed loop error system in Eq. (37), using the definition in Eq. (25), and substituting the control law in Eq. (36) yields

$$\dot{V}_1 = k_e\boldsymbol{e}^T\left(\mathbf{r} - \alpha\mathbf{e}\right) + \boldsymbol{r}^T\left(-K_r\,\mathbf{r} - k_e\,\mathbf{e} + Y_s(\mathbf{x}, \dot{\mathbf{x}}, t)\,\tilde{\boldsymbol{\theta}}_s + \boldsymbol{\epsilon}_1\right) - \frac{1}{\gamma}\tilde{\boldsymbol{\theta}}_s^T\nabla^2\psi(\hat{\boldsymbol{\theta}}_s)\dot{\hat{\boldsymbol{\theta}}}_s. \tag{55}$$

Consider the non-implementable form of the adaptation law in Eq. (38)

$$\dot{\hat{\boldsymbol{\theta}}}_{\boldsymbol{s}} = \text{proj}\left\{\gamma\left[\nabla^2\psi(\hat{\boldsymbol{\theta}}_{\boldsymbol{s}})\right]^{-1}Y_s(\boldsymbol{x}_i)^T\boldsymbol{r}_i + \gamma\left[\nabla^2\psi(\hat{\boldsymbol{\theta}}_s)\right]^{-1}K_{CL}\sum_{i=1}^{N_s}Y_{s_i}^T\left(Y_{s_i}\tilde{\boldsymbol{\theta}}_s + \boldsymbol{\epsilon}_{\mathbf{1}_i}\right)\right\}. \tag{56}$$

Substituting Eq. (56) into the Lyapunov derivative yields

$$\dot{V}_1 = -k_e\boldsymbol{e}^T\alpha\mathbf{e} - \mathbf{r}^TK_r\mathbf{r} + \mathbf{r}^T\boldsymbol{\epsilon}_1 - \tilde{\boldsymbol{\theta}}_{\boldsymbol{s}}{}^TK_{CL}\sum_{i=1}^{N_s}Y_{s_i}^TY_{s_i}\tilde{\boldsymbol{\theta}}_{\boldsymbol{s}} - \tilde{\boldsymbol{\theta}}_s^T\sum_{i=1}^{N_s}Y_{s_i}{}^T\boldsymbol{\epsilon}_{\mathbf{1}_i}. \tag{57}$$

Before $t = T$, the matrix $\sum_{i=1}^{N_s}Y_{s_i}^TY_{s_i}$ is at least positive semi-definite. Therefore, the corresponding term can be upper bounded by zero during this period. The term $\tilde{\boldsymbol{\theta}}_s^T\sum_{i=1}^{N_s}Y_{s_i}{}^T\boldsymbol{\epsilon}_{\mathbf{1}_i}$ can be upper bounded as

$$\left\|\tilde{\boldsymbol{\theta}}_s^T\sum_{i=1}^{N_s}Y_{s_i}{}^T\boldsymbol{\epsilon}_{\mathbf{1}_i}\right\| \leq \xi. \tag{58}$$

where the facts that $\tilde{\boldsymbol{\theta}}_{\boldsymbol{s}}$ remains bounded due to the projection algorithm, the orbital quantities in Y_{s_i} are physically limited, as well as $\|\boldsymbol{\epsilon}_{\mathbf{1}}\| \leq \bar{\epsilon}_1$ with $\bar{\epsilon}_1 \in \mathbb{R}_{>0}$, were used.

The Lyapunov function derivative can then be upper bounded as follows

$$\dot{V}_1 \leq -\chi\|\boldsymbol{\eta}\|^2 + c_1, \tag{59}$$

where $c_1 = \xi + \frac{1}{2}\bar{\epsilon}_1^2$.

From this point on, the procedure to find the upper bound of $\|\boldsymbol{\eta}\|$ is identical as the one presented for Theorem 1, but using the Lyapunov function bounds from Eq. (50), obtaining the result in (53).

□

Theorem 3. *Given the spacecraft relative dynamics in Eq.* (2)*, the control and adaptation laws in Eqs.* (36) *and* (38)*, respectively, yield a globally ultimately bounded result for the error states in* $\boldsymbol{\eta}$ *and estimation error* $\tilde{\boldsymbol{\theta}}_s$ *such that*

$$\|z\| \leq \sqrt{\frac{\bar{\zeta}}{\underline{\zeta}}}\|\boldsymbol{z(T)}\|\exp\left(-\frac{\chi_1}{2\bar{\zeta}}(t-T)\right) + \sqrt{\frac{c_1\bar{\zeta}}{\chi_1\underline{\zeta}}},\ \ \forall\, t > T \tag{60}$$

Proof. Let us start from Eq. (57). Now, after the FE condition in Assumption 3 (i.e., Eq. (51)) is satisfied, the matrix $\sum_{i=1}^{N_s}Y_{s_i}^TY_{s_i}$ is guaranteed to be positive-definite . Therefore, $\dot{V}_1$ can be upper bounded as

$$\dot{V}_1 \leq -\chi_1\|\boldsymbol{z}\|^2 + c_1, \tag{61}$$

where $\chi_1 = \min\left\{\chi, \lambda_{min}\left\{K_{CL}\sum_{i=1}^{N_s}Y_{s_i}^TY_{s_i}\right\}\right\}$. Using the bounds of V_1 in Eq. (52) yields

$$\dot{V}_1 \leq -\frac{\chi_1}{\bar{\zeta}}V_1 + c_1, \tag{62}$$

and by Comparison lemma we get

$$V_1 \leq \left(V_1(T) - \frac{c_1\bar{\zeta}}{\chi_1}\right)\exp\left(-\frac{\chi_1}{\bar{\zeta}}(t-T)\right) + \frac{c_1\bar{\zeta}}{\chi_1}, \tag{63}$$

and further using the bounds in Eq. (52) yields the result in Eq. (60).

□

V. Simulation

To validate the proposed controller, we conducted a numerical simulation of the relative orbit motion of the chaser-target spacecraft in low Earth orbit (LEO). The true translational dynamics of both spacecraft are propagated in the Earth–Centered Inertial (ECI) frame using nonlinear two–body gravity that includes the effect of atmospheric drag. The simulation is executed for 350 target orbits, equivalent to approximately 548 hours (≈ 22.8 days). This simulation duration is chosen to provide sufficient time for the controller to learn/estimate the dynamics of the differential drag.

The initial orbit of the target spacecraft is a near-circular orbit, for which the orbital elements were chosen to match the International Space Station (ISS) orbit with an altitude of approximately 470 km. The orbit elements of the target are listed in Table 1. The chaser is initialized in a closeby orbit by randomly applying a small offset to the target's semi-major axis, eccentricity, inclination, and true anomaly, while keeping the same values of RAAN and argument of perigee. Specifically, the variation of the chaser position are set to $a_c = a_t + [0, 10]$ [km], $e_c = e_t + [0, 2 \times 10^{-4}]$, $\nu_c = \nu_t \pm 0.04$ [deg], respectively, and $i_c = i_t - 0.01$ [deg].

Table 1 Initial conditions for the target spacecraft.

a_t [km]	e_t	i_t [deg]	Ω_t [deg]	ω_t [deg]	ν_t [deg]
6.845×10^3	5.0×10^{-4}	51.64	229.18	286.07	0

Table 2 lists the parameters used to compute the true drag acceleration. In the simulations, the two spacecraft have identical mass, but their drag coefficients are slightly different. The cross-sectional areas of both spacecraft are set to vary in time to emulate tumbling and/or attitude maneuvers. The atmospheric density is simulated using the Harris–Priester model[12]. As a result, the differential drag accelerations are time-varying and unknown to the controller. The controller only relies on the SINDy-inspired library of functions to achieve state tracking and estimate the differential drag.

Table 2 Spacecraft physical properties.

Parameter	Target	Chaser
Mass m [kg]	50	50
Average area A [m^2]	2.0	2.8
Drag coefficient C_D	2.2	2.1

To rendezvous with the target, the chaser uses the control laws developed in Section III depending on the stage. The control gains are presented in Table 3. The adaptation law combines a Bregman-based gradient term with a concurrent learning term. The Bregman-Concurrent Learning adaptation gains are summarized in Table 4. Initially, the CL-based input-out pair accumulation term is disabled for a duration referred to as the warm-up period, which is set to 80 target orbital periods (T_{orb}). In all simulations, the CL term is kept fixed, i.e., no additional data accumulation through the summation, after the FE condition threshold $\bar{\lambda}$ is first satisfied.

Table 3 Controller gains.

Gains	Value
α	$1.0 \times 10^{-3} \mathcal{I}$
K_r	$1.0 \times 10^{-8} \mathcal{I}$
k_e	3.0×10^{-6}

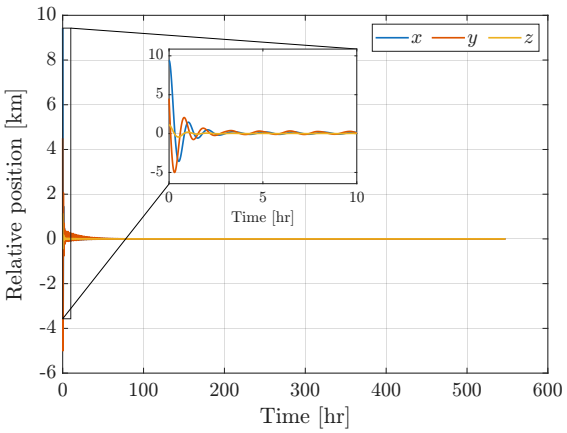
Table 4 Bregman–Concurrent Learning gains.

Parameter	Value
FE threshold $\bar{\lambda}$	600
CL gain K_{CL}	$10\mathcal{I}$
Learning gain γ	5.0×10^{-6}
Bregman regularization δ	1.0×10^{-5}
q	8
Warm-up duration	$80\,T_{\text{orb}}$

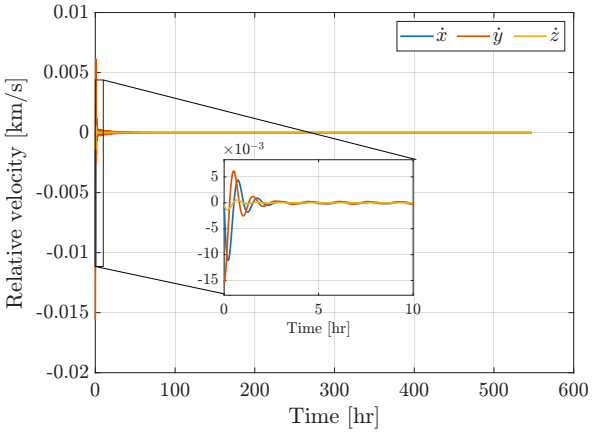
VI. Results

A. Case A: Bregman-CL with warm-up + activated coefficient selection

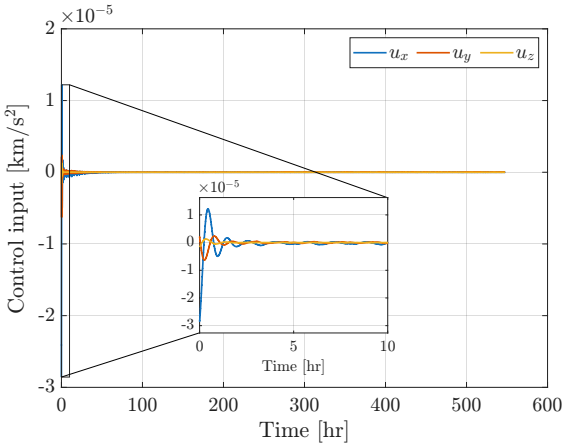
The rendezvous maneuvers of the chaser in the LVLH frame over 350 orbits is shown ing Fig. 2, with Fig. 2a showing the relative positions, Fig. 2b plotting the relative velocities, and Fig. 2d showing the trajectory plot. The controller drives the chaser along a decaying spiral, and its trajectory converges smoothly toward the small neighborhood of the origin within the first tens of hours. As seen in Fig. 2c, the control inputs remained bounded and decreased to zero.



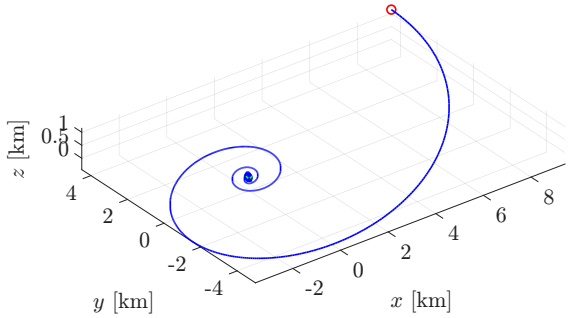
(a) Relative positions



(b) Relative velocities



(c) Control input

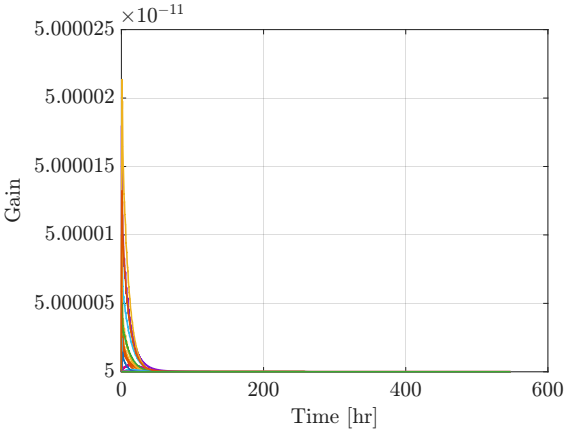


(d) Trajectory

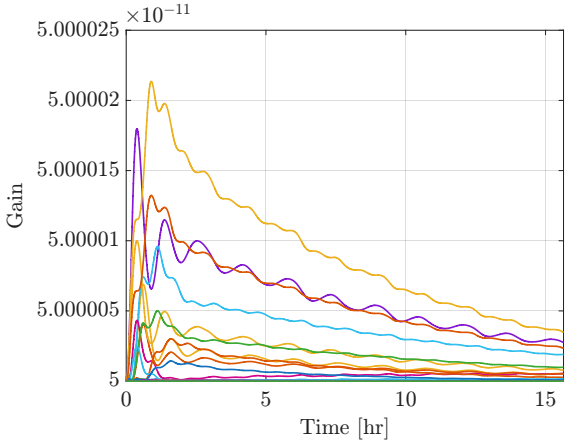
Fig. 2 Rendezvous maneuvers of the chaser spacecraft in LVLH frame.

The time history of the diagonal entries of $\left[\nabla^2\psi(\hat{\boldsymbol{\theta}})\right]^{-1}$ is shown in Fig. 3. Figure 3a shows overall trends of

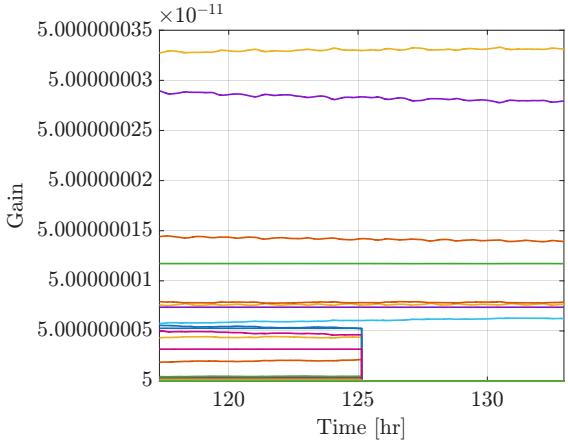
Bregman's gains, Fig 3b provides a zoomed-in view during the first 10 orbits and its decay, and Fig 3c shows the gains around the end of the warm-up time t_w. It can be seen that during this phase, Bregman's gains (i.e., $\left[\nabla^2\psi(\hat{\theta})\right]^{-1}$) evolve rapidly and then decay toward a nearly constant value. At the end of the warm-up time (after 80 orbits), as shown in Fig 3c, it is noticeable that some terms are more active than others, which is a result of the Bregman sparsity enforcement. This illustrates how introducing the Bregman term helps automatically promote sparsity. The controller can select the gains that are more important without any manual gain tuning, which is quite effective, particularly in this scenario, where we have a large number of coefficients (54 in this case). Among the 54, only 8 were used and the irrelevant gains are reset to its initial value for the rest of the simulation as seen Fig. 3c.



(a) Bregman's gains over time



(b) Zoomed-in view at the first 10 orbits



(c) Zoomed-in view at the end of warm-up time

Fig. 3 Behavior of Bregman's gains.

Figure 4 shows the FE metric $\lambda_{min}\left\{\sum_{i=1}^{N_s} Y_{s_i}^T Y_{s_i}\right\}$. Initially, during the warm-up stage, $\lambda_{min}\left\{\sum_{i=1}^{N_s} Y_{s_i}^T Y_{s_i}\right\}$ is disabled, therefore, it is remains as zero. Once the warm-up time is completed (after 80 orbits), the most active coefficients, ranked based on the Bregman's gains (15 % of all coefficients) are selected to form the CL term's input-output pairs. As seen in Fig. 4, the FE metric began to grow until it reached the threshold value, at which the CL term was enabled and summation truncated.

The estimated values of the 54 drag-related parameters are shown in Fig. 5, with overall time history in Fig. 5a, and a zoomed-in view around the CL activation in Fig. 5b. The coefficients begin with rapid dynamic variation in the first ten hours and converge to nearly constant values during the warm-up phase. The zoomed-in view in Fig. 5b shows the exponential convergence of a small set of coefficients when the CL term is activated, as described in the stability analysis section. As a result, despite the large number of coefficients, the controller automatically selects a few dominant terms that represent the drag disturbance and effectively discards the less relevant terms.

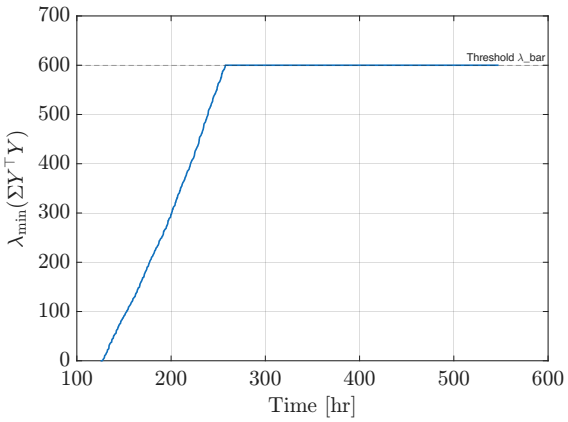
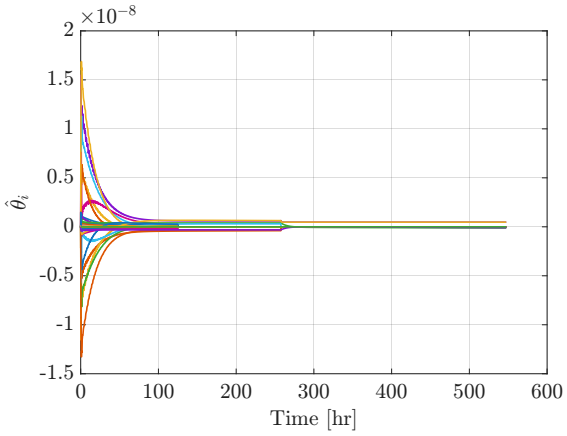
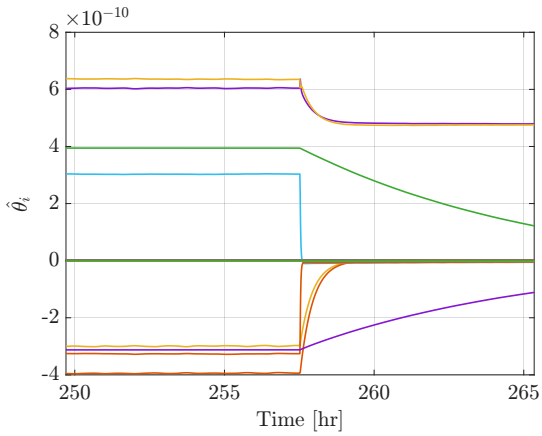


Fig. 4 Behavior of the $\lambda_{min}\left\{\sum_{i=1}^{N_s} Y_{s_i}^T Y_{s_i}\right\}$



(a) Unknown parameters $\hat{\Theta}$ over time

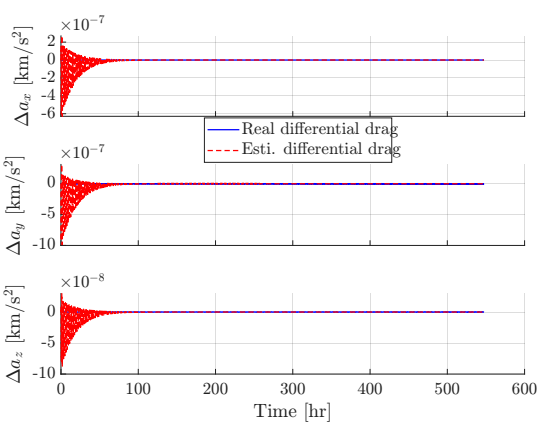


(b) Zoomed-in view when the CL term is activated

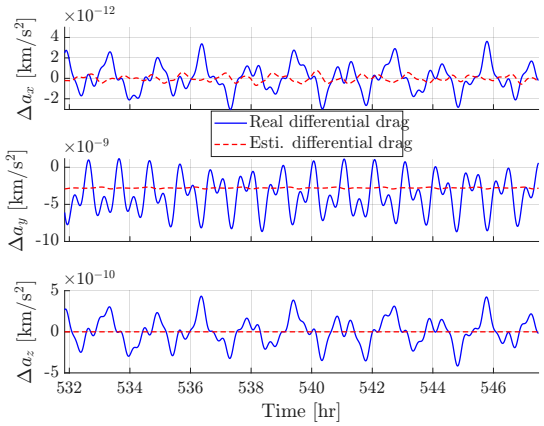
Fig. 5 Behavior of the unknown parameters $\hat{\Theta}$.

Figure 6a shows the real and reconstructed differential drag in the LVLH frame. At the beginning, there is a significant mismatch between the real and estimated drag because the controller is still adapting, and CL is disabled. However, the estimation gradually moves closer to the truth value as time evolves, and after the CL term is activated. The zoomed-in view for the last 10 orbits is shown Fig. 6b. The radial component oscillates around zero, the cross-track is zero in this case, while the along-track component has a negative bias with a similar oscillatory pattern. In particular, the controller correctly estimates that the drag has the largest value contributed in the along-track (y-component), as expected for the differential drag. The remaining mismatch is mainly caused by the simplified choice of the library function and density model, and could be reduced further with additional tuning of the controller.

Based on the selected coefficient from the library $\Gamma(\mathbf{x}, \dot{\mathbf{x}}, t)$, the identified equation for the sparse representation of differential drag $\Delta\mathbf{a}_D^L$ are given in (64)–(66). As expected from the structure of the differential drag in the LVLH frame, the largest coefficients are in the along-track (y-component). Unknown to the controller which are the most important terms, it is evident that the controller can correctly capture the along-track relative-velocity terms of both chaser $K_c(t)\, v^L_{\text{rel},cy}$ and target $K_t(t)\, v^L_{\text{rel},ty}$, respectively. The identified coefficient in radial (x-component) is relatively smaller than those in the equation (65), which is almost negligible in this scenario, and it indicates that it captures the numerical noise.



(a) Real vs estimated differential drag over time



(b) Zoomed-in view for the last 10 orbits

Fig. 6 Real vs reconstructed estimated differential drag in LVLH frame.

$$\Delta\hat{a}^L_{D,x} = 10^{-12}\Big(1.03\,K_c(t)\,v^L_{\text{rel},cy} + 1.36\,K_t(t)\,v^L_{\text{rel},ty}\Big), \tag{64}$$

$$\begin{aligned}\Delta\hat{a}^L_{D,y} = 10^{-10}\Big(&4.759\,K_c(t)\,v^L_{\text{rel},cy} + 4.763\,K_t(t)\,v^L_{\text{rel},ty}\\ &- 1.226\,K_c(t)\,v^L_{\text{rel},cz}\,C(t) - 0.829\,K_t(t)\,v^L_{\text{rel},tz}\,C(t)\\ &- 0.0345\,K_t(t)\,v^L_{\text{rel},ty}\,S(t) - 0.0233\,K_t(t)\,v^L_{\text{rel},ty}\,C(t)\Big),\end{aligned} \tag{65}$$

$$\Delta\hat{a}^L_{D,z} = 0. \tag{66}$$

B. Case B: Bregman CL without warm-up and no sparsity enforcement

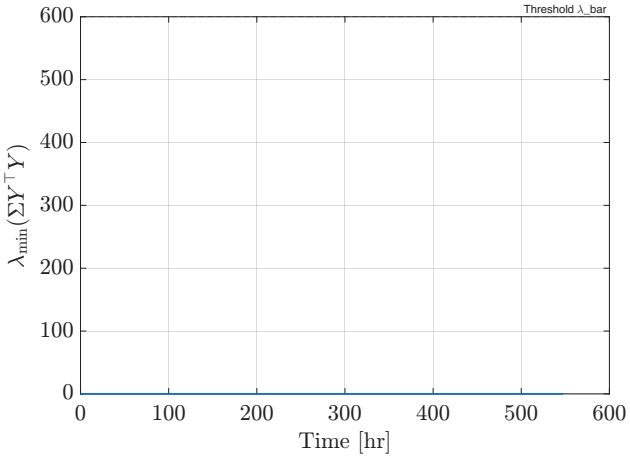


Fig. 7 Behavior of the $\lambda_{min}\left\{\sum_{i=1}^{N_s} Y_{s_i}^T Y_{s_i}\right\}$

To illustrate the importance of the warm-up stage and sparsity enforcement, we ran a baseline case without having warm-up stage and the controller was forced to use all coefficient in the library. This setup is used to show whether the controller can still learn when the library contains a large set of coefficient. In this baseline case, the controller still ensures relative state regulation and keeps the control input bounded with similar behavior as seen in the Fig. 2,

therefore, the plots are omitted for brevity. However, as shown in the Fig. 7, the the FE metric $\lambda_{min}\left\{\sum_{i=1}^{N_s} Y_{s_i}^T Y_{s_i}\right\}$ remains at zero. This is because a large library contains many unknown coefficient and the maneuvers does not provide sufficient independent information to separate each term. As result, many coefficient become similar to each other and the controller cannot uniquely identify each term correctly. The baseline scenario show the importance of the warm-up stage in helping identify the active terms in order to reduce the number of coefficients that the controller needs to learn.

Overall, simulation results confirm that the proposed Bregman-CL adaptation law is effective as it can handle a library with large number of coefficient and automatically choose a small set of influential coefficients to recover the differential drag while significantly reduce the effort of manual tuning or coefficient (i.e., dominant nonlinear functions) selection.

VII. Conclusion

In this study, we addressed the problem of learning a differential drag model online while performing a rendezvous maneuver in low Earth orbit. Building on the adaptive concurrent–learning framework of [16], we incorporated a SINDy–inspired library and a Bregman sparsity enforcement to design an adaptive controller that simultaneously regulates the relative motion and identifies a sparse representation of the drag disturbance. First, we proposed a Bregman–CL adaptation law that can handle a differential drag disturbance modeled using a large library of functions. The approach combines the warm-up phase, in which only the gradient and Bregman terms are active, with a subsequent concurrent learning phase. Second, we introduced a Bregman-based feature selection that involves automatically ranking the coefficients using the inverse of Hessian, which results in a small set of coefficients retained in the regressor. Third, we demonstrated in a rendezvous scenario that the proposed controller can drive the chaser to the origin, and reconstructed the dominant structure of the differential drag acceleration in the LVLH frame. The results show that ML-inspired ideas such as SINDy can be embedded into adaptive control through the Bregman framework while ensuring stability. By enabling data collection "on the fly" while maintaining stability, this approach supports the safe development of data-driven methods in safety-critical applications, such as those in space, where large datasets are often unavailable. Future work will extend this framework in several directions. First, we will consider expanding the library to include more functions that can handle more complicated scenarios with higher fidelity. Second, we will further evaluate the proposed approach to Differential Drag-based maneuvering and other aerospace control applications to assess the generalization and robustness of the approach. Finally, we will implement the proposed Bregman–CL controller in Software–in–the–Loop and, ultimately, on–orbit experiments to further validate its suitability for real spacecraft guidance and control.

References

[1] Leonard, C. L., Hollister, M., and Bergman, E. V., "Orbital Formationkeeping with Differential Drag," *Journal of Guidance, Control and Dynamics*, Vol. 12, No. 1, 1989, pp. 108–113. https://doi.org/10.2514/3.20374.

[2] Bevilacqua, R., and Romano, M., "Rendezvous Maneuvers of Multiple Spacecraft by Differential Drag under J2 Perturbation," *Journal of Guidance, Control and Dynamics*, Vol. 31, No. 6, 2008, pp. 1595–1607. https://doi.org/10.2514/1.36362.

[3] Perez, D., and Bevilacqua, R., "Differential Drag Spacecraft Rendezvous Using an Adaptive Lyapunov Control Strategy," *Acta Astronautica*, Vol. 83, 2012, pp. 196–207. https://doi.org/10.1016/j.actaastro.2012.09.005.

[4] Harris, M. W., and Açıkmeşe, B., "Minimum TIme Rendezvous of Multiple Spacecraft Using Differential Drag," *Journal of Guidance, Control and Dynamics*, Vol. 37, No. 2, 2014, pp. 365–373. https://doi.org/10.2514/1.61505.

[5] Maclay, T. D., and Tuttle, C., "Satellite Stationkeeping of the ORBCOMM Constellation via Active Control of Atmospheric Drag: Operations, Constraints, and Performance," *Advances in the Astronautical Sciences*, Vol. 120, 2005, pp. 763–773. AAS Paper 05-152.

[6] Foster, C., Mason, J., Vittaldev, V., Beukelaers, L., Stepan, L., and Zimmerman, R., "Constellation Phasing with Differential Drag on Planet Labs Satellites," *Journal of Spacecraft and Rockets*, Vol. 55, No. 2, 2018, pp. 473–483. https://doi.org/10.2514/1.A33927.

[7] Schweighart, S. A., and Sedwick, R. J., "High-Fidelity Linearized J Model for Satellite Formation Flight," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 6, 2002, pp. 1073–1080. https://doi.org/10.2514/2.4986, URL https://doi.org/10.2514/2.4986.

[8] Horsley, M., "An Investigation into Using Differential Drag for Controlling a Formation of CubeSats," *Proceedings of the Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS)*, Maui, HI, USA, 2011.

[9] Varma, S., and Kumar, K. D., "Multiple Satellite Formation Flying Using Differential Aerodynamic Drag," *Journal of Spacecraft and Rockets*, Vol. 49, No. 2, 2012, pp. 325–336. https://doi.org/10.2514/1.52395.

[10] Gaias, G., Ardaens, J.-S., and Montenbruck, O., "Model of J2 perturbed satellite relative motion with time-varying differential drag," *Celestial Mechanics and Dynamical Astronomy*, Vol. 123, 2015, pp. 441–433. https://doi.org/10.1007/s10569-015-9643-2.

[11] Ivanov, D., Kushniruk, M., and Ovchinnikov, M., "Study of satellite formation flying control using differential lift and drag," *Acta Astronautica*, Vol. 152, 2018, pp. 88–100. https://doi.org/10.1016/j.actaastro.2018.07.047.

[12] Harris, I., and Priester, W., "Time-dependent structure of the upper atmosphere," *Journal of the Atmospheric Sciences*, Vol. 19, No. 4, 1962, pp. 286–301. https://doi.org/10.1175/1520-0469(1962)019<0286:TDSOTU>2.0.CO;2.

[13] Picone, J. M., Hedin, A. E., Drob, D. P., and Aikin, A. C., "NRLMSISE-00 empirical model of the atmosphere: Statistical comparisons and scientific issues," *Journal of Geophysical Research: Space Physics*, Vol. 107, No. A12, 2002, pp. SIA 15–1–SIA 15–16. https://doi.org/https://doi.org/10.1029/2002JA009430, URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2002JA009430.

[14] Pérez, D., and Bevilacqua, R., "Neural Network based calibration of atmospheric density models," *Acta Astronautica*, Vol. 110, 2015, pp. 58–76. https://doi.org/https://doi.org/10.1016/j.actaastro.2014.12.018, URL https://www.sciencedirect.com/science/article/pii/S0094576515000077, dynamics and Control of Space Systems.

[15] Vallado, D. A., and Finkleman, D., "A critical assessment of satellite drag and atmospheric density modeling," *Acta Astronautica*, Vol. 95, 2014, pp. 141–165.

[16] Riano-Rios, C., Bevilacqua, R., and Dixon, W. E., "Differential drag-based multiple spacecraft maneuvering and on-line parameter estimation using integral concurrent learning," *Acta Astronautica*, Vol. 174, 2020, pp. 189–203. https://doi.org/https://doi.org/10.1016/j.actaastro.2020.04.059, URL https://www.sciencedirect.com/science/article/pii/S0094576520302745.

[17] Sun, R., Riano-Rios, C., Bevilacqua, R., Fitz-Coy, N. G., and Dixon, W. E., "CubeSat Adaptive Attitude Control with Uncertain Drag Coefficient and Atmospheric Density," *Journal of Guidance, Control, and Dynamics*, Vol. 44, No. 2, 2021, pp. 379–388. https://doi.org/10.2514/1.G005515.

[18] Riano-Rios, C., Bevilacqua, R., and Dixon, W. E., "Adaptive control for differential drag-based rendezvous maneuvers with an unknown target," *Acta Astronautica*, Vol. 181, 2021, pp. 733–740. https://doi.org/https://doi.org/10.1016/j.actaastro.2020.03.011, URL https://www.sciencedirect.com/science/article/pii/S0094576520301399.

[19] Parikh, A., Kamalapurkar, R., and Dixon, W. E., "Integral concurrent learning: Adaptive control with parameter convergence using finite excitation," *International Journal of Adaptive Control and Signal Processing*, Vol. 33, No. 12, 2019, pp. 1775–1787. https://doi.org/https://doi.org/10.1002/acs.2945, URL https://onlinelibrary.wiley.com/doi/abs/10.1002/acs.2945.

[20] Brunton, S. L., Proctor, J. L., and Kutz, J. N., "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," *Proceedings of the National Academy of Sciences*, Vol. 113, No. 15, 2016, pp. 3932–3937. https://doi.org/10.1073/pnas.1517384113, URL https://www.pnas.org/doi/abs/10.1073/pnas.1517384113.

[21] Brunton, S. L., Proctor, J. L., and Kutz, J. N., "Sparse Identification of Nonlinear Dynamics with Control (SINDYc)," *IFAC-PapersOnLine*, Vol. 49, No. 18, 2016, pp. 710–715. https://doi.org/10.1016/j.ifacol.2016.10.249.

[22] Fasel, U., Kaiser, E., Kutz, J. N., Brunton, B. W., and Brunton, S. L., "SINDy with Control: A Tutorial," *2021 60th IEEE Conference on Decision and Control (CDC)*, 2021, pp. 16–21. https://doi.org/10.1109/CDC45484.2021.9683120.

[23] Kaiser, E., Kutz, J. N., and Brunton, S. L., "Sparse identification of nonlinear dynamics for model predictive control in the low-data limit," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, Vol. 474, No. 2219, 2018, p. 20180335. https://doi.org/10.1098/rspa.2018.0335.

[24] Bregman, L., "The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming," *USSR Computational Mathematics and Mathematical Physics*, Vol. 7, No. 3, 1967, pp. 200–217. https://doi.org/https://doi.org/10.1016/0041-5553(67)90040-7, URL https://www.sciencedirect.com/science/article/pii/0041555367900407.

[25] Gaudio, J. E., Gibson, T. E., Annaswamy, A. M., Bolender, M. A., and Lavretsky, E., "Connections Between Adaptive Control and Optimization in Machine Learning," *2019 IEEE 58th Conference on Decision and Control (CDC)*, 2019, pp. 4563–4568. https://doi.org/10.1109/CDC40024.2019.9029197.

[26] Fradkov, A. L., "Early History of Machine Learning," *IFAC-PapersOnLine*, Vol. 53, No. 2, 2020, pp. 1385–1390. https://doi.org/https://doi.org/10.1016/j.ifacol.2020.12.1888, URL https://www.sciencedirect.com/science/article/pii/S2405896320325027, 21st IFAC World Congress.

[27] Annaswamy, A. M., and Fradkov, A. L., "A historical perspective of adaptive control and learning," *Annual Reviews in Control*, Vol. 52, 2021, pp. 18–41. https://doi.org/https://doi.org/10.1016/j.arcontrol.2021.10.014, URL https://www.sciencedirect.com/science/article/pii/S1367578821000894.

[28] Lee, T., Kwon, J., and Park, F. C., "A Natural Adaptive Control Law for Robot Manipulators," *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 1–9. https://doi.org/10.1109/IROS.2018.8593727.

[29] Boffi, N. M., and Slotine, J.-J. E., "Implicit Regularization and Momentum Algorithms in Nonlinearly Parameterized Adaptive Control and Prediction," *Neural Computation*, Vol. 33, No. 3, 2021, pp. 590–673. https://doi.org/10.1162/neco_a_01360.

[30] Fradkov, A., "Lyapunov-Bregman functions for speed-gradient adaptive control of nonlinear time-varying systems," *IFAC-PapersOnLine*, Vol. 55, No. 12, 2022, pp. 544–548.

[31] Clohessy, W. H., and Wiltshire, R. S., "Terminal Guidance System for Satellite Rendezvous," *Journal of the Aerospace Sciences*, Vol. 27, No. 9, 1960, pp. 653–658. https://doi.org/10.2514/8.8704, URL https://doi.org/10.2514/8.8704.

[32] Vallado, D. A., *Fundamentals of astrodynamics and applications*, Vol. 12, Springer Science & Business Media, 2001.

[33] Tibshirani, R., "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, Vol. 58, No. 1, 1996, pp. 267–288.

[34] Dixon, W. E., Behal, A., Dawson, D. M., and Nagarkatti, S. P., *Nonlinear Control of Engineering Systems: A Lyapunov-Based Approach*, Control Engineering, Springer Science+Business Media, 2003. https://doi.org/10.1007/978-1-4612-0031-4, URL https://link.springer.com/book/10.1007/978-1-4612-0031-4.