Spring 2022

Theoretical and experimental application of neural networks in spaceflight control systems

Pavel Galchenko

Department: Mechanical and Aerospace Engineering

THEORETICAL AND EXPERIMENTAL APPLICATION OF NEURAL NETWORKS IN SPACEFLIGHT CONTROL SYSTEMS

by

PAVEL GALCHENKO

A DISSERTATION

Presented to the Graduate Faculty of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

DOCTOR OF PHILOSOPHY

in

AEROSPACE ENGINEERING

2022

Approved by:

Henry Pernicka, Advisor
Jagannathan Sarangapani
Carl Seubert
Thomas Vojta
Serhat Hosder

PUBLICATION DISSERTATION OPTION

This dissertation consists of the following three articles, formatted in the style used by the Missouri University of Science and Technology.

Paper I: Pages 7-48 have been submitted to the Journal of Guidance, Control and Dynamics.

Paper II: Pages 49-87 are intended for submission to the Journal of Aircraft.

Paper III: Pages 88-130 are intended for submission to the Journal of Spacecraft and Rockets.

ABSTRACT

Spaceflight systems can enable advanced mission concepts that can help expand our understanding of the universe. To achieve the objectives of these missions, spaceflight systems typically leverage guidance and control systems to maintain some desired path and/or orientation of their scientific instrumentation. A deep understanding of the natural dynamics of the environment in which these spaceflight systems operate is required to design control systems capable of achieving the desired scientific objectives. However, mitigating strategies are critically important when these dynamics are unknown or poorly understood and/or modelled. This research introduces two neural network methodologies to control the translation and rotation dynamics of spaceflight systems. The first method uses a neural network to perform nonlinear estimation in the control space for both translational and attitude control. The second method uses an observer with a neural network to perform estimation outside the control space, and input-output feedback linearization using the estimated dynamics for both translational and attitude control. The methods are demonstrated for attitude control through simulation and hardware testing on the Wallops Arc-Second Pointer, a high-altitude balloon-borne spaceflight system. Results show that the two new methodologies can provide improved attitude control performance over the heritage control system. The methods are also demonstrated for translational and attitude control of two small spacecraft in a deep space environment, where they provide improved position and attitude control performance as compared to a traditional control method. This work demonstrates, through simulation and hardware testing, that the two neural network methods presented can offer improved translational and attitude control performance of spaceflight systems where the dynamic environment may be unknown or poorly understood and/or modeled.

ACKNOWLEDGMENTS

First and foremost, I would like to acknowledge and thank my advisor, Dr. Henry Pernicka, for your mentorship and support throughout my undergraduate and graduate education. Your passion, dedication, and expertise in orbital dynamics inspired me to pursue this career in spaceflight. Not only have you provided me with an invaluable education, your personal kindness and support to my family will always be fondly remembered.

I would also like to thank the other members of my committee for their instruction and contributions. To Dr. Jagannathan Sarangapani, your expertise on nonlinear systems, control, and neural networks have guided my work beyond measure. To Dr. Carl Seubert, your mentorship at NASA Jet Propulsion Laboratory introduced me to the wonderful world of guidance and control and set me on this research path, for which I am forever grateful. To Drs. Thomas Vojta and Serhat Hosder, your guidance on computational methods and nonlinear systems have been an incredible asset. Finally, I would also like to thank Dr. Sivasubramanya Balakrishnan, who sadly passed during the course of my graduate education. You've always believed in me and I am eternally grateful for your mentorship, instruction, and kindness. You are sorely missed.

I am thankful to the NASA Pathways program, where my experience at Wallops Flight Facility and Goddard Space Flight Center greatly accelerated my education and understanding of spaceflight systems. I am especially thankful to Jim Lanzi, Scott Heatwole, and Zach Peterson for their mentorship, support, and contributions to applying this research to NASA spaceflight systems. This work would be incomplete without you.

Special thanks to my dearest friends, Yezad Anklesaria, Jill Davis, Donna Jennings, and Chris Yoder, with whom I've taken this incredible journey. Your friendship made the journey so much more meaningful. May our paths always cross.

And finally, I am most thankful to my wife, Iuliia. Thank you for your never ending support. You are truly my best and most cherished friend.

TABLE OF CONTENTS

Page

SECTION

PAPER

SECTION

LIST OF ILLUSTRATIONS

Figure Page

PAPER I

PAPER II

PAPER III

SECTION

1. INTRODUCTION

> The missile knows were it is at all times. It knows this because it knows where it isn't. By subtracting where it is from it isn't, or where it isn't from where it is (whichever is greater), it attains a difference or deviation. The guidance system uses deviations to generate corrective commands to drive the missile from a position where it is to a position where it isn't, arriving at a position where it wasn't, but is now. [. . .] Simple.
>
> *Colonel (Ret) George Gril, Association of Air Force Missileers* [1]

One of the overarching goals of spaceflight systems is the expansion of human knowledge through scientific exploration of the universe. To achieve the desired scientific objectives, spaceflight systems typically leverage guidance and control systems to maintain some desired path and/or orientation of their scientific instrumentation. When designing a control system for spaceflight systems, two general reference systems are considered. The first reference system is trajectory control, which can range from stationkeeping a spacecraft to a desired orbit or that of complex formation control of multiple spacecraft, forming large virtual structures. In essence, the desired objective is to maintain the system at some desired position within some frame of reference. The second reference system consists of maintaining the relative spatial orientation of a body to achieve some desired objective. This could be as simple as pointing a spacecraft's scientific instrument at some distant astronomical object, or as complex as pointing a telescope at the Sun from a dynamic platform the system does not control. These two categories can be described by translational and rotational dynamics. The focus of this research is to synthesize and implement a novel approach to the translation and attitude control problem for spaceflight systems; in other words, "how to take a complex spaceflight system in a complex dynamic environment from a state that it isn't, apply some corrective command, and arrive at a state that it wasn't. Simple."

1.1. MOTIVATION

Designing control algorithms for spaceflight systems is not new and incredible feats of engineering and scientific exploration have been achieved over the decades. However, because many of these control systems rely on having a deep understanding of the natural dynamics of the environment in which they operate, mitigating strategies are critically important when these dynamics are unknown or poorly understood and/or modelled. While there exists a large body of work on robust control methods that can assure some level of stability, one emerging field of study is the application of neural networks to nonlinear control systems. Neural networks, originally modeling the functions of neurons in the human brain, can be used for universal function approximation. While a large portion of this field focuses on offline learning and application to identification systems, a subset of the research has been applied to online learning in control systems. By learning online, the neural network can begin to “learn” unknown dynamics or those that were poorly understood and/or modeled. A literature review has shown that, to date, neural network control systems have yet to be applied to the control of translational and rotational dynamics of on-orbit spaceflight systems, though they have shown success in theoretical analysis and experimentally in other fields such as terrestrial air and land vehicles. This research seeks to synthesize and apply neural network control methods to spaceflight systems and to demonstrate their feasibility not only through simulation, but also through software-in-the-loop (SWIL) and hardware-in-the-loop (HWIL) testing on physical hardware for spaceflight missions.

1.2. NEURAL NETWORK OVERVIEW

The utilization of neural networks for open- and closed-loop systems stems from the original mathematical models developed in the early 1940s to describe the human nervous system [2, 3]. Over the next few decades, these models evolved and interest in artificial neural networks began to grow. Specifically, it was realized that the adaptability and learning capabilities could be applied to problems that would otherwise require significant computation. Early work focused on open-loop systems such as speech and image recognition. As early work showed promise with these applications, interest quickly grew and neural networks are now extensively used for classification, pattern recognition, and function approximation. The application to closed-loop systems such a feedback control systems started in the 1980s and research in this field continues to quickly expand as the adaptive and learning capabilities of neural networks greatly benefit the control of complex nonlinear systems, such as those used in aerospace applications [3].

The mathematics of artificial neural network systems stem from the mathematical model of a single neuron, which is graphically represented in Figure 1.1. The neuron has a series of branching dendrites that bring a variety of signals to the neurons body, the soma. The signals are acted upon and gathered by the soma and once the total signal exceeds some threshold, the signal is transmitted via the axon. An artificial neural network models this biological behavior for a multitude of neurons in a unified network. To paraphrase, a series of inputs (the basis vector) are acted upon by weighting coefficients (the weight matrices) and filtered through an activation function within hidden layers such that a specific output is achieved, as shown in Figure 1.2. For this work, the outputs of the artificial neural network are the estimated dynamics, $F(\mathbf{x})$, within which a spaceflight system may operate. The goal of artificial neural networks then is to learn how to change the weighting coefficients such that the output from the network, based on a basis vector defined by a set of input signals, matches some desired goal. This process is typically termed as "training" a neural network and typically requires a predefined input and output. A large body of literature exists on these techniques for open-loop systems for classification, pattern recognition, and function approximation [2, 4, 5].

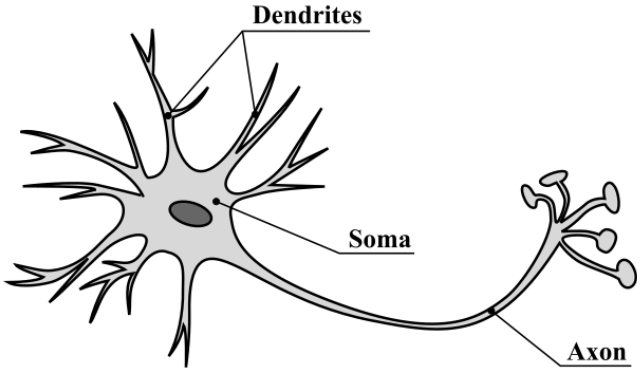


Figure 1.1. A single neuron acquires input signals from the dendrites and outputs data through the axon.

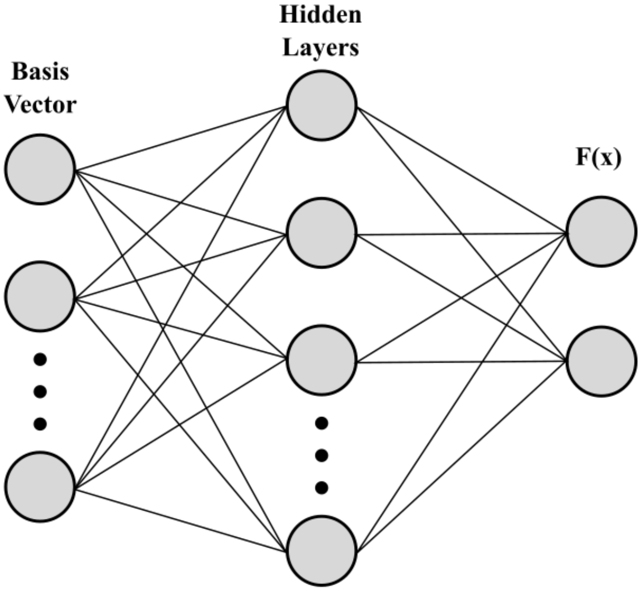


Figure 1.2. An artificial neural network can be represented by a basis vector of input signals being acted on by weighting coefficients and activation functions within hidden layers before an output is generated.

The application of artificial networks to control systems brings several additional complications. Many training techniques used for open-loop systems require training through backpropagation, requiring the final output to be known. However, for a closed-loop control system, not only is the desired output potentially unknown, the system must operate in real time such that instability in the systems behavior is not introduced, which could lead to potential critical hardware failures. For example, assume that the disturbance dynamics of a spaceflight system are unknown. The neural network must be able to learn these dynamics in real time, without ever knowing the true dynamics *apriori*, while keeping the spaceflight system closed-loop dynamics stability. Work by Lewis and Jagannathan brought a significant level of maturity to the application of neural networks to closed-loop control systems, showing generalized techniques and mathematical stability proofs guaranteeing stability [3]. This work expands on some of these methods, specifically providing practical applications to spaceflight systems.

1.3. LITERATURE REVIEW

An in-depth literature review is included in each of the papers presented.

1.4. CONTRIBUTIONS

The following objectives and contributions are presented in this work.

1. *Developed the theoretical framework for the application of neural network control methods to both translational and rotational dynamics in spaceflight systems.* This work focuses on the mathematical formulation of two neural network control methods and how they can be applied to both translational and rotational dynamics. The mathematics for the two formulations are presented in Papers I, II, and III.

2. *Implemented neural network control methods for both translational and rotational spaceflight systems to demonstrate the viability of the methodology while characterizing the performance.* This work focuses on the numerical simulation of two neural network control methods when applied to various spaceflight systems to model the performance in translational and rotational control. The simulation environment allows for adding and modifying the system dynamics, enabling the observation of how the system behaves in various conditions. Paper I presents the implementation for attitude dynamics while Paper III presents the implementation for both translation and attitude dynamics.

3. *Experimentally tested neural network control methods on physical hardware and compared performance to a traditional control method.* This work applies the two neural network control methods to the rotational dynamics of a physical spaceflight system, where hardware-in-the-loop testing was completed to verify the functionality and benefits of the proposed methods. Results show that the two new methodologies provide improved pointing performance over the heritage control system.

1.5. ORGANIZATION

Section 1 presents the motivation and contributions of this work.

Three papers are presented on the design, implementation, and testing of two neural network control methodologies for spaceflight systems. Paper I presents the theoretical framework for the two neural network methods and demonstrates through simulation the viability of the two methods for attitude control of the Wallops Arc-Second Pointer (WASP), a balloon-borne spaceflight system. Paper II presents the implementation, tuning, and testing of the two neural network methods for attitude control of the WASP system, demonstrating that the novel methods provide improved pointing performance through software and hardware testing in a flight-like environment. Paper III further expands the two neural network methodologies and applies them to not only the attitude dynamics of a spaceflight system, but also to that of translational control. The paper shows through simulation that the methods can be adapted easily to formation control of small spacecraft in a deep space environment, achieving desired objectives for both translation and attitude control performance.

Section 2 presents future work and several potential NASA missions where these methodologies may soon be used.

Section 3 summarizes the work and makes concluding remarks.

PAPER

I. NEURAL NETWORK ATTITUDE CONTROL SYSTEM DESIGN FOR THE WALLOPS ARC-SECOND POINTER

Pavel Galchenko[1] and Henry Pernicka[2]
Missouri University of Science and Technology, Rolla, MO, 65409-0050

ABSTRACT

This research introduces two neural-network-based methodologies to control the pointing performance and stability of the Wallops Arc-Second Pointer, a pointing system suspended from a high altitude balloon platform. The first method uses filtered tracking error and a two-layer neural network to perform nonlinear estimation in the control space for the control system. The second method uses an observer and a one-layer neural network to perform nonlinear estimation outside the controller, and input-output feedback linearization for the control system. Lyapunov proofs are provided to show ultimately upper bounded stability and software-in-the-loop testing was conducted to test the control systems against the heritage proportional-integral-derivative control system. Results show that the two new methodologies provide improved pointing performance in all tested cases.

[1] Ph.D. Candidate, Department of Mechanical and Aerospace Engineering, Missouri University of Science and Technology, 400 W 13th St, Rolla, MO 65409-0050.

[2] Curators' Distinguished Teaching Professor of Aerospace Engineering, Department of Mechanical and Aerospace Engineering, Missouri University of Science and Technology, 400 W 13th St, Rolla, MO 65409-0050.

NOMENCLATURE

$\mathbf{x}_1 = \left[\ \theta_\alpha \quad \theta_\beta \quad \theta_\gamma\ \right]^{\mathrm{T}}$	=	roll, pitch, and yaw rotations [rad]
$\mathbf{x}_2 = \left[\ \omega_\alpha \quad \omega_\beta \quad \omega_\gamma\ \right]^{\mathrm{T}}$	=	roll, pitch, and yaw angular velocities [rad/s]
$\mathbf{u} = [\ u_\alpha \quad u_\beta \quad u_\gamma\ ]^{\mathrm{T}}$	=	roll, pitch, and yaw control torques [ft-lb]
$\mathbf{f}_{\mathbf{x}}(\mathbf{x}_2)$	=	known system dynamics
$\mathbf{g}_{\mathbf{x}}$	=	control mapping to dynamics
$\boldsymbol{\delta}(\mathbf{x})$	=	known disturbance dynamics
J	=	system inertia matrix [slug-ft^2]
$\mathbf{x}_d = [\ \mathbf{x}_{d1} \quad \mathbf{x}_{d2}\ ]^T$	=	desired orientation [rad] and angular rates [rad/s]
$\mathbf{f}_d$	=	desired dynamics
$\mathbf{e} = [\ \mathbf{e}_1 \quad \mathbf{e}_2\ ]^T$	=	error states
A, B	=	known linear dynamics matrices
K	=	linear control gain matrix
A_{CL}	=	closed-loop dynamics
K_p	=	proportional [ft-lb/rad] gain
K_i	=	integral [ft-lb/(rad-sec)] gain
K_d	=	derivative [ft-lb/(rad/sec)] gains
ω_n	=	controller natural frequency [rad/s]
ζ	=	damping ratio
α	=	tuning parameter
$\mathcal{F}_{\mathbf{x}}, \hat{\mathcal{F}}_{\mathbf{x}}$	=	true and estimated dynamics
$\boldsymbol{\Delta}(\mathbf{x})$	=	unmodeled perturbations/dynamics
σ, σ'	=	activation function and its derivative
$\phi\,(\mathbf{z})$	=	basis vector
H_β, H_γ	=	pitch/yaw angles from gimbal hubs (rad)
ε	=	bounded neural network estimation error

$W^T, V^T, \hat{W}^T, \hat{V}^T$	=	true and estimated neural network weighting matrices
$\tilde{W}, \tilde{V}$	=	neural network weight matrices estimation error
$\mathbf{r}$	=	filtered tracking error
λ	=	gain coefficient
K_v, K_z	=	controller gain matrices
$\mathbf{u}_r$	=	robustifying control
$\tilde{\mathcal{F}}_{\mathbf{x}}$	=	dynamics estimation error
M, N	=	neural network adaptation gain matrices
κ	=	design parameter
L	=	Lyapunov function
$tr\{\}$	=	trace operator
K_2	=	linear observer gain matrix
$\mathbf{e}_a$	=	observer estimation error
Γ	=	observer neural network adaptation rate
B_w	=	system bandwidth (Hz)

1. INTRODUCTION

The National Aeronautics and Space Administration's (NASA) Balloon Program Office (BPO) provides scientists with a High Altitude Balloon (HAB) platform to conduct novel science in the Earth's upper atmosphere [1, 2]. At a float altitude of 120,000 ft, the HAB platform places experiments above 99.5% of the atmosphere (by mass), offering scientists a space-like environment at significant reductions in cost compared to typical spaceflight missions. Many science missions require the ability to accurately point the scientific instruments at targets of interest to perform observations and achieve objectives [3]. The ability to point science instruments with arcsecond accuracy and stability is needed by multiple science disciplines such as Planetary, Heliophysics, Astrophysics, and Earth

Sciences. The Wallops Arc-Second Pointer (WASP) is one such unique pointing control system developed by NASA's Wallops Flight Facility (WFF), part of the Goddard Space Flight Center (GSFC). The WASP system provides a platform to point large telescopes and scientific instruments with sub-arcsecond accuracy and stability while suspended from a HAB [4, 5].

The WASP system is built around an external gondola structure, which is suspended from a rotator gimbal attached to the HAB. The rotator provides initial targeting and coarse azimuth tracking and stabilization for the external gondola. The outer gimbal frame is fixed to the external gondola structure and is connected to the inner gimbal frame with a pitch motor and encoder hub pair to provide elevation control. The inner gimbal frame connects to the center body structure through a yaw motor and encoder hub pair and provides fine azimuth control. This design is similar to a gimbaled inertial platform, with pitch-yaw articulated axes. To minimize cross-coupling terms in the dynamics, the rotator provides coarse azimuth pointing of the external gondola such that the center body is within 1 degree of the inner gimbal frame, minimizing the disturbances seen in the fine yaw control channel. The motor hubs include large diameter brushless direct-current (DC) torque motors to drive the system and utilize rotating shafts to eliminate static friction in the bearings. The shafts are counterrotated in the hub pairs to minimize residual kinetic friction. The attitude is determined by integrating angular rates provided by a Northrop Grumman LN251, a fiber-optic gyro-based inertial measurement unit. Absolute pointing information is provided from a custom star tracker camera developed at WFF and a six-state extended Kalman filter is used to provide quaternion state data to the control system. A modified proportional-integral-derivate (PID) control law is used to compute control torques to the pitch/yaw motor hubs to provide sub-arcsecond pointing for the center body.

The WASP platform was successfully demonstrated with a test flight in 2011 followed by a second test flight in 2012. Science operations began in 2013 with the Hyper-Spectral Imager for Climate Science (HySICS) mission. Two additional science flights were conducted in 2014, one being HySICS II [6] and the second flight being the Observatory for Planetary Investigations from the Stratosphere (OPIS) mission [7]. The five flights completed the development phase of the WASP platform and showed the capability of the WASP platform. Since then, WASP has conducted an additional four science missions, including two flights of the X-Calibur mission and the latest two science flights in 2019 for the Balloon-borne Investigation of Temperature and Speed of Electrons in the corona (BITSE) mission and the Planetary Imaging Concept Testbed Using a Recoverable Experiment - Coronagraph (Picture-C) mission [8–11].

1.1. LITERATURE REVIEW

In 1968, the STRATOSCOPE II mission set the first precedent for sub-arcsecond pointing from a HAB observatory [3]. While the mission did not directly measure its pointing performance, post-flight analysis of the various onboard systems showed that the telescope was able to achieve 0.02 arcsecond pointing accuracy. The system used two pointing control systems. This first achieved coarse pointing to an accuracy of 15 arcseconds using an active control loop. A secondary pointing system positionally adjusted a transfer lens from ground commands until the image was fixed in the photographic plane, leading to the final 0.02 arcsecond pointing accuracy [12]. It should also be noted that the coarse pointing could only occur when not imaging, as jitter and disturbances from coarse control prevented the gathering of science data. While this precedent was set in the late 1960s, it wasn't until the early 2000s that arcsecond and sub-arcsecond pointing would be again attempted and achieved from a HAB observatory (several of which are discussed here).

The Balloon-borne Large Aperture Submillimeter Telescope (BLAST) was a suborbital surveying experiment to study star formation in local galaxies, conducting one test flight and two science mission flights from 2003 through 2006 [13]. The pointing system on the gondola was able to provide 30 arcsecond pointing in flight, while post-processing was able to update the pointing reconstruction to within 5 arcsecond RMS. Another mission that brought pointing into the sub-arcsecond region was the SUNRISE mission flown in 2009 [14]. The primary control system was designed to point the telescope within 7.5 arcsecond RMS, while a secondary fine optics based system refined the pointing to within 0.05 arcseconds. During flight operations, wind gusts and external disturbances significantly limited the performance and the longest period in which primary pointing was within the allowed maximum ± 46 arcsecond range was only 45 minutes. Post-processing of the data showed final secondary pointing accuracy and stability in the 0.1 arcsecond range. In 2015 a test flight of the Balloon-Borne Imaging Testbed (BIT) was conducted, which sought to bring primary pointing into the 1-2 arcsecond RMS and proposed secondary systems refining the pointing into the sub-arcsecond regime. The test flight was successful and achieved 0.68 arcsecond RMS over periods of integration between 10 to 30 minutes [15]. A follow-on to the mission was proposed and hardware testing in a laboratory was conducted for the addition of a secondary optical control system that would bring the pointing to sub 0.1 arcsecond RMS [16]. The Balloon-borne Imaging Testbed, Sub- arcsecond Telescope And BaLloon Experiment (BIT-STABLE) uses the BIT platform to bring the instrumentation within 2 arcsecond RMS and the secondary system, STABLE, was able to demonstrate pointing stability within the 0.1 arcsecond threshold [3]. While the system was demonstrated successfully in the laboratory, the funding to perform in-flight testing was not available.

Over the ten-plus years of operations of the WASP platform, the pointing performance has been consistently demonstrated to be sub-arcsecond with the latest flights from 2019 demonstrating pointing performance down to 0.2 arcseconds RMS over long integration time periods for two separate missions. While significant improvements in the pointing performance would likely require hardware modifications (such as the addition of reaction wheels or improved sensing), it is still desirable to determine if pointing performance could be improved through software modifications. For example, during the 2019 flight campaign the WASP system would occasionally see brief excursions exceeding 0.5 arcseconds. While the excursion was quickly captured and brought back towards the average RMS, these excursions could decrease the performance of certain science instruments. This work suggests to augment the WASP platform with the addition of a neural network control system, which can be activated in lieu of the traditional PID system.

1.2. SCHOLARLY CONTRIBUTIONS

This research offers several contributions to the nonlinear control field for gimbaled pointing systems:

1. SNNARC, a two-layer neural network controller, is formulated for an inertially gimbaled control system suspended from a high altitude balloon platform. Simulation shows improved pointing performance is achieved as compared to the heritage PID control system.
2. MSO, a one-layer neural network observer and an input-output feedback linearization controller, is formulated for an inertially gimbaled control system suspended from a high altitude balloon platform. Simulation shows improved pointing performance is achieved as compared to the heritage PID control system.

1.3. ORGANIZATION

This work is organized as follows. In Section 2, a brief background on the dynamics and control theory is presented, as well as a description of the heritage controller design implemented for the WASP system. Section 3 introduces neural networks and the design of two control methodologies utilizing neural networks. Section 4 discusses the simulation environment and the configuration of the three control methodologies. Section 5 presents software-in-the-loop simulation results. Finally, Section 6 gives an overview of the results and presents future work.

2. BACKGROUND

By defining θ_α, θ_β, θ_γ as rotations, corresponding to roll, pitch, yaw, respectively, about a primary orthogonal axes and ω_α, ω_β, ω_γ as the angular velocities about these respective axes, a typical unconstrained second-order three degrees of freedom attitude dynamics model [17, 18] can be given by a two-vector state system as

$$\begin{aligned} \dot{\mathbf{x}}_1 &= B(\theta_\alpha, \theta_\gamma)\mathbf{x}_2 \\ \dot{\mathbf{x}}_2 &= \mathbf{f}_\mathbf{x}(\mathbf{x}_2) + \boldsymbol{\delta}(\mathbf{x}) + \mathbf{g}_\mathbf{x}\mathbf{u} \end{aligned} \tag{1}$$

where the parameterization matrix, $B(\theta_\alpha, \theta_\gamma)$, can be described by

$$B(\theta_\alpha, \theta_\gamma) = \begin{bmatrix} 1 & -\cos\theta_\alpha \tan\theta_\gamma & \sin\theta_\alpha \tan\theta_\gamma \\ 0 & \cos\theta_\alpha \sec\theta_\gamma & -\sin\theta_\alpha \sec\theta_\gamma \\ 0 & \sin\theta_\alpha & \cos\theta_\gamma \end{bmatrix} \tag{2}$$

and where $\mathbf{f}_\mathbf{x}(\mathbf{x}_2) = -J^{-1}[\mathbf{x}_2 \times]J\mathbf{x}_2$, $\mathbf{x}_1 = \begin{bmatrix} \theta_\alpha & \theta_\beta & \theta_\gamma \end{bmatrix}^\mathrm{T}$, $\mathbf{x}_2 = \begin{bmatrix} \omega_\alpha & \omega_\beta & \omega_\gamma \end{bmatrix}^\mathrm{T}$, $\mathbf{u} = [\, u_\alpha \quad u_\beta \quad u_\gamma \,]^\mathrm{T}$, and where J is the inertia matrix (assumed constant) of the (rigid) body with respect to the center of mass and in terms of the body-fixed axes, $\mathbf{x}_1$ and $\mathbf{x}_2$ are the rotation and angular velocity vectors, respectively, of the body with respect to the inertial frame, $\delta(\mathbf{x})$ is some known disturbance dynamics, $\mathbf{g}_\mathbf{x}$ maps the controller dynamics to the system, and $\mathbf{u}$ is the applied torque, i.e. the control input. It should be noted that J^{-1} is also embedded into the disturbance dynamics $\delta(\mathbf{x})$ and the mapping of the controller dynamics, $\mathbf{g}_\mathbf{x}$, to simplify the formulation.

A variety of models exist to describe the dynamics of two-axis gimbal systems [15, 19–21], but these can all be simplified to the form given in Eq. (1) with variations in the definitions of the attitude parameterization, $B(\theta_\alpha, \theta_\gamma)$, in Eq. (2), as well as definitions of $\mathbf{f}_\mathbf{x}(\mathbf{x}_2)$ and $\delta(\mathbf{x})$. The nature of the mechanical design of WASP leads to reductions in cross-coupling terms in the gimbal dynamics, allowing for several model modifications.

First, the center bodies roll angle is fixed and null in the gimbal frame, such that $\theta_\alpha = 0$. As such, the parameterization matrix, $B(\theta_\alpha, \theta_\gamma)$, can now be described by

$$B(\theta_\alpha, \theta_\gamma) = \begin{bmatrix} 1 & -\tan\theta_\gamma & 0 \\ 0 & \sec\theta_\gamma & 0 \\ 0 & 0 & \cos\theta_\gamma \end{bmatrix} \tag{3}$$

where the evolution of the rotations, $\dot{\mathbf{x}}_1$, are now only dependent on the yaw state, θ_γ, and the angular velocity vector, $\mathbf{x}_2$. The mechanical design of the WASP platform with an integrated rotator is designed such that the yaw rotation of the centerbody within the gimbal frame is always small. In fact, during pointing operations, this angle is typically less that 1.0 degrees. This allows for a small angle approximation, whereby the parameterization matrix, $B(\theta_\alpha, \theta_\gamma)$, becomes the identity matrix, $I_{3\times3}$. The WASP platform hardware configuration was specifically designed with this in mind, allowing of for the PID control system to treat the pitch, θ_β, and yaw, θ_γ, channels independently. Using the small angle approximation on the updated parameterization matrix in Eq. (3) and plugging that result into Eq. (1), the simplified dynamics are now given by

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \mathbf{x}_2 \\ \dot{\mathbf{x}}_2 &= \mathbf{f}_\mathbf{x}(\mathbf{x}_2) + \delta(\mathbf{x}) + \mathbf{g}_\mathbf{x}\mathbf{u} \end{aligned} \tag{4}$$

where the system dynamics are in Brunovsky canonical form [22].

Due to the large mass and volume of the WASP platform and the center body, only approximations of the model inertia values can be made. Fine balancing of the center body occurs during integration and is typically performed through the manual addition of ballast masses at various locations of the structure. As such, the use of a simplified model is further justified with the addition of integrator action in the control system to compensate for model inaccuracies and static imbalances. While significant effort has been presented for the formulation of the system dynamics, the new methodology introduced in this research makes no assumptions on the details of $\mathbf{f_x}(\mathbf{x}_2)$ and $\delta(\mathbf{x})$ and only assumes the dynamics can be described by the form given in Eq. (4).

2.1. ERROR DYNAMICS

In order to synthesize an accurate control system for the rotational system, it is useful to rewrite the dynamics in terms of the error between the current and desired reference system states. The desired states, $\mathbf{x}_d$, are defined as $\mathbf{x}_d = [\ \mathbf{x}_{d1} \quad \mathbf{x}_{d2}\ ]^T$, where $\mathbf{x}_{d1}$ is a vector of desired rotations and $\mathbf{x}_{d2}$ is a vector of desired angular rates. The reference system dynamics are given as $\dot{\mathbf{x}}_{d1} = \mathbf{x}_{d2}$ and $\dot{\mathbf{x}}_{d2} = \mathbf{f}_d$, where $\mathbf{f}_d$ can be a set of desired dynamics for the reference system. Note that for a regulatory reference system, $\mathbf{x}_{d2} = 0$ and $\mathbf{f}_d = 0$. The error states are now defined as $\mathbf{e} = [\ \mathbf{e}_1 \quad \mathbf{e}_2\ ]^T$, where $\mathbf{e}_1 = \mathbf{x}_1 - \mathbf{x}_{d1}$ and $\mathbf{e}_2 = \mathbf{x}_2 - \mathbf{x}_{d2}$, such that taking the derivative of the error vectors gives $\dot{\mathbf{e}} = [\ \dot{\mathbf{e}}_1 \quad \dot{\mathbf{e}}_2\ ]^T$ with

$$\begin{aligned}\dot{\mathbf{e}}_1 &= \mathbf{e}_2\\ \dot{\mathbf{e}}_2 &= \mathbf{f_x}(\mathbf{x}) - \mathbf{f}_d + \delta(\mathbf{x}) + \mathbf{g_x}\mathbf{u},\end{aligned} \tag{5}$$

defining the error dynamics used for the various controller formulations.

2.2. INPUT-OUTPUT FEEDBACK LINEARIZATION CONTROLLER DESIGN

It is desired that a controller be designed that will allow a nonlinear system to maintain its attitude about some reference. By first considering a linear system, $\dot{\mathbf{x}} = A\mathbf{x}+B\mathbf{u}$, where A and B are known constant matrices, $\mathbf{u}$ is the state feedback controller given by $\mathbf{u} = -K\mathbf{x}$, and K is some gain matrix, the closed-loop system dynamics are given as $\dot{\mathbf{x}} = (A - BK)\mathbf{x}$. The origin of the linear closed-loop system can be shown to be asymptotically stable if the matrix $(A - BK)$ is Hurwitz [23]. This is achieved by choosing the gain matrix K such that the eigenvalues of the linear closed-loop system lie in the left-half of the complex s-plane.

By using input-output feedback linearization [23] and carefully designing a nonlinear controller in terms of the error dynamics in Eq. (5), it is possible to make the nonlinear system behave in a similar manner. A controller is chosen such that

$$\mathbf{u} = \frac{1}{\mathbf{g_x}} \left(-\mathbf{f}_x(\mathbf{x}) + \mathbf{f}_d - \delta(\mathbf{x}) - \mathbf{k}_p \mathbf{e}_1 - \mathbf{k}_d \mathbf{e}_2 \right) \tag{6}$$

where $\mathbf{k}_p$ and $\mathbf{k}_d$ are proportional and derivative controller gain vectors, respectively, and $|\mathbf{g_x}| > 0$. The closed-loop error dynamics in Eq. (5) are reduced to $\dot{\mathbf{e}} = A_{CL}\ \mathbf{e}$, where $A_{CL} = \begin{bmatrix} 0 & 1 \\ -\mathbf{k}_p & -\mathbf{k}_d \end{bmatrix}$, such that the origin of the unperturbed nonlinear system will be asymptotically stable if A_{CL} is Hurwitz. This method of nonlinear control is the basis from which the neural network controllers are designed.

2.3. PID CONTROLLER DESIGN

The control system for the WASP platform uses a traditional proportional-integral-derivative (PID) controller to achieve sub-arcsecond pointing accuracy and stability. The system is tuned as follows. First, it is assumed that the inertia of the system is diagonal with no cross products and that the dynamics of the second-order system can be written separately for the pitch/yaw channels. The system can then be described by the plant $P(s) = J_i^{-1}/s^2$, where J_i is the inertia of specified axis. By choosing a PID control input, the closed-loop characteristic equation becomes $s^3 + [J_i^{-1}K_d]s^2 + [J_i^{-1}K_p]s + [J_i^{-1}K_i]$ where K_p, K_i, and K_d are proportional, integrator, and derivative gains, respectively. Using the typical (ζ, ω_n) parametrization and fixing the third pole at $-\alpha\omega_n$, the desired closed-loop characteristic equation is given by $s^3 + [(2\zeta + \alpha)\omega_n]s^2 + [(2\zeta\alpha + 1)\omega_n^2]s + [\alpha\omega_n^3]$ where ω_n is the natural frequency of the system, ζ is the damping ratio, and α is a design parameter. The controller gains can now be selected to match the design characteristic equation by

$$K_p = \frac{(2\zeta\alpha + 1)\omega_n^2}{J_i^{-1}} \qquad K_i = \frac{\alpha\omega_n^3}{J_i^{-1}} \qquad K_d = \frac{(2\zeta + \alpha)\omega_n}{J_i^{-1}} \tag{7}$$

where by fixing the design parameter at $\alpha = 1$, the values for ζ and ω_n are tuned to give desired performance characteristics. It should be noted that for $\alpha = 0$, the gain tuning becomes that of a proportional-derivative (PD) controller for the same system.

3. NEURAL NETWORK DESIGN

It has been shown that with the formulation of the dynamics and the input-output feedback linearization controller that the rotational systems error dynamics have guaranteed asymptotic stability. However, two potentially limiting assumptions were made that must be addressed before implementation with the WASP system can be considered. It was assumed that the model of $\mathbf{f}_x(\mathbf{x})$ is "truth" and that $\delta(\mathbf{x})$ describes all other contributing disturbance dynamics to the system. This is of course impossible; all models have assumptions and approximations, and while one can predict a number of disturbances (which are approximations as well), it is impossible to perfectly predict and model all effects.

Controlling nonlinear systems with unmodeled dynamics is an extensively studied field and there exists a number of methodologies with which these issues have been addressed in the past. These include assumptions of linearity of the system about specific operating points [24, 25], the addition of an error integral to the control system [26], utilizing sliding mode controllers [26, 27] and Lyapunov redesign [28], as well as the addition of neural networks [28–31]. This research focuses on the application of neural networks as they offer some unique advantages over other adaptive control methods.

Assume that the true dynamics are given as $\mathcal{F}_{\mathbf{x}} = \mathbf{f}_x(\mathbf{x}) + \delta(\mathbf{x}) + \boldsymbol{\Delta}(\mathbf{x})$, where $\mathbf{f}_x(\mathbf{x})$ and $\delta(\mathbf{x})$ are as described in Eq. (4), and $\boldsymbol{\Delta}(\mathbf{x})$ is a set of all other unmodeled dynamics. Using neural networks, estimating the true dynamics can be attempted using $\hat{\mathcal{F}}_{\mathbf{x}}$. A one-layer neural network is given by

$$\hat{\mathcal{F}}_{\mathbf{x}} = \hat{W}^T \phi(\mathbf{z}) \qquad \text{while} \qquad \mathcal{F}_{\mathbf{x}} = W^T \phi(\mathbf{z}) + \varepsilon \tag{8}$$

and a two-layer neural network is given as

$$\hat{\mathcal{F}}_{\mathbf{x}} = \hat{W}^T \sigma\left(\hat{V}^T \phi(\mathbf{z})\right) \qquad \text{while} \qquad \mathcal{F}_{\mathbf{x}} = W^T \sigma\left(V^T \phi(\mathbf{z})\right) + \varepsilon \tag{9}$$

where σ is some activation function, $\phi(\mathbf{z})$ is the basis vector, W^T and V^T are the true weights that give the value of $\mathcal{F}_{\mathbf{x}}$ with some bounded approximation error, ε, through the universal function approximation properties of neural networks [22], and $\hat{W}^T$ and $\hat{V}^T$ are the approximated weights. It should also be noted that the while the basis vector, $\phi(\mathbf{z})$, is typically composed of the state vector, $\mathbf{x}$, it may have additional elements and/or formulations that are not part of the state but correspond to the dynamics of the system.

The two-vector state systems from Eq. (4) can now be rewritten as

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \mathbf{x}_2 \\ \dot{\mathbf{x}}_2 &= \mathcal{F}_{\mathbf{x}} + \mathbf{g}_{\mathbf{x}}\mathbf{u} \end{aligned} \tag{10}$$

and the error dynamics in Eq. (5) as

$$\begin{aligned} \dot{\mathbf{e}}_1 &= \mathbf{e}_2 \\ \dot{\mathbf{e}}_2 &= \mathcal{F}_{\mathbf{x}} - \mathbf{f}_d + \mathbf{g}_{\mathbf{x}}\mathbf{u} \end{aligned} \tag{11}$$

forming the foundation of the two neural network methodologies studied in this research.

The first method operates in terms of the error dynamics, Eq. (11), and as such the neural network learning occurs in the control space. The second method introduces an observer for the state dynamics, Eq. (10), and as such the neural network learning occurs outside the control space. Both methods can be described by a Lyapunov candidate function and are shown in this study to be Ultimately Upper Bounded (UUB), demonstrating Lyapunov stability.

3.1. SNNARC FORMULATION

The first method, originally developed for robotic manipulators [22], is the Subarcsecond Neural Network Attitude Reference Controller (SNNARC). Its formulation here is given generically for a two-layer neural network, Eq. (9), and can be applied to any set of dynamics that can be written in Brunovsky canonical form. The error dynamics from Eq. (11) are written as a filtered tracking error where

$$\begin{aligned} \mathbf{r} &= \Lambda\mathbf{e} \\ \Lambda &= [\,\lambda \quad 1\,] \end{aligned} \quad \longrightarrow \quad \begin{aligned} \mathbf{r} &= \mathbf{e}_2 + \lambda\mathbf{e}_1 \\ \dot{\mathbf{r}} &= \mathcal{F}_{\mathbf{x}} - \mathbf{f}_d + \lambda\mathbf{e}_2 + \mathbf{g}_{\mathbf{x}}\mathbf{u} \end{aligned} \tag{12}$$

where λ is some positive gain coefficient and $\mathbf{r}$ is now a vector of scalar errors for each control axis. It is now possible to use the estimated dynamics, $\hat{\mathcal{F}}_{\mathbf{x}}$, in the controller design such that

$$\mathbf{u} = \frac{1}{\mathbf{g}_{\mathbf{x}}}\left(-\hat{\mathcal{F}}_{\mathbf{x}} + \mathbf{f}_d - K_v\mathbf{r} - \lambda\mathbf{e}_2 + \mathbf{u}_r\right) \quad \text{where} \quad \mathbf{u}_r = -K_z\left(\left\|\hat{\Theta}\right\| + \Theta_m\right)\mathbf{r} \tag{13}$$

where K_v and K_z are some positive controller gain matrices, $|\mathbf{g}_{\mathbf{x}}| > 0$, and the weighting matrices are defined as

$$\Theta = \begin{bmatrix} V & 0 \\ 0 & W \end{bmatrix} \quad \text{and} \quad \hat{\Theta} = \begin{bmatrix} \hat{V} & 0 \\ 0 & \hat{W} \end{bmatrix} \quad \text{and} \quad \|\Theta\| \leq \Theta_m$$

where Θ_m is a user-defined bound on the norm of the weighting matrices. The closed-loop system becomes

$$\dot{\mathbf{r}} = -K_v\mathbf{r} + \tilde{\mathcal{F}}_{\mathbf{x}} + \mathbf{u}_r \tag{14}$$

where $\tilde{\mathcal{F}}_{\mathbf{x}}$ is the estimation error defined by $\tilde{\mathcal{F}}_{\mathbf{x}} = \mathcal{F}_{\mathbf{x}} - \hat{\mathcal{F}}_{\mathbf{x}}$. Neural network weight update laws are selected as

$$\begin{aligned} \dot{\hat{W}} &= M\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\phi\left(\mathbf{z}\right)\right)\mathbf{r}^T - \kappa\left\|\mathbf{r}\right\|M\hat{W} \\ \dot{\hat{V}} &= N\phi\left(\mathbf{z}\right)\mathbf{r}^T\hat{W}^T\hat{\sigma}' - \kappa\left\|\mathbf{r}\right\|N\hat{V} \end{aligned} \tag{15}$$

where M and N are positive definite matrices, $\hat{\sigma} = \sigma\left(\hat{V}^T\phi\left(\mathbf{z}\right)\right)$, $\hat{\sigma}' = \sigma'\left(\hat{V}^T\phi\left(\mathbf{z}\right)\right)$, and κ is a design parameter such that $\kappa > 0$.

To prove the stability of the system, the candidate Lyapunov function is selected as

$$L = \frac{1}{2}\mathbf{r}^T\mathbf{r} + \frac{1}{2}tr\left\{\tilde{W}^TM^{-1}\tilde{W}\right\} + \frac{1}{2}tr\left\{\tilde{V}^TN^{-1}\tilde{V}\right\} \tag{16}$$

where $tr\left\{\right\}$ is the trace operator. By taking the time derivative of the Lyapunov function in Eq. (16) and using the definitions of $\mathbf{r}$ and $\dot{\mathbf{r}}$ in Eq. (12) and Eq. (14), respectively, and the weight update law in Eq. (15), stability in a compact set about the origin can be shown [31]. The full proof is shown in the appendix, demonstrating that the Lyapunov function is UUB in both $\left\|\mathbf{r}\right\|$ and $\left\|\tilde{\Theta}\right\|$ and thus showing Lyapunov stability of the proposed SNNARC method. This completes the formulation of the first neural network control strategy.

3.2. MSO FORMULATION

The second method, originally developed for estimating unmodeled dynamics [32–34], utilizes a Modified State Observer (MSO) and an input-output feedback linearization controller. First the dynamics in Eq. (10) are reduced to a single vector system as the dynamics that are estimated occur in the angular velocity states. The system is rewritten as a one-vector state system as

$$\dot{\mathbf{x}} = \mathcal{F}_{\mathbf{x}} + \mathbf{g}_{\mathbf{x}}\mathbf{u} \tag{17}$$

where $\dot{\mathbf{x}} = \dot{\mathbf{x}}_2$, or the derivative of the angular rates, from the previous formulation. For this system, an observer can be designed as

$$\dot{\hat{\mathbf{x}}} = \hat{\mathcal{F}}_{\mathbf{x}} + \mathbf{g}_{\mathbf{x}}\mathbf{u} + K_2\left(\mathbf{x} - \hat{\mathbf{x}}\right) \tag{18}$$

where $\hat{\mathbf{x}}$ is the estimated state and K_2 is some gain matrix. As with with SNNARC, $\hat{\mathcal{F}}_{\mathbf{x}}$ can be approximated by a neural network. A single layer neural network, Eq. (8), has been used in the past to learn and estimate the nonlinear dynamics for the observer. The state, Eq. (17), and observer, Eq. (18), are rewritten in terms of observer error, $\mathbf{e}_a = \mathbf{x} - \hat{\mathbf{x}}$, such that the observer error dynamics are given as

$$\dot{\mathbf{e}}_a = -K_2\mathbf{e}_a + \tilde{W}^T\phi\left(\mathbf{z}\right) + \varepsilon \tag{19}$$

where $\phi\left(\mathbf{z}\right)$ is the basis vector and $\tilde{W}$ is the error between the true weights and estimated weights such that $\tilde{W} = W - \hat{W}$. The neural network weight update law is given as

$$\dot{\hat{W}} = \Gamma\phi\left(\mathbf{z}\right)\mathbf{e}_a^T - \kappa\left\|\mathbf{e}_a\right\|\Gamma\hat{W} \tag{20}$$

where Γ is the adaptation rate and κ is the modification factor, which bound the weights and provides robustness. The stability of the system in Eq. (19) can be shown with the candidate Lyapunov function

$$L = \frac{1}{2}\mathbf{e}_a^T\mathbf{e}_a + \frac{1}{2}tr\left\{\tilde{W}^T\Gamma^{-1}\tilde{W}\right\} \tag{21}$$

where $tr\left\{\right\}$ is the trace operator. By taking the time derivative of the Lyapunov function in Eq. (21) and using the definitions of $\mathbf{e}_a$, $\dot{\mathbf{e}}_a$ in Eq. (19), and the weight update law in Eq. (20), stability in a compact set about the origin can be shown [31]. The proof is shown in the appendix, demonstrating that the Lyapunov function is UUB in both $\left\|\mathbf{e}_a\right\|$ and $\left\|\tilde{W}\right\|$ and thus shows Lyapunov stability of the observer in the proposed MSO method.

This formulation of the observer not only provides a state estimate, but also is able to estimate the nonlinear dynamics driving the system. It should also be noted that this formulation is independent of the controller design; estimation can occur with any controller paradigm. As such, it can be enabled before control is even applied to the system.

Once the nonlinear dynamics, $\hat{\mathcal{F}}_{\mathbf{x}}$, are estimated by the neural network, an input-output feedback linearization controller given by

$$\mathbf{u} = \frac{1}{\mathbf{g_x}}\left(-\hat{\mathcal{F}}_{\mathbf{x}} + \mathbf{f}_d - \mathbf{K}_p\mathbf{e}_1 - \mathbf{K}_d\mathbf{e}_2\right) \tag{22}$$

can be applied to the two-vector system in Eq. (11) such that the closed-loop system dynamics become

$$\begin{aligned} \dot{\mathbf{e}}_1 &= \mathbf{e}_2 \\ \dot{\mathbf{e}}_2 &= -\mathbf{K}_p\mathbf{e}_1 - \mathbf{K}_d\mathbf{e}_2 + \tilde{W}^T\phi\left(\mathbf{z}\right) + \varepsilon \end{aligned} \quad \longrightarrow \quad \dot{\mathbf{e}} = A_{CL}\,\mathbf{e} + d$$
$$\text{where} \quad A_{CL} = \begin{bmatrix} 0 & 1 \\ -\mathbf{K}_p & -\mathbf{K}_d \end{bmatrix} \quad \text{and} \quad d = \begin{bmatrix} 0 \\ \tilde{W}^T\phi\left(\mathbf{z}\right) + \varepsilon \end{bmatrix} \tag{23}$$

where d is bounded by $\|d\| \leq \|d_m\|$. By using the candidate Lyapunov function

$$L = \frac{1}{2}\mathbf{e}^T P\mathbf{e} \tag{24}$$

and taking its time derivative using the definition of $\dot{\mathbf{e}}$ in Eq. (23) and where P satisfies the equation $A^TP + PA = -Q$, stability in a compact set about the origin can be shown if $\|\mathbf{e}\| \geq 2\frac{\lambda_{max}(P)d_m}{\lambda_{min}(Q)}$. The proof is shown in the appendix, demonstrating that the Lyapunov functions in Eq. (21) and Eq. (24) are UUB in $\|\mathbf{e}\|$, $\|\mathbf{e}_a\|$, and $\left\|\tilde{W}\right\|$ and thus shows Lyapunov stability of the proposed MSO method. This completes the formulation of the second neural network control strategy.

4. WASP IMPLEMENTATION

The initial verification and validation of the pointing system was performed using high fidelity simulations through software-in-the-loop (SWIL) testing. The WASP system was simulated using NASA's high fidelity Portable Object Simulation (PortOSim) flight-control simulation framework. The framework is able to interface with the flight software and provides data to the software as if in flight. PortOSim uses JPL ephemeris models in conjunction with custom models for the individual sensors and actuators aboard WASP, which introduces noise to the sensor measurements and actuator commands. The SWIL simulation is performed through the ground support equipment (GSE) and software, creating an environment similar to flight. During flight operations, once the HAB system and WASP is released and begins ascending, ground operators are on standby until the system reaches a float altitude above 100,000 ft. Once WASP is at or near float, mission operations begin such as uncaging WASP, initiating an inertial hold, and finally target acquisition for science operations. In SWIL testing, the simulation begins at float conditions with the WASP system uncaged and in an inertial hold, leaving the user to begin mission operations starting at target acquisition. The system defaults to using the heritage PID control system for its initial inertial hold.

The SNNARC and MSO algorithms were written into two separate controller classes within the C++ flight software. The GSE allows the user to select the desired control system, allowing quick switching between the various controllers. Two cases were tested in the SWIL environment in preparation for the PICTURE-C mission. The simulation environment had the WASP platform track one of the inertial science targets, focusing on pointing performance and stability. A nominal case was run with the assumed WASP and telescope inertia values and disturbance environment anticipated for the mission. The inertia values were assumed to be $J_1 = 92$, $J_2 = 572$, and $J_3 = 554$ in slug-ft^2, corresponding to roll, pitch, and yaw axes, respectively. The second case applied various changes to the disturbance environment such as center-of-gravity (CG) misalignment and bearing friction noise. Each controller was tuned for sufficient performance in the nominal case and then tuning was held constant. A comparison between PID, SNNARC, and MSO performance and stability was performed and is presented.

4.1. PID SETUP

The heritage PID system was tuned until the desired performance for the Picture-C mission was achieved. It is desired that pointing remain sub-arcsecond and that the bandwidth of the controller is set sufficiently low so as to not interfere with the science optics control systems, which operate at high bandwidths. The WASP platform bandwidth is approximated as $B_w = \frac{\omega_n}{2\pi}$ where the bandwidth is given in the frequency space (H_z). By default, the typical bandwidth is set to $B_w = 0.4\ H_z$ and the damping ratio is set to $\zeta = 0.9$. The final gains for all the parameters are given as approximately $K_p = 14200$, $K_d = 5630$, and $K_i = 12700$.

4.2. SNNARC SETUP

The SNNARC algorithm was initialized in the flight software as follows. The inputs into the filtered tracking error, $\mathbf{r}$, are the pitch, $\theta_{e\beta}$, and yaw, $\theta_{e\gamma}$, pointing errors and the inertial angular velocities of those channels, ω_β and ω_γ, respectively. The basis vector, $\phi(\mathbf{z})$, was selected as $\phi(\mathbf{z}) = \begin{bmatrix}\theta_{e\beta} & \theta_{e\gamma} & \omega_\alpha & \omega_\beta & \omega_\gamma & H_\beta & H_\gamma\end{bmatrix}^T$ where α corresponds to the roll channel, β corresponds to the pitch channel, γ corresponds to the yaw channel, and $H_{\beta,\gamma}$ are the gimbal angles within the motor pitch/yaw hubs. The neural network weights for $\hat{W}$ and $\hat{V}$ were initialized as null matrices. While the input and output dimensions of the weights are fixed, a two-layer neural network allows for an infinite number of hidden nodes. For this application, it was found that 40 hidden nodes gave sufficient performance without significantly affecting the CPU performance. The *sigmoid* activation function was chosen, where $\sigma(x) = \frac{1}{1+e^{-x}}$.

Initial gain tuning of K_v and λ were completed with the neural network portion of the controller disabled and by looking at the relationship with a PD controller. By using the gain tuning algorithms defined in Eq. (7) and selecting the design parameter $\alpha = 0$, a PD control system is given where $K_p = \frac{\omega_n^2}{J_i^{-1}}$ and $K_d = \frac{2\zeta\omega_n}{J_i^{-1}}$. The relationship between K_v/λ and K_p/K_d is given by $J_i^{-1}K_p = K_v\lambda$ and $J_i^{-1}K_d = K_v + \lambda$, where it can be shown that a real solution for K_v and λ only exists for $\zeta \geq 1$. The relationship is given by

$$K_v = \omega_n\zeta + \sqrt{\zeta^2 - 1} \qquad \text{and} \qquad \lambda = \omega_n\zeta - \sqrt{\zeta^2 - 1} \tag{25}$$

where $K_v = \lambda$ for $\zeta = 1$. This methodology was chosen for a more intuitive approach to gain tuning, similar to the heritage system, such that tuning can be understood in terms of system bandwidth. Because the damping ratio is typically near being critically damped, the tuning for when $\zeta < 1$ is given by $K_v = \lambda = \omega_n\zeta$, which gives a close approximation for the equivalent K_p/K_d values. Using this methodology without the neural network enabled, the selected K_v and λ gains give the controller a stable response similar to a PD controller where there is a bias in the pointing error due to disturbances. The neural network was then enabled and tuned until sufficient performance was achieved. Final tuning gains for all the parameters were determined as $K_v = 5.0265$, $\lambda = 5.0265$, $M = 1.2$, $N = 1.2$, $\kappa = 0.1$, $\Theta_m = 0.01$, and $K_z = 0.001$, where the equivalent PD gains are $K_p = 14000$ and $K_d = 2760$.

4.3. MSO SETUP

Whereas the SNNARC algorithm is limited to estimation in its control channels only, the MSO algorithm estimation occurs as an observer, and, as such, can measure the full state of the system. This has potential advantages where dynamics in the roll channel may affect the dynamics in the others. The observer states are given as $\hat{\mathbf{x}} = [\hat{\omega}_\alpha \quad \hat{\omega}_\beta \quad \hat{\omega}_\gamma]^T$, which correspond to the estimated inertial roll, pitch, and yaw angular rates, respectively. When the MSO algorithm is first enabled, the estimated states are set to the current measured state vector $\mathbf{x} = [\omega_\alpha \quad \omega_\beta \quad \omega_\gamma]^T$.

The MSO algorithm is setup in a similar manner to SNNARC so that a fair comparison can be made. The same basis vector, $\phi(\mathbf{z}) = \begin{bmatrix}\theta_{e\beta} & \theta_{e\gamma} & \omega_\alpha & \omega_\beta & \omega_\gamma & H_\beta & H_\gamma\end{bmatrix}^T$, is used and the neural network weights, $\hat{W}$, are initialized as null. Because the MSO algorithm uses only a single neural network, the size of the weight matrix is fixed by the input/output size and no hidden nodes are present. For tuning of the linear controller gains, as with SNNARC, the design parameter in Eq. (7) is set as $\alpha = 0$ and gains are tuned using ω_n and ζ such that $K_p = \frac{\omega_n^2}{J_i^{-1}}$ and $K_d = \frac{2\zeta\omega_n}{J_i^{-1}}$ in Eq. (22). Once the PD performance is stable, the neural network is enabled and tuned until sufficient performance was achieved. Final tuning gains for all the parameters are given as $K_p = 14000$, $K_d = 2760$, $K_2 = 10$, $\Gamma = 120$, and $\sigma = 0.001$.

5. SIMULATION RESULTS

With the controller tuning configured, the parameters were input into a configuration file that gets loaded in the flight software on startup. The flight software interfaces with PortOSim, creating a flight-like simulation environment. The simulation was allowed to run a few minutes to permit settling of any initial transients in the system. The three control methods were then tested for approximately ten minutes for three cases including nominal environment, elevated disturbance environment, and severe disturbance environment. The elevated disturbance environment moves the center of mass of the centerbody relative to the gimbal center such that there is a 0.25 ft-lbs imbalance along each axis while for the severe disturbance environment the imbalance is 0.5 ft-lbs along each axis. Disturbances in the shaft rotation motors cause a sinusoidal disturbance in pointing, while disturbances in the upper atmosphere winds increase the pendulous motion imparted to the platform from the high-altitude balloon dynamics. The models and dynamics are internal to the PortOSim environment, with tuning of the disturbance magnitude being controllable. As compared to the nominal configuration, the elevated disturbance environment doubles the magnitude of the disturbances, while the severe disturbance environment increase the disturbance by an order of magnitude.

5.1. PID RESULTS

For the nominal configuration, the PID control system performed well. Over a ten minute period of pointing, the control system was able to maintain an RMS error value of 0.254 arcseconds in pitch ($\theta_{e\beta}$) and 0.266 arcseconds in yaw ($\theta_{e\gamma}$), as seen in Figure 1. This performance is on a par with previous SWIL testing of other missions.

With the system baselined for the nominal case, the two disturbance cases were run. In the elevated disturbance case, the RMS error value increased to 0.387 arcseconds in pitch and 0.402 arcseconds in yaw, while the severe disturbance environment increased the RMS error values to 0.725 arcseconds in pitch and 0.735 arcseconds in yaw, as seen in Figure 2.

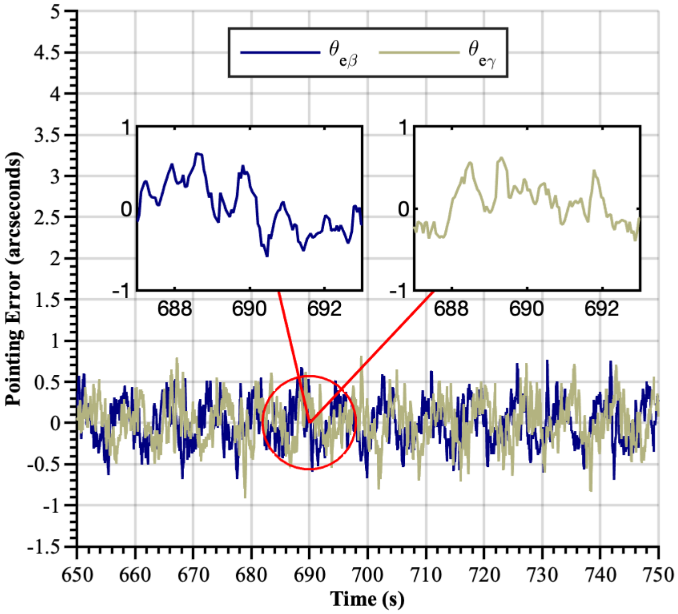
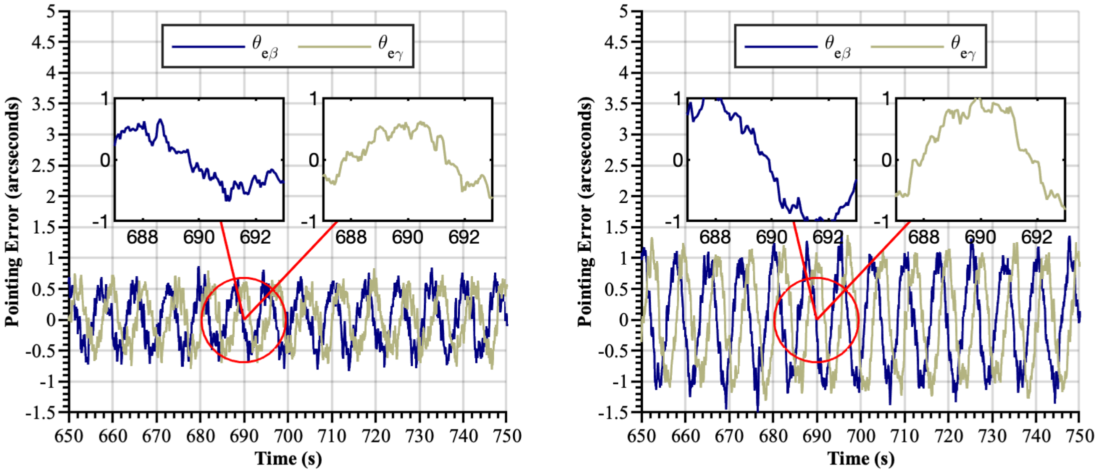


Figure 1. The performance of the PID control system for the nominal case is demonstrated giving an RMS error value of 0.254 arcseconds in pitch ($\theta_{e\beta}$) and 0.266 arcseconds in yaw ($\theta_{e\gamma}$).

While the elevated disturbance environment only resulted in a marginal degradation of performance, the loss in performance for the severe case was significant, where the pointing exceed the one arcsecond pointing boundary. The science requirements for this mission require that pointing remains within one arcsecond, indicating that the selected control tuning is insufficient for these severe conditions. While the flight software allows for modifying controller gains in real time, it is desired that the system can remain within mission requirements without user intervention.



(a)

(b)

Figure 2. The performance of the PID control system under elevated disturbance (a) has an RMS error value of 0.387 arcseconds in pitch ($\theta_{e\beta}$) and 0.402 arcseconds in yaw ($\theta_{e\gamma}$), while for the severe disturbance (b) an RMS error value of 0.725 arcseconds in pitch ($\theta_{e\beta}$) and 0.735 arcseconds in yaw ($\theta_{e\gamma}$).

5.2. SNNARC RESULTS

In the nominal configuration, the SNNARC control system performed marginally better compared to the PID system. Over a ten minute period of pointing, the control system was able to maintain an RMS error value of 0.198 arcseconds in pitch and 0.212 arcseconds in yaw, as seen in Figure 3. This is about a 0.05 arcsecond improvement compared to the PID

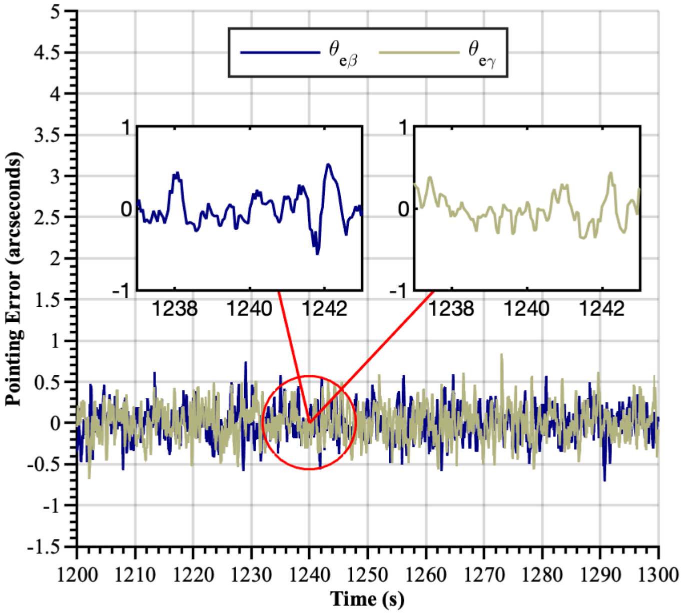


Figure 3. The performance of the SNNARC control system for the nominal case is demonstrated giving an RMS error value of 0.198 arcseconds in pitch ($\theta_{e\beta}$) and 0.212 arcseconds in yaw ($\theta_{e\gamma}$), showing a slight improvement as compared to the heritage PID control system.

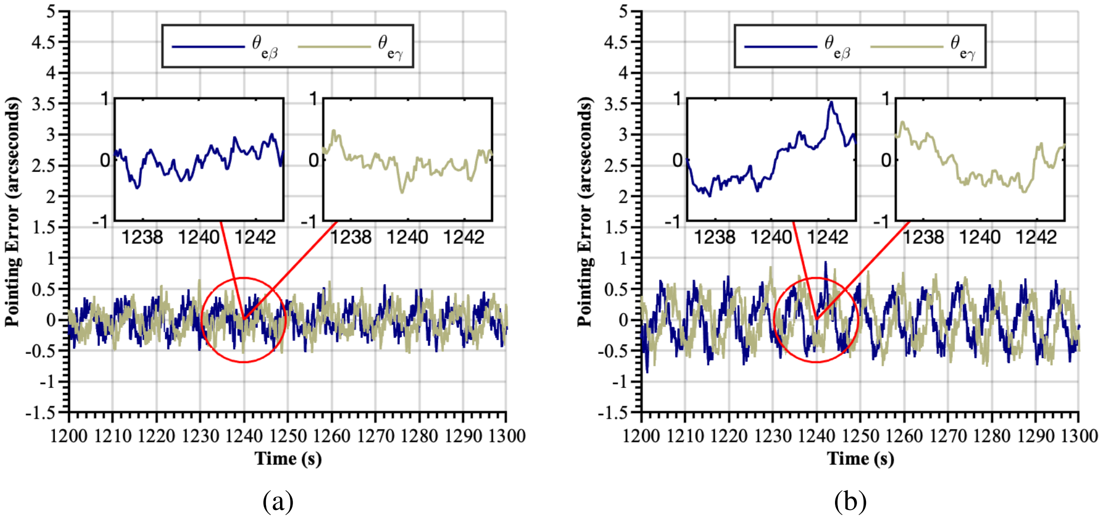


Figure 4. The performance of the SNNARC control system under elevated disturbance (a) has an RMS error value of 0.200 arcseconds in pitch ($\theta_{e\beta}$) and 0.211 arcseconds in yaw ($\theta_{e\gamma}$), while for the severe disturbance (b) an RMS error value of 0.332 arcseconds in pitch ($\theta_{e\beta}$) and 0.348 arcseconds in yaw ($\theta_{e\gamma}$).

control system. In the elevated disturbance case, the RMS error value was 0.200 arcseconds in pitch and 0.211 arcseconds in yaw, while for the severe disturbance environment the RMSerror values increased to 0.332 arcseconds in pitch and 0.348 arcseconds in yaw, as seen in Figure 4. It can be seen that for the elevated disturbance case, there was almost no change in the pointing performance from the SNNARC algorithm. Even with the severe case, there was only a marginal degradation of performance, with the pointing still outperforming the PID system in the elevated disturbance case.

5.3. MSO RESULTS

In the nominal configuration, the MSO control system performed slightly better compared to the SNNARC system. Over a ten minute period of pointing, the control system was able to maintain an RMS error value of 0.182 arcseconds in pitch and 0.200 arcseconds in yaw, as seen in Figure 5.

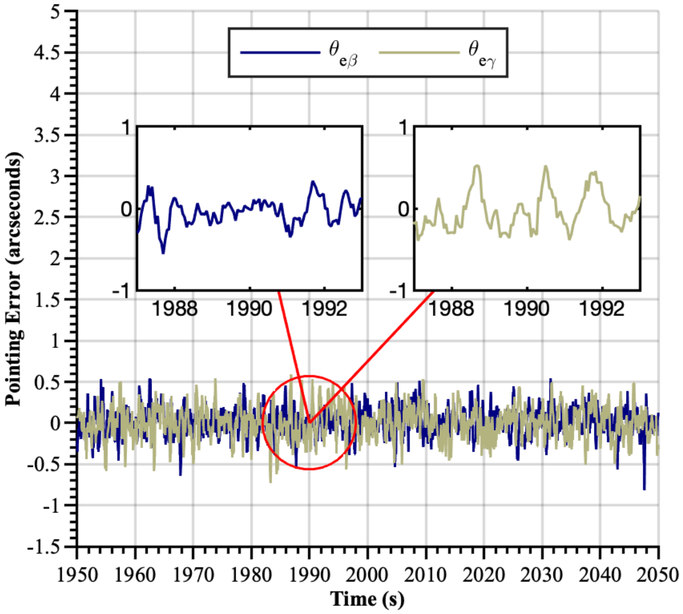
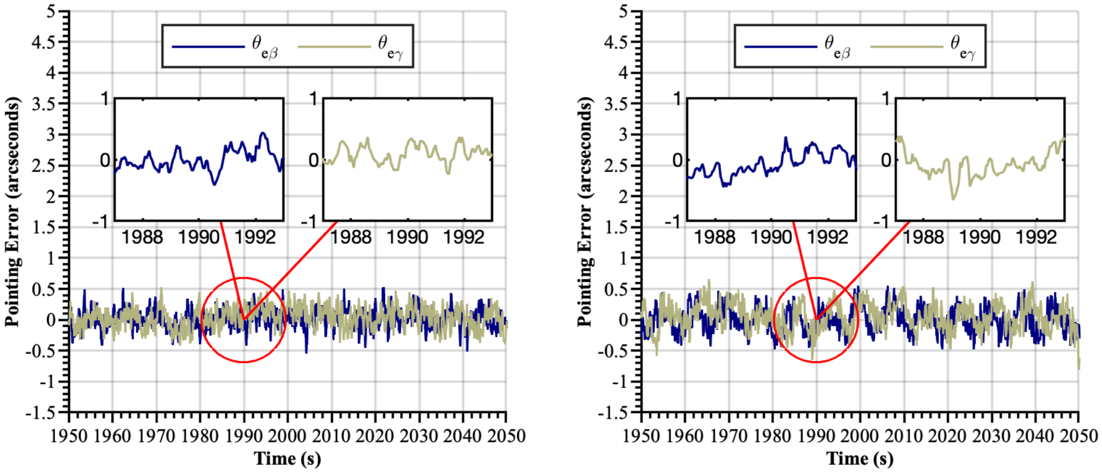


Figure 5. The performance of the MSO control system for the nominal case is demonstrated giving an RMS error value of 0.182 arcseconds in pitch ($\theta_{e\beta}$) and 0.200 arcseconds in yaw ($\theta_{e\gamma}$), showing a slight improvement as compared to the heritage PID control system and the SNNARC system.

In the elevated disturbance case, the RMS error value actually improved to 0.162 arcseconds in pitch and 0.170 arcseconds in yaw, while for the severe disturbance environment the RMS error values remained near the nominal case at 0.201 arcseconds in pitch and 0.232 arcseconds in yaw, as seen in Figure 6. It can be seen that for the elevated disturbance case, there was actually a slight improvement in pointing. With the severe case, there was almost no degradation of pointing performance as compared to nominal, outperforming both the PID and SNNARC algorithms.



(a) (b)

Figure 6. The performance of the MSO control system under elevated disturbance (a) has an RMS error value of 0.162 arcseconds in pitch ($\theta_{e\beta}$) and 0.170 arcseconds in yaw ($\theta_{e\gamma}$), while for the severe disturbance (b) an RMS error value of 0.201 arcseconds in pitch ($\theta_{e\beta}$) and 0.232 arcseconds in yaw ($\theta_{e\gamma}$).

5.4. RESULTS DISCUSSION

Under nominal conditions, all control systems performed as expected with only marginal improvements in the performance of the SNNARC and MSO control systems as compared to PID. Under these conditions, all three systems converged to steady state quickly, with no significant difference in settling times. This in itself gives confidence in the application of the SNNARC and MSO methodologies, indicating that they could, at a minimum, match the performance of the heritage control system. The expected benefit of the two methodologies, however, lies in their ability to compensate for unpredicted nonlinear dynamics. This capability is clearly seen when the disturbance environment is changed from the nominal. While the performance of the PID system began to degrade, both SNNARC and MSO maintained similar performance under elevated disturbances. The MSO algorithm actually produced a slight improvement in performance, which is likely related to increased measured dynamics leading to an improvement in the nonlinear estimation. As the disturbance environment continued to increase, the PID performance continued to degrade until the performance was no longer adequate. The SNNARC control system performance had only a marginal degradation of performance at the maximum disturbance tested, whereas the MSO algorithm had almost no degradation. In fact, the performance of the MSO algorithm at maximum disturbance exceed that of the PID system at nominal. As in the nominal case, it was found that all three control systems had similar settling times, reaching steady state values in only a few seconds.

The difference in performance between the three control systems is readily apparent by examining an empirical cumulative distribution function (CDF), as seen in Figure 7 for the pitch channel (yaw exhibits similar behavior). The empirical CDF, $F(x)$, is the proportion of the values in x less than or equal to q, where x is the pointing error in arcseconds and q is a value in x. It can be seen that the CDF is nearly constant for the MSO case and changes minimally for SNNARC as the disturbance environment is increased. However, the CDF moves significantly to the right for the PID system, illustrating the degradation in pointing performance.

While it is understood that the neural network learning leads to significant improvements in pointing as compared to the PID system, the differences between the two neural network methods are a little more nuanced. The MSO method is at a slight advantage as its estimation occurs, in essence, outside the control space. In fact, the MSO method is looking at all three angular velocity channels and estimating their dynamics, whereas the SNNARC method can only estimate the disturbances in the control channels. Along with this extra

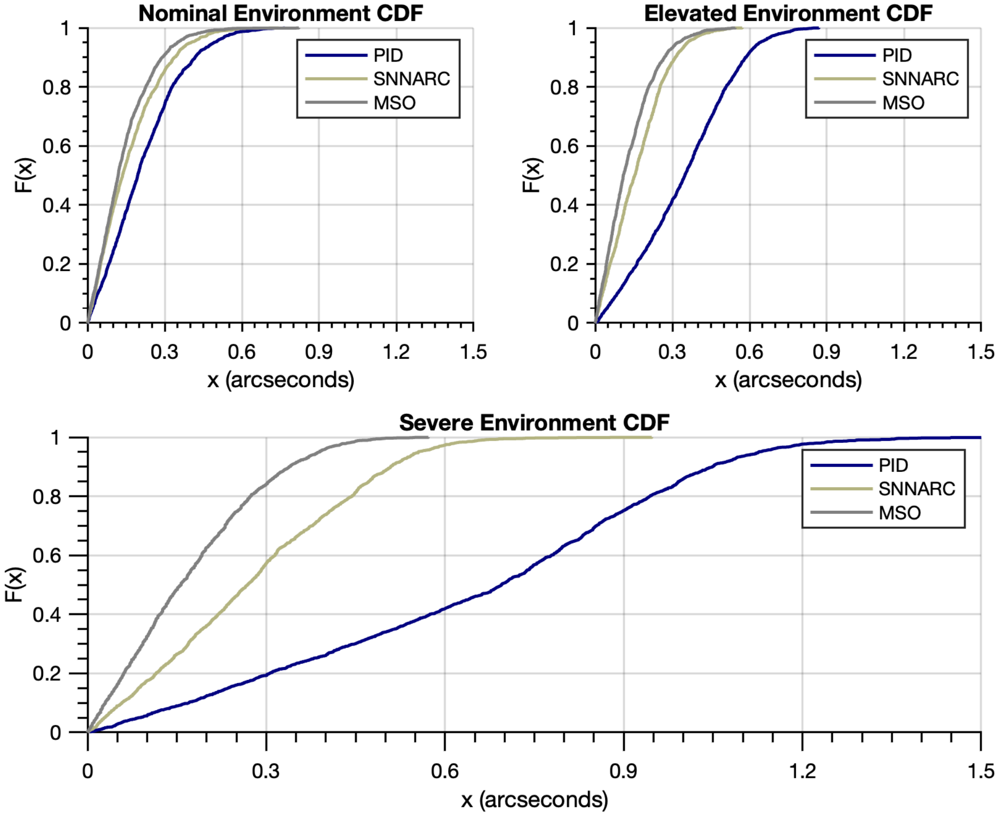


Figure 7. The empirical cumulative distribution function shows that MSO and SNNARC have minimal changes in pitch pointing performance, whereas the PID control system degrades quickly.

channel, the estimation through an observer produces a filtering effect as the tracking error does not drive the learning. Together, this has given the MSO method a slight advantage in pointing performance as compared to SNNARC.

6. CONCLUSION

This research has implemented two new control methodologies to the Wallops Arc-Second Pointer system for High Altitude Balloon platforms. The SNNARC methodology uses a filtered tracking error and a two-layer neural network to estimate nonlinear dynamics within the control space. The MSO methodology uses an observer and a one-layer neural network to estimate the nonlinear dynamics outside the control space and then used input-output feedback linearization to control the system. Lyapunov-based proofs show that both control system were ultimately upper bounded. The two new control systems were written into separate C++ classes and implemented into the flight software. Software-in-the-loop testing was conducted using the NASA PortOSim simulation environment, which provides a flight-like dynamics environment. The heritage PID control system and the two new methodologies were tested for a nominal case and two disturbance cases. It was found that while all three performed similarly for the nominal case, the two new methodologies were able to maintain their performance while the PID system performance degraded significantly. This research has shown that the addition of a neural network to a gimbal controlled inertial pointing system suspended from a high-altitude balloon platform can provide improved performance as compared to traditional heritage pointing systems.

With the theoretical framework formulated, flight software written, and software-in-the-loop testing completed, future work revolves around hardware testing in preparation for science operations of the the second Picture-C mission. The mission is expected to fly in the Fall 2021 BPO balloon campaign from Fort Sumner, New Mexico. Once launched, the two new control methodologies will be tested against the heritage PID system to confirm that increased pointing performance is achieved while performing science mission operations.

APPENDIX

1. SNNARC LYAPUNOV PROOF

Before $\tilde{\mathcal{F}}_{\mathbf{x}}$ is defined, additional parameters are first defined. Let

$$\tilde{\sigma} = \sigma\left(V^T \phi\left(\mathbf{z}\right)\right) - \sigma\left(\hat{V}^T \phi\left(\mathbf{z}\right)\right)$$

and take the Taylor series expansion of σ about $\hat{V}^T\phi(\mathbf{z})$.

$$\sigma\left(V^T\phi(\mathbf{z})\right) = \sigma\left(\hat{V}^T\phi(\mathbf{z})\right) + \sigma'\left(\hat{V}^T\phi(\mathbf{z})\right)\left(V^T\phi(\mathbf{z}) - \hat{V}^T\phi(\mathbf{z})\right) + \text{H.O.T.}$$
$$\sigma\left(V^T\phi(\mathbf{z})\right) = \sigma\left(\hat{V}^T\phi(\mathbf{z})\right) + \sigma'\left(\hat{V}^T\phi(\mathbf{z})\right)\left(\tilde{V}^T\phi(\mathbf{z})\right) + \text{H.O.T.}$$

Then

$$\tilde{\sigma} = \sigma\left(\hat{V}^T\phi(\mathbf{z})\right) + \sigma'\left(\hat{V}^T\phi(\mathbf{z})\right)\left(\tilde{V}^T\phi(\mathbf{z})\right) + \text{H.O.T.} - \sigma\left(\hat{V}^T\phi(\mathbf{z})\right)$$
$$\tilde{\sigma} = \sigma'\left(\hat{V}^T\phi(\mathbf{z})\right)\left(\tilde{V}^T\phi(\mathbf{z})\right) + \text{H.O.T..}$$

For simplicity, define

$$\sigma = \sigma\left(V^T\phi(\mathbf{z})\right) \quad \text{and} \quad \hat{\sigma} = \sigma\left(\hat{V}^T\phi(\mathbf{z})\right) \quad \text{and} \quad \hat{\sigma}' = \sigma'\left(\hat{V}^T\phi(\mathbf{z})\right)$$

such that

$$\tilde{\sigma} = \hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + \text{H.O.T.}$$

We can now take the filtered tracking error from Eq. (14), temporarily remove $\mathbf{u}_r$, and apply the definitions above such that the closed-loop filtered tracking error now becomes

$$\dot{\mathbf{r}} = -K_v\mathbf{r} + W^T\sigma\left(V^T\phi(\mathbf{z})\right) - \hat{W}^T\sigma\left(\hat{V}^T\phi(\mathbf{z})\right) + \epsilon \qquad \rightarrow \qquad \dot{\mathbf{r}} = -K_v\mathbf{r} + W^T\sigma - \hat{W}^T\hat{\sigma} + \epsilon$$

and to this add and subtract $W^T\hat{\sigma}$

$$\dot{\mathbf{r}} = -K_v\mathbf{r} + W^T\sigma - \hat{W}^T\hat{\sigma} + W^T\hat{\sigma} - W^T\hat{\sigma} + \epsilon$$
$$\dot{\mathbf{r}} = -K_v\mathbf{r} + W^T(\sigma - \hat{\sigma}) + \left(W^T - \hat{W}^T\right)\hat{\sigma} + \epsilon$$
$$\dot{\mathbf{r}} = -K_v\mathbf{r} + W^T\tilde{\sigma} + \tilde{W}^T\hat{\sigma} + \epsilon$$

and then add and subtract $\hat{W}^T\tilde{\sigma}$

$$
\begin{aligned}
\dot{\mathbf{r}} &= -K_v\mathbf{r} + W^T\tilde{\sigma} + \tilde{W}^T\hat{\sigma} + \hat{W}^T\tilde{\sigma} - \hat{W}^T\tilde{\sigma} + \epsilon \\
\dot{\mathbf{r}} &= -K_v\mathbf{r} + \left(W^T - \hat{W}^T\right)\tilde{\sigma} + \tilde{W}^T\hat{\sigma} + \hat{W}^T\tilde{\sigma} + \epsilon \\
\dot{\mathbf{r}} &= -K_v\mathbf{r} + \tilde{W}^T\tilde{\sigma} + \tilde{W}^T\hat{\sigma} + \hat{W}^T\tilde{\sigma} + \epsilon \\
\dot{\mathbf{r}} &= -K_v\mathbf{r} + \tilde{W}^T\hat{\sigma} + \tilde{W}^T\tilde{\sigma} + \hat{W}^T\tilde{\sigma} + \epsilon
\end{aligned}
$$

and then substitute in the definition of $\tilde{\sigma}$

$$
\begin{aligned}
\dot{\mathbf{r}} &= -K_v\mathbf{r} + \tilde{W}^T\hat{\sigma} + \tilde{W}^T\left(\hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + \text{H.O.T.}\right) + \hat{W}^T\left(\hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + \text{H.O.T.}\right) + \epsilon \\
\dot{\mathbf{r}} &= -K_v\mathbf{r} + \tilde{W}^T\hat{\sigma} + \tilde{W}^T\hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + \tilde{W}^T\text{H.O.T.} + \hat{W}^T\hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + \hat{W}^T\text{H.O.T.} + \epsilon \\
\dot{\mathbf{r}} &= -K_v\mathbf{r} + \tilde{W}^T\hat{\sigma} + \tilde{W}^T\hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + \hat{W}^T\hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + \left(\tilde{W}^T + \hat{W}^T\right)\text{H.O.T.} + \epsilon \\
\dot{\mathbf{r}} &= -K_v\mathbf{r} + \tilde{W}^T\hat{\sigma} + \tilde{W}^T\hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + \hat{W}^T\hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + \left(W^T - \hat{W}^T + \hat{W}^T\right)\text{H.O.T.} + \epsilon \\
\dot{\mathbf{r}} &= -K_v\mathbf{r} + \tilde{W}^T\hat{\sigma} + \tilde{W}^T\hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + \hat{W}^T\hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + W^T\text{H.O.T.} + \epsilon \\
\dot{\mathbf{r}} &= -K_v\mathbf{r} + \tilde{W}^T\hat{\sigma} + \tilde{W}^T\hat{\sigma}'\left(V^T\phi(\mathbf{z}) - \hat{V}^T\phi(\mathbf{z})\right) + \hat{W}^T\hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + W^T\text{H.O.T.} + \epsilon \\
\dot{\mathbf{r}} &= -K_v\mathbf{r} + \tilde{W}^T\hat{\sigma} + \tilde{W}^T\hat{\sigma}'V^T\phi(\mathbf{z}) - \tilde{W}^T\hat{\sigma}'\hat{V}^T\phi(\mathbf{z}) + \hat{W}^T\hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + W^T\text{H.O.T.} + \epsilon \\
\dot{\mathbf{r}} &= -K_v\mathbf{r} + \tilde{W}^T\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\phi(\mathbf{z})\right) + \tilde{W}^T\hat{\sigma}'V^T\phi(\mathbf{z}) + \hat{W}^T\hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + W^T\text{H.O.T.} + \epsilon \\
\dot{\mathbf{r}} &= -K_v\mathbf{r} + \tilde{W}^T\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\phi(\mathbf{z})\right) + \hat{W}^T\hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + \left(\tilde{W}^T\hat{\sigma}'V^T\phi(\mathbf{z}) + W^T\text{H.O.T.} + \epsilon\right) \\
\dot{\mathbf{r}} &= -K_v\mathbf{r} + \tilde{W}^T\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\phi(\mathbf{z})\right) + \hat{W}^T\hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + \omega \\
&\qquad\qquad \text{where} \quad \omega = \tilde{W}^T\hat{\sigma}'V^T\phi(\mathbf{z}) + W^T\text{H.O.T.} + \epsilon \\
\dot{\mathbf{r}} &= -K_v\mathbf{r} + \tilde{\mathcal{F}}_{\mathbf{x}} \quad \text{where} \quad \tilde{\mathcal{F}}_{\mathbf{x}} = \tilde{W}^T\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\phi(\mathbf{z})\right) + \hat{W}^T\hat{\sigma}'\tilde{V}^T\phi(\mathbf{z}) + \omega
\end{aligned}
$$

and where it can be shown [22] that the higher order terms in ω are bounded by

$$
\|\omega\| \leq C_0 + C_1\left\|\tilde{\Theta}\right\| + C_2\|\mathbf{r}\|\left\|\tilde{\Theta}\right\|.
$$

Replace back the robustifying portion of the control, $\mathbf{u}_r$, such that the final closed-loop system is given by

$$\dot{\mathbf{r}} = -K_v\mathbf{r} + \tilde{W}^T\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\phi\left(\mathbf{z}\right)\right) + \hat{W}^T\hat{\sigma}'\tilde{V}^T\phi\left(\mathbf{z}\right) + \omega + u_r$$

By using this closed-loop filtered tracking error and the weight update law in Eq. (15), take the derivative of the Lyapunov function in Eq. (16) piecewise for simplicity where

$$\begin{aligned} L &= L_1 + L_2 + L_3 \\ L_1 &= \frac{1}{2}\mathbf{r}^T\mathbf{r} \\ L_2 &= \frac{1}{2}tr\left\{\tilde{W}^T M^{-1}\tilde{W}\right\} \\ L_3 &= \frac{1}{2}tr\left\{\tilde{V}^T N^{-1}\tilde{V}\right\}, \end{aligned}$$

therefore

$$\begin{aligned} \dot{L}_1 = \frac{1}{2}\left(\dot{\mathbf{r}}^T\mathbf{r} + \mathbf{r}^T\dot{\mathbf{r}}\right) \;\rightarrow\; \dot{L}_1 &= -\mathbf{r}^T K_v\mathbf{r} + \frac{1}{2}\left(\tilde{\mathcal{F}}_{\mathbf{x}}^T\mathbf{r} + \mathbf{r}^T\tilde{\mathcal{F}}_{\mathbf{x}}\right) \\ \dot{L}_1 &= -\mathbf{r}^T K_v\mathbf{r} + tr\left\{\frac{1}{2}\left(\tilde{\mathcal{F}}_{\mathbf{x}}^T\mathbf{r} + \mathbf{r}^T\tilde{\mathcal{F}}_{\mathbf{x}}\right)\right\} \\ \dot{L}_1 &= -\mathbf{r}^T K_v\mathbf{r} + tr\left\{\mathbf{r}^T\tilde{\mathcal{F}}_{\mathbf{x}}\right\} \\ \dot{L}_1 &= -\mathbf{r}^T K_v\mathbf{r} \\ &\quad + tr\left\{\mathbf{r}^T\left[\tilde{W}^T\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\phi\left(\mathbf{z}\right)\right) + \hat{W}^T\hat{\sigma}'\tilde{V}^T\phi\left(\mathbf{z}\right) + \omega + u_r\right]\right\} \end{aligned}$$

and because $tr\left\{AB\right\} = tr\left\{BA\right\}$

$$\begin{aligned} \dot{L}_1 &= -\mathbf{r}^T K_v\mathbf{r} + tr\left\{\tilde{W}^T\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\phi\left(\mathbf{z}\right)\right)\mathbf{r}^T\right\} + tr\left\{\hat{W}^T\hat{\sigma}'\tilde{V}^T\phi\left(\mathbf{z}\right)\mathbf{r}^T\right\} + \mathbf{r}^T\left(\omega + u_r\right) \\ \dot{L}_1 &= -\mathbf{r}^T K_v\mathbf{r} + tr\left\{\tilde{W}^T\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\phi\left(\mathbf{z}\right)\right)\mathbf{r}^T\right\} + tr\left\{\tilde{V}^T\phi\left(\mathbf{z}\right)\mathbf{r}^T\hat{W}^T\hat{\sigma}'\right\} + \mathbf{r}^T\left(\omega + u_r\right) \end{aligned}$$

then

$$
\begin{aligned}
\dot{L}_2 = tr\left\{\tilde{W}^T M^{-1}\dot{\tilde{W}}\right\} \quad \rightarrow \quad \dot{L}_2 &= tr\left\{\tilde{W}^T M^{-1}\left[\dot{W} - \dot{\hat{W}}\right]\right\} \\
\dot{L}_2 &= tr\left\{-\tilde{W}^T M^{-1}\left[(M\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\phi\left(\mathbf{z}\right)\right)\mathbf{r}^T - \kappa\left\|\mathbf{r}\right\| M\hat{W}\right]\right\} \\
\dot{L}_2 &= tr\left\{-\tilde{W}^T\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\phi\left(\mathbf{z}\right)\right)\mathbf{r}^T + \kappa\left\|\mathbf{r}\right\|\tilde{W}^T\hat{W}\right\} \\
\dot{L}_2 &= -tr\left\{\tilde{W}^T\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\phi\left(\mathbf{z}\right)\right)\mathbf{r}^T\right\} + \kappa\left\|\mathbf{r}\right\| tr\left\{\tilde{W}^T\hat{W}\right\}
\end{aligned}
$$

and

$$
\begin{aligned}
\dot{L}_3 = tr\left\{\tilde{V}^T N^{-1}\dot{\tilde{V}}\right\} \quad \rightarrow \quad \dot{L}_3 &= tr\left\{\tilde{V}^T N^{-1}\left[\dot{V} - \dot{\hat{V}}\right]\right\} \\
\dot{L}_3 &= tr\left\{-\tilde{V}^T N^{-1}\left[N\phi\left(\mathbf{z}\right)\mathbf{r}^T\hat{W}^T\hat{\sigma}' - \kappa\left\|\mathbf{r}\right\| N\hat{V}\right]\right\} \\
\dot{L}_3 &= tr\left\{-\tilde{V}^T\phi\left(\mathbf{z}\right)\mathbf{r}^T\hat{W}^T\hat{\sigma}' + \kappa\left\|\mathbf{r}\right\|\tilde{V}^T\hat{V}\right\} \\
\dot{L}_3 &= -tr\left\{\tilde{V}^T\phi\left(\mathbf{z}\right)\mathbf{r}^T\hat{W}^T\hat{\sigma}'\right\} + \kappa\left\|\mathbf{r}\right\| tr\left\{\tilde{V}^T\hat{V}\right\}
\end{aligned}
$$

so that when combining gives

$$
\begin{aligned}
\dot{L} = &- \mathbf{r}^T K_v\mathbf{r} + tr\left\{\tilde{W}^T\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\phi\left(\mathbf{z}\right)\right)\mathbf{r}^T\right\} + tr\left\{\tilde{V}^T\phi\left(\mathbf{z}\right)\mathbf{r}^T\hat{W}^T\hat{\sigma}'\right\} + \mathbf{r}^T\left(\omega + u_r\right) \\
&- tr\left\{\tilde{W}^T\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\phi\left(\mathbf{z}\right)\right)\mathbf{r}^T\right\} + \kappa\left\|\mathbf{r}\right\| tr\left\{\tilde{W}^T\hat{W}\right\} \\
&- tr\left\{\tilde{V}^T\phi\left(\mathbf{z}\right)\mathbf{r}^T\hat{W}^T\hat{\sigma}'\right\} + \kappa\left\|\mathbf{r}\right\| tr\left\{\tilde{V}^T\hat{V}\right\} \\
\dot{L} = &- \mathbf{r}^T K_v\mathbf{r} + \kappa\left\|\mathbf{r}\right\| tr\left\{\tilde{W}^T\hat{W}\right\} + \kappa\left\|\mathbf{r}\right\| tr\left\{\tilde{V}^T\hat{V}\right\} + \mathbf{r}^T\left(\omega + u_r\right) \\
\dot{L} = &- \mathbf{r}^T K_v\mathbf{r} + \kappa\left\|\mathbf{r}\right\| tr\left\{\tilde{\Theta}^T\hat{\Theta}\right\} + \mathbf{r}^T\left(\omega + u_r\right)
\end{aligned}
$$

where

$$
\tilde{\Theta} = \begin{bmatrix} \tilde{V} & 0 \\ 0 & \tilde{W} \end{bmatrix} \qquad \text{and} \qquad \hat{\Theta} = \begin{bmatrix} \hat{V} & 0 \\ 0 & \hat{W} \end{bmatrix}.
$$

Because $\hat{\Theta} = \Theta - \tilde{\Theta}$, the trace component can be rewritten as

$$
tr\left\{\tilde{\Theta}^T\left(\Theta - \tilde{\Theta}\right)\right\} \quad \rightarrow \quad tr\left\{\tilde{\Theta}^T\Theta\right\} - tr\left\{\tilde{\Theta}^T\tilde{\Theta}\right\}
$$

and using the properties of the $tr\left\{\right\}$ operator and the bound $\|\Theta\| < \Theta_m$ it can be shown that

$$\begin{aligned}
tr\left\{\tilde{\Theta}^T\hat{\Theta}\right\} &\leq \left\|\tilde{\Theta}\right\|\Theta_m - \left\|\tilde{\Theta}\right\|^2 \\
tr\left\{\tilde{\Theta}^T\hat{\Theta}\right\} &\leq \left\|\tilde{\Theta}\right\|\left(\Theta_m - \left\|\tilde{\Theta}\right\|\right)
\end{aligned}$$

and using the property $\mathbf{r}^T\mathbf{r} = \|\mathbf{r}\|^2$ gives

$$\dot{L} \leq -K_v\|\mathbf{r}\|^2 + \kappa\|\mathbf{r}\|\left\|\tilde{\Theta}\right\|\left(\Theta_m - \left\|\tilde{\Theta}\right\|\right) + \mathbf{r}^T\left(\omega + u_r\right).$$

The robustifying control and bound can now be inserted on the disturbance, ω, such that

$$\begin{aligned}
\dot{L} \leq& -K_v\|\mathbf{r}\|^2 + \kappa\|\mathbf{r}\|\left\|\tilde{\Theta}\right\|\left(\Theta_m - \left\|\tilde{\Theta}\right\|\right) \\
&+ \mathbf{r}^T\left(C_0 + C_1\left\|\tilde{\Theta}\right\| + C_2\|\mathbf{r}\|\left\|\tilde{\Theta}\right\| - K_z\left(\left\|\hat{\Theta}\right\| + \Theta_m\right)\mathbf{r}\right) \\
\dot{L} \leq& -K_v\|\mathbf{r}\|^2 + \kappa\|\mathbf{r}\|\left\|\tilde{\Theta}\right\|\left(\Theta_m - \left\|\tilde{\Theta}\right\|\right) \\
&+ \|\mathbf{r}\|C_0 + \|\mathbf{r}\|C_1\left\|\tilde{\Theta}\right\| + C_2\|\mathbf{r}\|^2\left\|\tilde{\Theta}\right\| - K_z\|\mathbf{r}\|^2\left(\left\|\hat{\Theta}\right\| + \Theta_m\right)
\end{aligned}$$

where as long as $K_z > C_2$

$$\dot{L} \leq -K_v\|\mathbf{r}\|^2 + \kappa\|\mathbf{r}\|\left\|\tilde{\Theta}\right\|\left(\Theta_m - \left\|\tilde{\Theta}\right\|\right) + \|\mathbf{r}\|C_0 + \|\mathbf{r}\|C_1\left\|\tilde{\Theta}\right\|$$

We can now pull out $-\|\mathbf{r}\|$ and complete the square to get

$$\begin{aligned}
&\dot{L} \leq -\|\mathbf{r}\|\left\{K_v\|\mathbf{r}\| - \kappa\left\|\tilde{\Theta}\right\|\left(\Theta_m - \left\|\tilde{\Theta}\right\|\right) - C_0 - C_1\left\|\tilde{\Theta}\right\|\right\} \\
&\dot{L} \leq -\|\mathbf{r}\|\left\{K_v\|\mathbf{r}\| - \kappa\left\|\tilde{\Theta}\right\|\Theta_m + \kappa\left\|\tilde{\Theta}\right\|\left\|\tilde{\Theta}\right\| - C_0 - C_1\left\|\tilde{\Theta}\right\|\right\} \\
&\dot{L} \leq -\|\mathbf{r}\| \\
&\quad\left\{K_v\|\mathbf{r}\| - \kappa\left\|\tilde{\Theta}\right\|\Theta_m + \kappa\left\|\tilde{\Theta}\right\|\left\|\tilde{\Theta}\right\| - C_0 - \frac{\kappa}{\kappa}C_1\left\|\tilde{\Theta}\right\| + \kappa\left(\frac{\Theta_m + \frac{C_1}{\kappa}}{2}\right)^2 - \kappa\left(\frac{\Theta_m + \frac{C_1}{\kappa}}{2}\right)^2\right\}
\end{aligned}$$

$$\dot{L} \leq -\|\mathbf{r}\|\left\{K_v\|\mathbf{r}\|+\kappa\left[\|\tilde{\Theta}\|^2-\|\tilde{\Theta}\|\Theta_m-\frac{C_1}{\kappa}\|\tilde{\Theta}\|+\left(\frac{\Theta_m+\frac{C_1}{\kappa}}{2}\right)^2\right]-C_0-\kappa\left(\frac{\Theta_m+\frac{C_1}{\kappa}}{2}\right)^2\right\}$$

$$\dot{L} \leq -\|\mathbf{r}\|\left\{K_v\|\mathbf{r}\|+\kappa\left[\|\tilde{\Theta}\|^2-2\|\tilde{\Theta}\|\left(\frac{\Theta_m+\frac{C_1}{\kappa}}{2}\right)+\left(\frac{\Theta_m+\frac{C_1}{\kappa}}{2}\right)^2\right]-C_0-\kappa\left(\frac{\Theta_m+\frac{C_1}{\kappa}}{2}\right)^2\right\}$$

$$\dot{L} \leq -\|\mathbf{r}\|\left\{K_v\|\mathbf{r}\|+\kappa\left(\|\tilde{\Theta}\|-\frac{C_3}{2}\right)^2-C_0-\frac{\kappa}{4}C_3^2\right\} \qquad \text{where} \qquad C_3=\Theta_m+\frac{C_1}{\kappa}$$

$$\dot{L} \leq -\|\mathbf{r}\|\left\{K_v\|\mathbf{r}\|+\kappa\left(\|\tilde{\Theta}\|-\frac{C_3}{2}\right)^2-D_1\right\} \qquad \text{where} \qquad D_1=C_0+\frac{\kappa}{4}C_3^2$$

. Stability is finally shown in a compact set about the origin if either $\|\mathbf{r}\| > \delta_r$ or $\|\tilde{\Theta}\| > \delta_\Theta$, which are given as

$$\delta_r=\frac{D_1}{K_v} \qquad \text{and} \qquad \delta_\Theta=\frac{C_3}{2}+\sqrt{\frac{D_1}{\kappa}}.$$

This demonstrates that the Lyapunov function is Ultimately Upper Bounded (UUB) in both $\|\mathbf{r}\|$ and $\|\tilde{\Theta}\|$ and thus shows Lyapunov stability. This completes the proof.

2. MSO LYAPUNOV PROOF - PART ONE

By using the weight update law in Eq. (20) and the closed-loop observer error in Eq. (19), the derivative of the Lyapunov function in Eq. (21) is taken piecewise for simplicity where

$$L = L_1 + L_2$$
$$L_1 = \frac{1}{2}\mathbf{e}_a^T\mathbf{e}_a$$
$$L_2 = \frac{1}{2}tr\left\{\tilde{W}^T\Gamma^{-1}\tilde{W}\right\},$$

therefore

$$\begin{aligned} \dot{L}_1 = \frac{1}{2}\left(\dot{\mathbf{e}}_a^T\mathbf{e}_a + \mathbf{e}_a^T\dot{\mathbf{e}}_a\right) \quad \rightarrow \quad & \dot{L}_1 = -\mathbf{e}_a^T K_2 \mathbf{e}_a + \frac{1}{2}\left(\tilde{\mathcal{F}}_{\mathbf{x}}^T\mathbf{e}_a + \mathbf{e}_a^T\tilde{\mathcal{F}}_{\mathbf{x}}\right) \\ & \dot{L}_1 = -\mathbf{e}_a^T K_2 \mathbf{e}_a + tr\left\{+\frac{1}{2}\left(\tilde{\mathcal{F}}_{\mathbf{x}}^T\mathbf{e}_a + \mathbf{e}_a^T\tilde{\mathcal{F}}_{\mathbf{x}}\right)\right\} \\ & \dot{L}_1 = -\mathbf{e}_a^T K_2 \mathbf{e}_a + tr\left\{\mathbf{e}_a^T\tilde{\mathcal{F}}_{\mathbf{x}}\right\} \\ & \dot{L}_1 = -\mathbf{e}_a^T K_2 \mathbf{e}_a + tr\left\{\mathbf{e}_a^T\left[\tilde{W}^T\phi\left(\mathbf{z}\right) + \varepsilon\right]\right\} \end{aligned}$$

and because $tr\{AB\} = tr\{BA\}$

$$\dot{L}_1 = -\mathbf{e}_a^T K_2 \mathbf{e}_a + tr\left\{\tilde{W}^T\phi\left(\mathbf{z}\right)\mathbf{e}_a^T\right\} + \mathbf{e}_a^T\varepsilon$$

then

$$\begin{aligned} \dot{L}_2 = tr\left\{\tilde{W}^T\Gamma^{-1}\dot{\tilde{W}}\right\} \quad \rightarrow \quad & \dot{L}_2 = tr\left\{\tilde{W}^T\Gamma^{-1}\left[\dot{W} - \dot{\hat{W}}\right]\right\} \\ & \dot{L}_2 = tr\left\{-\tilde{W}^T\Gamma^{-1}\left[\Gamma\phi\left(\mathbf{z}\right)\mathbf{e}_a^T - \kappa\left\|\mathbf{e}_a\right\|\Gamma\hat{W}\right]\right\} \\ & \dot{L}_2 = tr\left\{-\tilde{W}^T\phi\left(\mathbf{z}\right)\mathbf{e}_a^T + \kappa\left\|\mathbf{e}_a\right\|\tilde{W}^T\hat{W}\right\} \\ & \dot{L}_2 = -tr\left\{\tilde{W}^T\phi\left(\mathbf{z}\right)\mathbf{e}_a^T\right\} + \kappa\left\|\mathbf{e}_a\right\| tr\left\{\tilde{W}^T\hat{W}\right\} \end{aligned}$$

so that combining gives

$$\begin{aligned} \dot{L} &= -\,\mathbf{e}_a^T K_2 \mathbf{e}_a + tr\left\{\tilde{W}^T\phi\left(\mathbf{z}\right)\mathbf{e}_a^T\right\} + \mathbf{e}_a^T\varepsilon - tr\left\{\tilde{W}^T\phi\left(\mathbf{z}\right)\mathbf{e}_a^T\right\} + \kappa\left\|\mathbf{e}_a\right\| tr\left\{\tilde{W}^T\hat{W}\right\} \\ \dot{L} &= -\,\mathbf{e}_a^T K_2 \mathbf{e}_a + \mathbf{e}_a^T\varepsilon + \kappa\left\|\mathbf{e}_a\right\| tr\left\{\tilde{W}^T\hat{W}\right\} \end{aligned}$$

Because $\hat{W} = W - \tilde{W}$, the trace component can be rewritten as

$$tr\left\{\tilde{W}^T\left(W - \tilde{W}\right)\right\} \quad \rightarrow \quad tr\left\{\tilde{W}^T W\right\} - tr\left\{\tilde{W}^T\tilde{W}\right\}$$

and using the properties of the $tr\,\{\}$ operator and the bound $\|W\| < W_m$ gives

$$\begin{aligned} tr\left\{\tilde{W}^T\hat{W}\right\} &\le \left\|\tilde{W}\right\| W_m - \left\|\tilde{W}\right\|^2 \\ tr\left\{\tilde{W}^T\hat{W}\right\} &\le \left\|\tilde{W}\right\| \left(W_m - \left\|\tilde{W}\right\|\right) \end{aligned}$$

and using the property $\mathbf{e}_a^T\mathbf{e}_a = \|\mathbf{e}\|_a^2$ and the bound of the approximation error $\varepsilon \le \varepsilon_m$ gives

$$\dot{L} \le -K_2\,\|\mathbf{e}\|_a^2 + \|\mathbf{e}_a\|\,\varepsilon_m + \kappa\,\|\mathbf{e}_a\|\,\left\|\tilde{W}\right\|\left(W_m - \left\|\tilde{W}\right\|\right)$$

by removing $-\|\mathbf{e}\|_a$ and completing the square

$$\begin{aligned} \dot{L} &\le -\|\mathbf{e}_a\|\left\{K_2\,\|\mathbf{e}_a\| - \varepsilon_m - \kappa\left\|\tilde{W}\right\|\left(W_m - \left\|\tilde{W}\right\|\right)\right\} \\ \dot{L} &\le -\|\mathbf{e}_a\|\left\{K_2\,\|\mathbf{e}_a\| - \varepsilon_m + \kappa\left\|\tilde{W}\right\|\left\|\tilde{W}\right\| - \kappa\left\|\tilde{W}\right\| W_m\right\} \\ \dot{L} &\le -\|\mathbf{e}_a\|\left\{K_2\,\|\mathbf{e}_a\| - \varepsilon_m + \kappa\left\|\tilde{W}\right\|\left\|\tilde{W}\right\| - \kappa\left\|\tilde{W}\right\| W_m + \frac{\kappa}{4}W_m^2 - \frac{\kappa}{4}W_m^2\right\} \\ \dot{L} &\le -\|\mathbf{e}_a\|\left\{K_2\,\|\mathbf{e}_a\| - \varepsilon_m + \kappa\left(\left\|\tilde{W}\right\| - \frac{1}{2}W_m\right)^2 - \frac{\kappa}{4}W_m^2\right\} \\ \dot{L} &\le -\|\mathbf{e}_a\|\left\{K_2\,\|\mathbf{e}_a\| + \kappa\left(\left\|\tilde{W}\right\| - \frac{1}{2}W_m\right)^2 - D_1\right\} \qquad \text{where} \qquad D_1 = \varepsilon_m + \frac{\kappa}{4}W_m^2 \end{aligned}$$

Stability is finally shown in a compact set about the origin if either $\|\mathbf{e}_a\| > \delta_{ea}$ or $\left\|\tilde{W}\right\| > \delta_W$, which are given as

$$\delta_{ea} = \frac{D_1}{K_2} \qquad \text{and} \qquad \delta_W = \frac{1}{2}W_m + \sqrt{\frac{D_1}{\kappa}}.$$

This demonstrates that the Lyapunov function is Ultimately Upper Bounded (UUB) in both $\|\mathbf{e}_a\|$ and $\left\|\tilde{W}\right\|$ and thus shows Lyapunov stability. This completes the proof.

3. MSO LYAPUNOV PROOF - PART TWO

By using the closed-loop observer error in Eq. (23), the derivative of the Lyapunov function in Eq. (24) is taken where

$$
\begin{aligned}
\dot{L}_1 = \frac{1}{2}\left(\dot{\mathbf{e}}^T\mathbf{e} + \mathbf{e}^T\dot{\mathbf{e}}\right) \quad \rightarrow \quad \dot{L}_1 &= \frac{1}{2}\mathbf{e}^T\left(A^TP + PA\right)\mathbf{e} + \mathbf{e}^TPd \\
\dot{L}_1 &= -\frac{1}{2}\mathbf{e}^TQ\mathbf{e} + \mathbf{e}^TPd \\
\dot{L}_1 &\leq -\frac{1}{2}\lambda_{min}(Q)\,||\mathbf{e}||^2 + ||\mathbf{e}||\,\lambda_{max}(P)d_m \\
\dot{L}_1 &\leq -\,||\mathbf{e}||\left(\frac{1}{2}\lambda_{min}(Q)\,||\mathbf{e}|| + \lambda_{max}(P)d_m\right)
\end{aligned}
$$

where stability is shown in a compact set about the origin if $||\mathbf{e}|| > \delta_e$, which is given as

$$
\delta_e = 2d_m\frac{\lambda_{max}(P)}{\lambda_{min}(Q)}.
$$

This demonstrates that the Lyapunov function is Ultimately Upper Bounded (UUB) in $||\mathbf{e}||$ and thus shows Lyapunov stability. This completes the proof.

FUNDING SOURCES

Portions of this research have been supported through the NASA Pathways Program and NASA Wallops Flight Facility.

ACKNOWLEDGMENTS

First and foremost, a heartfelt thank you to Dr Balakrishnan for his major contributions and being a source of inspiration for this work. Special thanks are also extended to Jim Lanzi, Scott Heatwole, and Zach Peterson for their mentorship, support, and contributions to applying this research to NASA spaceflight systems. Thanks are given also to the

Guidance, Navigation, and Control, and Mission System Engineering Branch (Code 598) of NASA Wallops Flight Facility and the Attitude Control Systems Engineering Branch (Code 591) of NASA Goddard Space Flight Facility, as well as the NASA Pathways Program.

REFERENCES

[1] Garde, G. J., and Fairbrother, D. A., "The NASA Balloon Program - Positioning For the Future," *Proceedings of the 2015 AIAA Balloon Systems Conference (June 2015, Dallas, TX)*, 2015, pp. 1–8. https://doi.org/10.2514/6.2015-2907.

[2] Fairbrother, D. A., "2017 NASA Balloon Program Update," *Proceedings of the 2017 AIAA Balloon Systems Conference (June 2017, Denver, CO)*, 2017, pp. 1–8. https://doi.org/10.2514/6.2017-3090.

[3] Jones-Wilson, L., Susca, S., Diaz, C., Chang, H., Duffy, E., Effinger, R., Lewis, D., Liewer, K., Lo, K., Ochoa, H., Perez, J., Rizvi, A., Seubert, C., Umsted, C., Borden, M., Clark, P., Massey, R., and Porter, M., "A Sub-Arcsecond Pointing Stability Fine Stage for a High Altitude Balloon Platform," *Proceedings of the 2017 IEEE Aerospace Conference (March 2017, Big Sky, MT)*, 2017, pp. 1–15. https://doi.org/10.1109/AERO.2017.7943590.

[4] Stuchlik, D. W., "The Wallops Arc Second Pointer - A Balloon Borne Fine Pointing System," *Proceedings of the 2015 AIAA Balloon Systems Conference (June 2015, Dallas, TX)*, 2015, pp. 1–15. https://doi.org/10.2514/6.2015-3039.

[5] Stuchlik, D. W., and Lanzi, R., "The NASA Wallops Arc-Second Pointer (WASP) System for Precision Pointing of Scientific Balloon Instruments and Telescopes," *Proceedings of the 2017 AIAA Balloon Systems Conference (June 2017, Denver, CO)*, 2017, pp. 1–8. https://doi.org/10.2514/6.2017-3609.

[6] Kopp, G., Smith, P., Belting, C., Castleman, Z., Drake, G., Espejo, J., Heuerman, K., Lanzi, J., and Stuchlik, D., "Radiometric flight results from the HyperSpectral Imager for Climate Science (HySICS)," *Geoscientific Instrumentation, Methods and Data Systems*, Vol. 6, 2017, pp. 169–191. https://doi.org/10.5194/gi-6-169-2017.

[7] Hurford, T. A., Mandell, A., Reddy, V., and Young, E., "Observatory for Planetary Investigations from the Stratosphere," *Proceedings of the 46th Lunar and Planetary Science Conference (March 2015, The Woodlands, TX)*, 2015, pp. 1–2.

[8] Mendillo, C. B., Hewawasam, K., Howe, G. A., Martel, J., Cook, T. A., and Chakrabarti, S., "The PICTURE-C exoplanetary direct imaging balloon mission: first flight preparation," *Techniques and Instrumentation for Detection of Exoplanets IX, Proceedings of the 2019 International Society for Optics and Photonics Conference (August 2019, San Diego, CA)*, Vol. 11117, SPIE, 2019, pp. 101 – 111. https://doi.org/10.1117/12.2529710.

[9] Mendillo, C. B., Hewawasam, K., Martel, J., Potter, T., Cook, T. A., and Chakrabarti, S., "The PICTURE-C Exoplanetary Imaging Balloon Mission: First Flight Preparation," *Techniques and Instrumentation for Detection of Exoplanets IX, Proceedings of the 2021 International Society for Optics and Photonics Conference (August 2021, San Diego, CA)*, Vol. 11823, SPIE, 2021, pp. 125 – 133. https://doi.org/10.1117/12.2594749.

[10] Gopalswamy, N., Newmark, J., Yashiro, S., Mäkelä, P., Reginald, N., Thakur, N., Gong, Q., Kim, Y. H., Cho, K. S., Choi, S. H., Baek, J. H., Bong, S. C., Yang, H. S., Park, J. Y., Kim, J. H., Park, Y. D., Lee, J. O., Kim, R. S., and Lim, E. K., "The Balloon-Borne Investigation of Temperature and Speed of Electrons in the Corona (BITSE): Mission Description and Preliminary Results," *Solar Physics*, Vol. 296, 2021. https://doi.org/10.1007/s11207-020-01751-8.

[11] Abarr, Q., Awaki, H., Baring, M., Bose, R., De Geronimo, G., Dowkontt, P., Errando, M., Guarino, V., Hattori, K., Hayashida, K., Imazato, F., Ishida, M., Iyer, N., Kislat, F., Kiss, M., Kitaguchi, T., Krawczynski, H., Lisalda, L., Matake, H., Maeda, Y., Matsumoto, H., Mineta, T., Miyazawa, T., Mizuno, T., Okajima, T., Pearce, M., Rauch, B., Ryde, F., Shreves, C., Spooner, S., Stana, T.-A., Takahashi, H., Takeo, M., Tamagawa, T., Tamura, K., Tsunemi, H., Uchida, N., Uchida, Y., West, A., Wulf, E., and Yamamoto, R., "XL-Calibur – a Second-Generation Balloon-Borne Hard X-Ray Polarimetry Mission," *Astroparticle Physics*, Vol. 126, 2021, p. 102529. https://doi.org/10.1016/j.astropartphys.2020.102529.

[12] McCarthy, D. J., "Operating Characteristics of the Stratoscope II Balloon-Borne Telescope," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 5, No. 2, 1969, pp. 323–329. https://doi.org/10.1109/TAES.1969.309922.

[13] Pascale, E., Ade, P. A. R., Bock, J. J., Chapin, E. L., Chung, J., Devlin, M. J., Dicker, S., Griffin, M., Gundersen, J. O., Halpern, M., Hargrave, P. C., Hughes, D. H., Klein, J., MacTavish, C. J., Marsden, G., Martin, P. G., Martin, T. G., Mauskopf, P., Netterfield, C. B., Olmi, L., Patanchon, G., Rex, M., Scott, D., Semisch, C., Thomas, N., Truch, M. D. P., Tucker, C., Tucker, G. S., Viero, M. P., and Wiebe, D. V., "The Balloon-Borne Large Aperture Submillimeter Telescope: BLAST," *The Astrophysical Journal*, Vol. 681, No. 1, 2008, pp. 400–414. https://doi.org/10.1086/588541.

[14] Barthol, P., Gandorfer, A., Solanki, S. K., Schüssler, M., Chares, B., Curdt, W., Deutsch, W., Feller, A., Germerott, D., Grauf, B., Heerlein, K., Hirzberger, J., Kolleck, M., Meller, R., Müller, R., Riethmüller, T. L., Tomasch, G., Knölker, M., Lites, B. W., Card, G., Elmore, D., Fox, J., Lecinski, A., Nelson, P., Summers, R., Watt, A., Martínez Pillet, V., Bonet, J. A., Schmidt, W., Berkefeld, T., Title, A. M., Domingo, V., Gasent Blesa, J. L., del Toro Iniesta, J. C., López Jiménez, A., Álvarez-Herrero, A., Sabau-Graziati, L., Widani, C., Haberler, P., Härtel, K., Kampf, D., Levin, T., Pérez Grande, I., Sanz-Andrés, A., and Schmidt, E., "The Sunrise Mission," *Solar Physics*, Vol. 268, 2011, pp. 1–34. https://doi.org/10.1007/s11207-010-9662-9.

[15] Romualdez, L. J., Damaren, C. J., Li, L., Galloway, M. N., Hartley, J. W., Netterfield, C. B., Clark, P., and Massey, R. J., "Precise Pointing and Stabilization Performance for the Balloon-Borne Imaging Testbed: 2015 Test Flight," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, Vol. 231, No. 4, 2017, pp. 713–727. https://doi.org/10.1177/0954410016641451.

[16] Borden, M., Lewis, D., Ochoa, H., Jones-Wilson, L., Susca, S., Porter, M., Massey, R., Clark, P., and Netterfield, B., "Thermal, Structural, and Optical Analysis of a Balloon-Based Imaging System," *Publications of the Astronomical Society of the Pacific*, Vol. 129, 2017. https://doi.org/10.1088/1538-3873/129/973/035001.

[17] Markley, F. L., and Crassidis, J. L., *Fundamentals of Spacecraft Attitude Determination and Control*, Springer, First Edition, 2014.

[18] Slotine, J. E., and Li, W., *Applied Nonlinear Control*, Prentice Hall, 1995.

[19] Abdo, M., Vali, A. R., Toloei, A., and Arvan, M. R., "Research on the Cross-Coupling of a Two Axes Gimbal System with Dynamic Unbalance," *International Journal of Advanced Robotic Systems*, Vol. 10, 2013, p. 357. https://doi.org/10.5772/56963.

[20] Ekstrand, B., "Equations of Motion for a Two-Axes Gimbal System," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 37, No. 3, 2001, pp. 1083–1091. https://doi.org/10.1109/7.953259.

[21] Liu, S., Lu, T., Shang, T., and Xia, Q., "Dynamic Modeling and Coupling Characteristic Analysis of Two-Axis Rate Gyro Seeker," *International Journal of Aerospace Engineering*, Vol. 2018, 2018. https://doi.org/10.1155/2018/8513684.

[22] Lewis, F. L., Jagannathan, S., and Yesildirek, A., *Neural Network Control of Robot Manipulators and Nonlinear Systems*, Taylor and Francis, 1999.

[23] Kahlil, H. K., *Nonlinear Control*, Pearson, First Edition, 2015.

[24] Darling, J. E., Legrand, K. A., Galchenko, P., Pernicka, H. J., DeMars, K. J., Shirley, A. T., McCabe, J. S., Schmid, C. L., Haberberger, S. J., and Mundahl, A. J., "Development and Flight of a Stereoscopic Imager for Use in Spacecraft Close Proximity Operations," *Proceedings of the 39th Annual AAS Rocky Mountain Section Guidance and Control Conference (February 2016, Breckenridge, CO)*, Vol. 157, 2016, pp. 489–500.

[25] Davis, J., Galchenko, P., Jennings, D., and Pernicka, H. J., "Development and Validation of a GNC Algorithm Using a Stereoscopic Imaging Sensors in Close Proximity Operations," *Proceedings of the 2017 AAS/AIAA Astrodynamics Specialist Conference (August 2017, Stevenson, WA)*, Vol. 162, 2018, pp. 3149–3161.

[26] Galchenko, P., and Pernicka, H. J., "Precision Control of Microsatellite Swarms Using Plasmonic Force Propulsion," *Proceedings of the 2018 AAS/AIAA Astrodynamics Specialist Conference (August 2018, Snowbird, UT)*, Vol. 167, 2018, pp. 935–954.

[27] Jennings, D., Davis, J., Galchenko, P., and Pernicka, H. J., "Validation of a GNC Algorithm Using a Stereoscopic Imaging Sensor to Conduct Close Proximity Operations," *Proceedings of the 41st Annual AAS Rocky Mountain Section Guidance and Control Conference (2018, Breckenridge, CO)*, Vol. 164, 2018, pp. 47–58.

[28] Galchenko, P., and Pernicka, H. J., "Precision Control of Microsatellite Swarms Using Plasmonic Force Propulsion," *Proceedings of the 2018 AAS/AIAA Astrodynamics Specialist Conference (August 2018, Snowbird, UT)*, Vol. 167, 2019, pp. 935–954.

[29] MacKunis, W., Leve, F., Patre, P., Fitz-Coy, N., and Dixon, W., "Adaptive Neural Network-Based Satellite Attitude Control in the Presence of CMG Uncertainty," *Aerospace Science and Technology*, Vol. 54, 2016, pp. 218–228. https://doi.org/10.1016/j.ast.2016.04.022.

[30] Ghafoor, A., Galchenko, P., Balakrishnan, S. N., Pernicka, H., and Yucelen, T., "ETNAC Design Enabling Formation Flight at Liberation Points," *Proceedings of the 2019 American Control Conference (July 2019, Philadelphia, PA)*, 2019, pp. 3689–3694. https://doi.org/10.23919/ACC.2019.8814922.

[31] Galchenko, P., Pernicka, H. J., and Balakrishnan, S. N., "Pointing System Design for the COronal Diagnostic Experiment (CODEX) using a Modified State Observer and a Neural Network Controller," *Proceedings of the 2020 AAS/AIAA Astrodynamics Specialist Conference (August 2020, South Lake Tahoe, CA)*, 2021, pp. 645–660.

[32] Rajagopal, K., Mannava, A., Balakrishnan, S., Nguyen, N., and Krishnakumar, K., "Neuroadaptive Model Following Controller Design for Non-Affine and Non-Square Aircraft Systems," *Proceedings of the 2009 AIAA Guidance, Navigation, and Control Conference (August 2009, Chicago, IL)*, 2009, pp. 1–21. https://doi.org/10.2514/6.2009-5737.

[33] Mannava, A., Balakrishnan, S., Tang, L., and Landers, R., "Optimal Tracking Control of Motion Systems," *IEEE Transactions on Control Systems Technology*, Vol. 20, 2012, pp. 1548–1558. https://doi.org/10.1109/TCST.2011.2168608.

[34] Harl, N., Rajagopal, K., and Balakrishnan, S. N., "Neural Network Based Modified State Observer for Orbit Uncertainty Estimation," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 4, 2013, pp. 1194–1209. https://doi.org/10.2514/1.55711.

II. NEURAL NETWORK ATTITUDE CONTROLLER HARDWARE TESTING FOR THE WALLOPS ARC-SECOND POINTER

Pavel Galchenko[3] and Henry Pernicka[4]
Missouri University of Science and Technology, Rolla, MO, 65409-0050

ABSTRACT

This research implements, tunes, and tests two neural-network-based methodologies to control the pointing performance and stability of the Wallops Arc-Second Pointer, a pointing system suspended from a high altitude balloon platform. The first method uses filtered tracking error and a two-layer neural network to perform nonlinear estimation in the control space for the pointing control system. The second method uses an observer and a one-layer neural network to perform nonlinear estimation outside the controller, and input-output feedback linearization for the pointing control system. The two neural network methods are tested against the heritage proportional-integral-derivative control system over a three month testing campaign on flight hardware in preparation for a science flight. Results show that the two new methodologies can provide improved pointing performance over the heritage control system.

NOMENCLATURE

$\mathbf{x}_1 = \left[\begin{array}{ccc}\theta_\alpha & \theta_\beta & \theta_\gamma\end{array}\right]^\mathrm{T}$	=	roll, pitch, and yaw rotations [rad]
$\mathbf{x}_2 = \left[\begin{array}{ccc}\omega_\alpha & \omega_\beta & \omega_\gamma\end{array}\right]^\mathrm{T}$	=	roll, pitch, and yaw angular velocities [rad/s]
$\mathbf{u} = [\begin{array}{ccc}u_\alpha & u_\beta & u_\gamma\end{array}]^\mathrm{T}$	=	roll, pitch, and yaw control torques [ft-lb]

[3] Ph.D. Candidate, Department of Mechanical and Aerospace Engineering, Missouri University of Science and Technology, 400 W 13th St, Rolla, MO 65409-0050.

[4] Curators' Distinguished Teaching Professor of Aerospace Engineering, Department of Mechanical and Aerospace Engineering, Missouri University of Science and Technology, 400 W 13th St, Rolla, MO 65409-0050.

$\mathbf{f}_{\mathbf{x}}(\mathbf{x}_2)$	=	known system dynamics
$\mathbf{g}_{\mathbf{x}}$	=	control mapping to dynamics
$\boldsymbol{\delta}(\mathbf{x})$	=	known disturbance dynamics
J	=	system inertia matrix [slug-ft^2]
$\mathbf{x}_d = [\ \mathbf{x}_{d1} \quad \mathbf{x}_{d2}\ ]^T$	=	desired orientation [rad] and angular rates [rad/s]
$\mathbf{f}_d$	=	desired dynamics
$\mathbf{e} = [\ \mathbf{e}_1 \quad \mathbf{e}_2\ ]^T$	=	error states
A, B	=	known linear dynamics matrices
K	=	linear control gain matrix
A_{CL}	=	closed-loop dynamics
K_p	=	proportional [ft-lb/rad] gain
K_i	=	integral [ft-lb/(rad-sec)] gain
K_d	=	derivative [ft-lb/(rad/sec)] gains
ω_n	=	controller natural frequency [rad/s]
ζ	=	damping ratio
α	=	tuning parameter
$\mathcal{F}_{\mathbf{x}}, \hat{\mathcal{F}}_{\mathbf{x}}$	=	true and estimated dynamics
$\boldsymbol{\Delta}(\mathbf{x})$	=	unmodeled perturbations/dynamics
σ, σ'	=	activation function and its derivative
$\phi\,(\mathbf{z})$	=	basis vector
H_β, H_γ	=	pitch/yaw angles from gimbal hubs (rad)
ε	=	bounded neural network estimation error
$W^T, V^T, \hat{W}^T, \hat{V}^T$	=	true and estimated neural network weighting matrices
$\tilde{W}, \tilde{V}$	=	neural network weight matrices estimation error
$\mathbf{r}$	=	filtered tracking error
λ	=	gain coefficient
K_v, K_z	=	controller gain matrices

$\mathbf{u}_r$	=	robustifying control
$\tilde{\mathcal{F}}_{\mathbf{x}}$	=	dynamics estimation error
M, N	=	neural network adaptation gain matrices
κ	=	design parameter
L	=	Lyapunov function
$tr\{\}$	=	trace operator
K_2	=	linear observer gain matrix
$\mathbf{e}_a$	=	observer estimation error
Γ	=	observer neural network adaptation rate
B_w	=	system bandwidth (Hz)

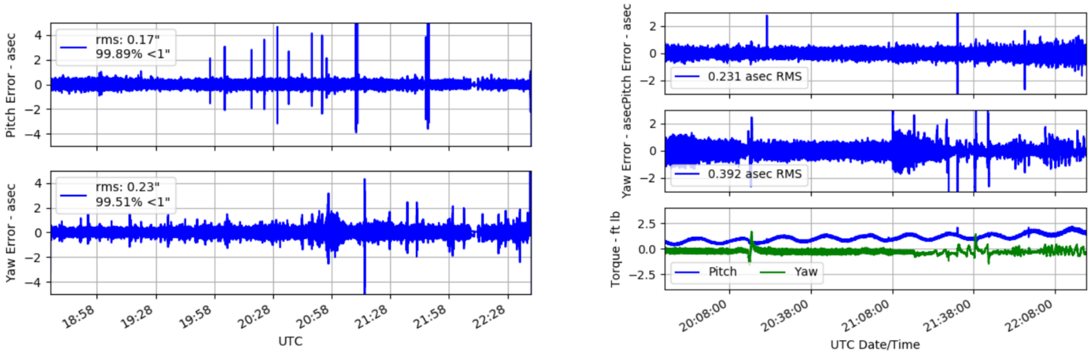
1. INTRODUCTION

The Wallops Arc-Second Pointer (WASP) is a pointing system platform, developed by National Aeronautics and Space Administration's (NASA) Wallops Flight Facility (WFF), part of the Goddard Space Flight Center (GSFC), for high-altitude balloon missions operated by the Balloon Program Office (BPO) [1–3]. High-altitude balloons provide scientists with a platform to conduct novel science in the Earth's upper atmosphere, with float altitudes typically above 100,000 ft [4]. This offers scientists a space-like environment, being above 99.5% of the Earth's atmosphere (by mass), while providing significant reductions in cost compared to traditional spaceflight missions. The WASP platform is capable of accurately pointing scientific instruments with sub arcsecond accuracy and stability, enabling scientific observations for multiple science disciplines such as Heliophysics, Earth Sciences, Astrophysics, and Planetary Sciences.

The WASP platform completed two successful test flights between 2011 and 2012 and began science operations in 2013. Over the ten-plus years of operations of the WASP platform, the team has conducted flights for seven science missions and has continuously demonstrated performance in the arcsecond to sub-arcsecond regime [5–10]. During the 2019 campaign, the WASP platform was able to maintain a 0.286 arcsecond total root-mean-square (RMS) error over a four hour period of solar observations for the BITSE mission, and a 0.455 arcsecond total RMS error for PICTURE-C during a two hour observation of α Lyr (Vega), the brightest star in the Lyra constellation, as seen in Figure 1. To significantly improve the WASP pointing performance, hardware modifications are required to increase sensing precision and actuator disturbance reduction (such as lowering the high frequency vibrations generated by the shaft rotation motors). Another viable option is to determine if pointing performance could be improved through software modifications such as improved filtering and/or different control algorithms. During the 2019 flight campaign, the WASP system would occasionally experience brief excursions exceeding 1.0 arcseconds and some overall variance in yaw pointing performance throughout the mission. While excursions were quickly captured and brought back toward the average RMS error, these excursions could negatively impact the performance of certain science instruments such as imaging instruments that need lengthy observation periods. This work builds on previous work suggesting to augment the WASP platform with the addition of a neural network control system, which can be activated in lieu of the traditional proportional-integral-derivative (PID) system. Specifically, two neural network algorithms have been implemented into the WASP flight software and hardware testing was completed during a three-month testing campaign in preparation for a second science flight of the PICTURE-C mission.

1.1. WASP DESIGN AND HISTORY

The WASP system is built around an external gondola structure and is capable of housing various science instruments as a centerbody. The gondola is suspended from a rotator, which provides initial targeting, coarse azimuth tracking, and stabilization for the external gondola. The entire system is suspended from the high-altitude balloon with approximately 100 meters of suspension rigging (cable ladder and parachute) between the



(a) BITSE In-Flight Performance

(b) PICTURE-C In-Flight Performance

Figure 1. The performance of the WASP pointing system is presented for the BITSE (a) and PICTURE-C (b) missions, where sub-arcsecond pointing is achieved for long observation periods.

rotator and the balloon. The WASP gondola and the suspended PICTURE-C science mission is shown in Figure 2. The outer gimbal frame is fixed to the external gondola structure and is connected to the inner gimbal frame with a pitch motor and resolver hub pair to provide elevation control. The inner gimbal frame connects to the science centerbody structure through a yaw motor and resolver hub pair and provides fine azimuth control. This design is similar to a gimbaled inertial platform, with pitch-yaw articulated axes. To minimize cross-coupling terms in the dynamics, the rotator provides coarse azimuth pointing of the external gondola such that the centerbody is within 1.0 degrees of the inner gimbal frame, minimizing the disturbances experienced in the fine yaw control channel. This also provides full 360 degree azimuth range for the science body, providing a large range of science targets. The motor hubs include large diameter brushless direct-current (BLDC) torque motors to drive the system and utilize rotating shafts to eliminate static friction in the bearings. The shafts are counterrotated in the hub pairs to minimize residual kinetic friction. The attitude is determined by integrating angular rates provided by a Northrop Grumman LN251, a fiber-optic gyro-based inertial measurement unit. Absolute pointing information is provided from a custom star tracker camera developed at WFF and a six-state extended Kalman filter is used to provide quaternion state data to the control system. A modified PID control law is used to compute control torques to the pitch/yaw motor hubs to provide sub-arcsecond pointing for the centerbody.

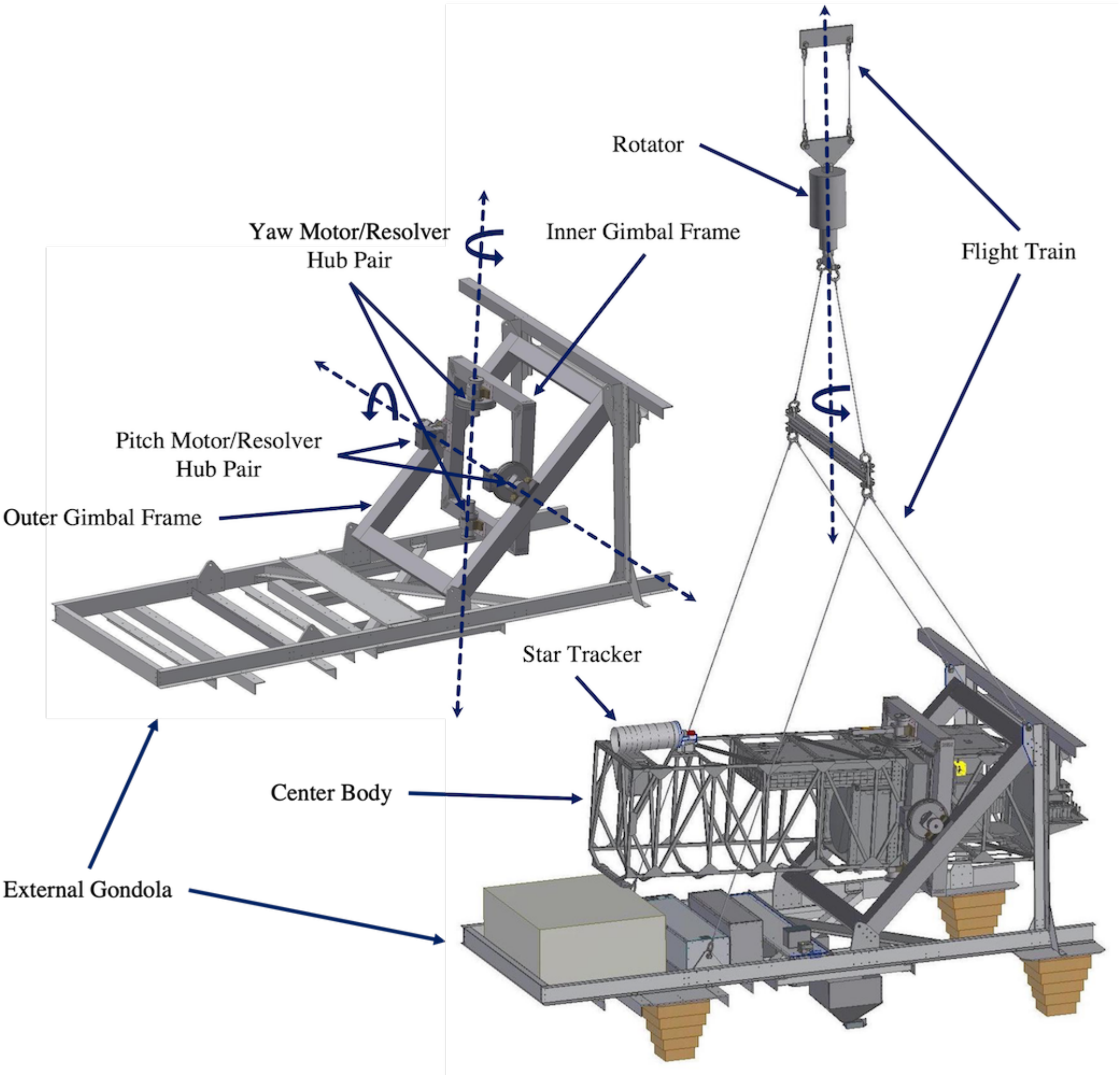


Figure 2. The centerbody (science instrument) is attached to the inner (yaw) and outer (pitch) gimbal frames such that the pitch-yaw motor controllers provide fine elevation and azimuth pointing, while the rotator provides coarse azimuth pointing of the external gondola structure and keeps the centerbody within 1.0 degrees of the inner gimbal frame.

1.2. SCHOLARLY CONTRIBUTIONS

This study offers several contributions to the nonlinear control field as relates to gimbaled pointing systems:

1. SNNARC, a two-layer neural network controller, is implemented, tuned, and tested on the flight processor and hardware in a flight-like environment for the gimbaled control system on the WASP platform.
2. MSO, a one-layer neural network observer and an input-output feedback linearization controller, is implemented, tuned, and tested on the flight processor and hardware in a flight-like environment for the gimbaled control system on the WASP platform.
3. Lessons learned in the implementation and tuning of the two neural network methods on flight hardware are presented and documented for further development and for use with future WASP (and similar) platform science mission flights.
4. The results from the two neural network methods show that improved pointing performance is achieved, as compared to the heritage control system, for the gimbaled control system on the WASP platform.

1.3. ORGANIZATION

This paper is organized as follows. In Section 2, a brief background on the dynamics and control theory is presented, as well as a description of the heritage controller design implemented for the WASP system. Section 3 introduces neural networks and the two neural network control methodologies tested on the WASP hardware. Section 4 discusses the tuning and debugging process of implementing the neural network control algorithms with the flight software and hardware. Section 5 presents final results from hardware testing in a flight-like environment, as well as some edge case testing. Section 6 presents future work, including a brief discussion of implementing the neural networks for the next WASP mission. Lastly, Section 7 gives an overview of the results and makes concluding remarks.

2. BACKGROUND

While a variety of models exist to describe the dynamics of two-axis gimbal systems [11–14], the typical unconstrained second-order three degrees of freedom attitude dynamics model [15] is presented here using a two-vector state system given as

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \mathbf{x}_2 \\ \dot{\mathbf{x}}_2 &= f_{\mathbf{x}}(\mathbf{x}_2) + \delta(\mathbf{x}) + g_{\mathbf{x}}\mathbf{u}, \end{aligned} \tag{1}$$

where $f_{\mathbf{x}}(\mathbf{x}_2) = -J^{-1}[\mathbf{x}_2\times]J\mathbf{x}_2$, where J is the inertia matrix (assumed constant) of the (rigid) body with respect to the center of mass and in terms of the body-fixed axes, $\delta(\mathbf{x})$ is some known disturbance dynamics, $g_{\mathbf{x}}$ maps the controller dynamics to the system, and $\mathbf{u}$ is the applied torque, i.e. the control input. The state vectors, $\mathbf{x}_1 = \begin{bmatrix} \theta_\alpha & \theta_\beta & \theta_\gamma \end{bmatrix}^{\mathrm{T}}$ and $\mathbf{x}_2 = \begin{bmatrix} \omega_\alpha & \omega_\beta & \omega_\gamma \end{bmatrix}^{\mathrm{T}}$, are the rotation and angular rate vectors, respectively, of the body with respect to the inertial frame, where θ_α, θ_β, θ_γ are rotations about primary orthogonal axes and ω_α, ω_β, ω_γ are the angular velocities about these respective axes. The control input is given as $\mathbf{u} = [\, u_\alpha \quad u_\beta \quad u_\gamma \,]^{\mathrm{T}}$. To simplify the formulation of the dynamics into a general form, J^{-1} is embedded into the disturbance dynamics $\delta(\mathbf{x})$ and the mapping of the controller dynamics, $g_{\mathbf{x}}$, in Eq. (1).

This formulation is compatible with the other dynamics models presented in literature [11–14] with variations in the definitions of $f_{\mathbf{x}}(\mathbf{x}_2)$ and $\delta(\mathbf{x})$. Utilizing a pitch-yaw gimbal and a rotator to keep the centerbody centered in the gimbal frames reduces cross-coupling terms in the gimbal dynamics, allowing for the more simplified model approach. Furthermore, due to the large mass and volume of the WASP platform and science instruments, only approximations of the total system inertia values can be made. Fine balancing of the centerbody is achieved through the manual addition of ballast masses at various locations of the structure. To compensate for model inaccuracies and static imbalances, the heritage system adds integrator action to the control system. The new methodology introduced in this research uses neural networks to estimate the dynamics while making no assumptions on the details of $f_{\mathbf{x}}(\mathbf{x}_2)$ and $\delta(\mathbf{x})$ and only assumes the dynamics can be described by the form given in Eq. (1).

2.1. CONTROLLER BASELINE

To develop a control algorithm, the dynamics are rewritten in terms of the error between the current and desired reference system states, where the desired states, $\mathbf{x}_d$, are defined as $\mathbf{x}_d = [\ \mathbf{x}_{d1} \quad \mathbf{x}_{d2}\ ]^T$, where $\mathbf{x}_{d1}$ is a vector of desired rotations and $\mathbf{x}_{d2}$ is a vector of desired angular rates. The reference system dynamics are given as $\dot{\mathbf{x}}_{d1} = \mathbf{x}_{d2}$ and $\dot{\mathbf{x}}_{d2} = f_d$, where f_d can be a set of desired dynamics for the reference system. Note that for a regulatory reference system (such as tracking an inertial stellar target for the PICTURE-C mission), $\mathbf{x}_{d2} = 0$ and $f_d = 0$. The error states are now defined as $\mathbf{e} = [\ \mathbf{e}_1 \quad \mathbf{e}_2\ ]^T$, where $\mathbf{e}_1 = \mathbf{x}_1 - \mathbf{x}_{d1}$ and $\mathbf{e}_2 = \mathbf{x}_2 - \mathbf{x}_{d2}$, such that taking the derivative of the error vectors gives $\dot{\mathbf{e}} = [\ \dot{\mathbf{e}}_1 \quad \dot{\mathbf{e}}_2\ ]^T$ with

$$\begin{aligned} \dot{\mathbf{e}}_1 &= \mathbf{e}_2 \\ \dot{\mathbf{e}}_2 &= f_{\mathbf{x}}(\mathbf{x}) - f_d + \delta(\mathbf{x}) + g_{\mathbf{x}}\mathbf{u}, \end{aligned} \tag{2}$$

defining the error dynamics used for the various controller formulations.

It is known that a linear closed-loop system can be shown to be asymptotically stable if the closed-loop matrix is Hurwitz [16]. By using input-output feedback linearizaiton [16] and carefully designing a nonlinear controller in terms of the error dynamics in Eq. (2), it is possible to make the nonlinear system behave in a linear manner. A controller is chosen such that $\mathbf{u} = \frac{1}{g_{\mathbf{x}}}\left(-f_x(\mathbf{x}) + f_d - \delta(\mathbf{x}) - \mathbf{K}_p\mathbf{e}_1 - \mathbf{K}_d\mathbf{e}_2\right)$, where $\mathbf{K}_p$ and $\mathbf{K}_d$ are proportional and derivative controller gain vectors, respectively, and $|g_{\mathbf{x}}| > 0$. The closed-loop error dynamics in Eq. (2) are reduced to $\dot{\mathbf{e}} = A_{cl}\,\mathbf{e}$, where $A_{cl} = \begin{bmatrix} 0 & 1 \\ -\mathbf{K}_p & -\mathbf{K}_d \end{bmatrix}$, such that the origin of the unperturbed nonlinear system will be asymptotically stable if A_{cl} is Hurwitz. This method of nonlinear control is the basis from which the neural network controllers are designed.

2.2. PID CONTROLLER DESIGN

The heritage control system for the WASP platform uses a PID controller and assumes that the inertia of the system is diagonal, where J_i is the inertia of specified axis, decoupling the dynamics of the pitch/yaw channels. The system can then be described by the plant $P(s) = J_i^{-1}/s^2$, where by choosing a PID control input, the closed-loop characteristic equation is given by $s^3 + [J_i^{-1}K_d]s^2 + [J_i^{-1}K_p]s + [J_i^{-1}K_i]$, where K_p, K_i, and K_d are proportional, integrator, and derivative gains, respectively. Using the typical (ζ, ω_n) parametrization and fixing the third pole at $-\alpha\omega_n$, the desired closed-loop characteristic equation is given by $s^3 + [(2\zeta + \alpha)\omega_n]s^2 + [(2\zeta\alpha + 1)\omega_n^2]s + [\alpha\omega_n^3]$ where ω_n is the natural frequency of the system, ζ is the damping ratio, and α is a design parameter. The controller gains can now be selected to match the design characteristic equation with

$$K_p = \frac{(2\zeta\alpha + 1)\omega_n^2}{J_i^{-1}} \qquad K_i = \frac{\alpha\omega_n^3}{J_i^{-1}} \qquad K_d = \frac{(2\zeta + \alpha)\omega_n}{J_i^{-1}} \tag{3}$$

where by fixing the design parameter at $\alpha = 1$, the values for ζ and ω_n are tuned to give desired performance characteristics. It should be noted that for $\alpha = 0$, the gain tuning becomes that of a proportional-derivative (PD) controller for the same system.

3. NEURAL NETWORK DESIGN

In formulating the input-output feedback linearization controller, it was assumed that the model of $f_x(\mathbf{x})$ is "truth" and that $\delta(\mathbf{x})$ describes all other contributing disturbance dynamics to the system. As errors are inherent in all models and all disturbances cannot be accounted for, consideration for unmodeled dynamics must be made. While controlling nonlinear systems with unmodeled dynamics is an extensively studied field [17–24], this research focuses on the application of neural networks as they offer some unique advantages over other adaptive control methods.

The true dynamics can be given as $\mathcal{F}_{\mathbf{x}} = f_x(\mathbf{x}) + \delta(\mathbf{x}) + \Delta(\mathbf{x})$, where $f_x(\mathbf{x})$ and $\delta(\mathbf{x})$ are as described in Eq. (1) and $\Delta(\mathbf{x})$ is a set of all other unmodeled dynamics. Using neural networks the true dynamics can be represented by $\mathcal{F}_{\mathbf{x}} = W^T\phi(\mathbf{z}) + \varepsilon$ for a one-layer network and $\mathcal{F}_{\mathbf{x}} = W^T\sigma\left(V^T\phi(\mathbf{z})\right) + \varepsilon$ for a two-layer network, where σ is some activation function, $\phi(\mathbf{z})$ is the basis vector (a vector of inputs into the neural network), and W^T and V^T are the true weights that give the value of $\mathcal{F}_{\mathbf{x}}$ with some bounded approximation error, ε, through the universal function approximation properties of neural networks [25]. As the true weighting matrices are unknown, the estimated dynamics, $\hat{\mathcal{F}}_{\mathbf{x}}$, are given as $\hat{\mathcal{F}}_{\mathbf{x}} = \hat{W}^T\phi(\mathbf{z})$ for a one-layer network and $\hat{\mathcal{F}}_{\mathbf{x}} = \hat{W}^T\sigma\left(\hat{V}^T\phi(\mathbf{z})\right)$ for a two-layer network, where $\hat{W}^T$ and $\hat{V}^T$ are the approximated weights.

The two-vector state systems from Eq. (1) are rewritten as

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \mathbf{x}_2 \\ \dot{\mathbf{x}}_2 &= \mathcal{F}_{\mathbf{x}} + g_{\mathbf{x}}\mathbf{u} \end{aligned} \tag{4}$$

and the error dynamics in Eq. (2) as

$$\begin{aligned} \dot{\mathbf{e}}_1 &= \mathbf{e}_2 \\ \dot{\mathbf{e}}_2 &= \mathcal{F}_{\mathbf{x}} - f_d + g_{\mathbf{x}}\mathbf{u} \end{aligned} \tag{5}$$

forming the foundation of the two neural network methodologies studied in this research. The first method operates in terms of the error dynamics, Eq. (5), and as such the neural network learning occurs in the control space. The second method introduces an observer for the state dynamics, Eq. (4), and as such the neural network learning occurs outside the control space. Both methods can be described by a Lyapunov candidate function and are shown to be Ultimately Upper Bounded (UUB), demonstrating Lyapunov stability.

3.1. SNNARC FORMULATION

The first method, originally developed for robotic manipulators [25], is the Sub-arcsecond Neural Network Attitude Reference Controller (SNNARC). The error dynamics from Eq. (5) are written in terms of filtered tracking error, $\mathbf{r}$, such that $\mathbf{r} = \mathbf{e}_2 + \lambda \mathbf{e}_1$ and $\dot{\mathbf{r}} = \mathcal{F}_{\mathbf{x}} - f_d + \lambda \mathbf{e}_2 + g_{\mathbf{x}} \mathbf{u}$, where λ is some positive gain coefficient and $\mathbf{r}$ is now a vector of scalar errors for each control axis. It is now possible to use the estimated dynamics, $\hat{\mathcal{F}}_{\mathbf{x}}$, in the controller design such that

$$\mathbf{u} = \frac{1}{g_{\mathbf{x}}} \left(-\hat{\mathcal{F}}_{\mathbf{x}} + f_d - K_v \mathbf{r} - \lambda \mathbf{e}_2 + \mathbf{u}_r \right) \quad \text{where} \quad \mathbf{u}_r = -K_z \left(\left\| \hat{\Theta} \right\| + \Theta_m \right) \mathbf{r} \tag{6}$$

where K_v and K_z are some positive controller gain matrices, $|g_{\mathbf{x}}| > 0$, and the weighting matrices are defined as $\Theta = \begin{bmatrix} V & 0 \\ 0 & W \end{bmatrix}$, $\hat{\Theta} = \begin{bmatrix} \hat{V} & 0 \\ 0 & \hat{W} \end{bmatrix}$, and $\|\Theta\| \le \Theta_m$. Using the control from Eq. (6) in the filtered tracking error, $\mathbf{r}$, the closed-loop system becomes

$$\dot{\mathbf{r}} = -K_v \mathbf{r} + \tilde{\mathcal{F}}_{\mathbf{x}} + \mathbf{u}_r \tag{7}$$

where $\tilde{\mathcal{F}}_{\mathbf{x}}$ is the estimation error defined by $\tilde{\mathcal{F}}_{\mathbf{x}} = \mathcal{F}_{\mathbf{x}} - \hat{\mathcal{F}}_{\mathbf{x}}$. Neural network weight update laws are selected as

$$\begin{aligned} \dot{\hat{W}} &= M \left(\hat{\sigma} - \hat{\sigma}' \hat{V}^T \phi\left(\mathbf{z}\right) \right) \mathbf{r}^T - \kappa \left\| \mathbf{r} \right\| M \hat{W} \\ \dot{\hat{V}} &= N \phi\left(\mathbf{z}\right) \mathbf{r}^T \hat{W}^T \hat{\sigma}' - \kappa \left\| \mathbf{r} \right\| N \hat{V} \end{aligned} \tag{8}$$

where M and N are positive definite matrices that give the learning rate, $\hat{\sigma} = \sigma\left(\hat{V}^T\phi(\mathbf{z})\right)$, $\hat{\sigma}' = \sigma'\left(\hat{V}^T\phi(\mathbf{z})\right)$, and κ is a modification factor such that $\kappa > 0$.

To prove the stability of the system, the candidate Lyapunov function is selected as $L = \frac{1}{2}\mathbf{r}^T\mathbf{r} + \frac{1}{2}tr\left\{\tilde{W}^T M^{-1}\tilde{W}\right\} + \frac{1}{2}tr\left\{\tilde{V}^T N^{-1}\tilde{V}\right\}$, where $tr\{\}$ is the trace operator. By taking the time derivative of the Lyapunov function and using the definitions of $\mathbf{r}$ and the closed-loop form of $\dot{\mathbf{r}}$ in Eq. (7), and the weight update law in Eq. (8), stability in a compact set about the origin can be shown, demonstrating that the Lyapunov function is UUB in both $\|\mathbf{r}\|$ and $\left\|\tilde{\Theta}\right\|$ and thus showing Lyapunov stability of the proposed SNNARC method [24].

3.2. MSO FORMULATION

The second method, originally developed for estimating unmodeled dynamics [26–28], is the Modified State Observer (MSO). The method utilizes an observer to estimate the dynamics and then uses the estimated dynamics in an input-output feedback linearization controller. First the dynamics in Eq. (4) are reduced to a single vector system as estimation occurs only in the angular rate states such that $\dot{\mathbf{x}} = \mathcal{F}_{\mathbf{x}} + g_{\mathbf{x}}\mathbf{u}$, where $\dot{\mathbf{x}}$ is the derivative of the angular rates ($\dot{\mathbf{x}}_2$ from the previous formulation). For this system, an observer can be designed as

$$\dot{\hat{\mathbf{x}}} = \hat{\mathcal{F}}_{\mathbf{x}} + g_{\mathbf{x}}\mathbf{u} + K_2\left(\mathbf{x} - \hat{\mathbf{x}}\right) \tag{9}$$

where $\hat{\mathbf{x}}$ is the estimated state and K_2 is some gain matrix. The dynamics, $\hat{\mathcal{F}}_{\mathbf{x}}$, can be approximated by a one-layer neural network in a manner similar to SNNARC. The state and the observer in Eq. (9) are rewritten in terms of observer error, $\mathbf{e}_a = \mathbf{x} - \hat{\mathbf{x}}$, such that the observer error dynamics are given as

$$\dot{\mathbf{e}}_a = -K_2\mathbf{e}_a + \tilde{W}^T\phi(\mathbf{z}) + \varepsilon \tag{10}$$

where $\phi\left(\mathbf{z}\right)$ is the basis vector and $\tilde{W}$ is the error between the true weights and estimated weights such that $\tilde{W} = W - \hat{W}$. The neural network weight update law is given as

$$\dot{\hat{W}} = \Gamma\phi\left(\mathbf{z}\right)\mathbf{e}_a^T - \kappa\left\|\mathbf{e}_a\right\|\Gamma\hat{W} \tag{11}$$

where Γ is the learning rate and κ is the modification factor, which bound the weights and provides robustness. The stability of the system in Eq. (10) can be shown with the candidate Lyapunov function $L = \frac{1}{2}\mathbf{e}_a^T\mathbf{e}_a + \frac{1}{2}tr\left\{\tilde{W}^T\Gamma^{-1}\tilde{W}\right\}$, where $tr\left\{\right\}$ is the trace operator. By taking the time derivative of the Lyapunov function and using the definitions of $\mathbf{e}_a$ and $\dot{\mathbf{e}}_a$ in Eq. (10), and the weight update law in Eq. (11), stability in a compact set about the origin can be shown, demonstrating that the Lyapunov function is UUB in both $\left\|\mathbf{e}_a\right\|$ and $\left\|\tilde{W}\right\|$ and thus shows Lyapunov stability of the observer in the proposed MSO method [24].

Once the nonlinear dynamics, $\hat{\mathcal{F}}_{\mathbf{x}}$, are estimated by the neural network, an input-output feedback linearization controller given by

$$\mathbf{u} = \frac{1}{g_{\mathbf{x}}}\left(-\hat{\mathcal{F}}_{\mathbf{x}} + f_d - \mathbf{K}_p\mathbf{e}_1 - \mathbf{K}_d\mathbf{e}_2\right) \tag{12}$$

can be applied to the two-vector system in Eq. (5) such that the closed-loop system dynamics become

$$\begin{aligned} \dot{\mathbf{e}}_1 &= \mathbf{e}_2 \\ \dot{\mathbf{e}}_2 &= -\mathbf{K}_p\mathbf{e}_1 - \mathbf{K}_d\mathbf{e}_2 + \tilde{W}^T\phi\left(\mathbf{z}\right) + \varepsilon \end{aligned} \tag{13}$$

where

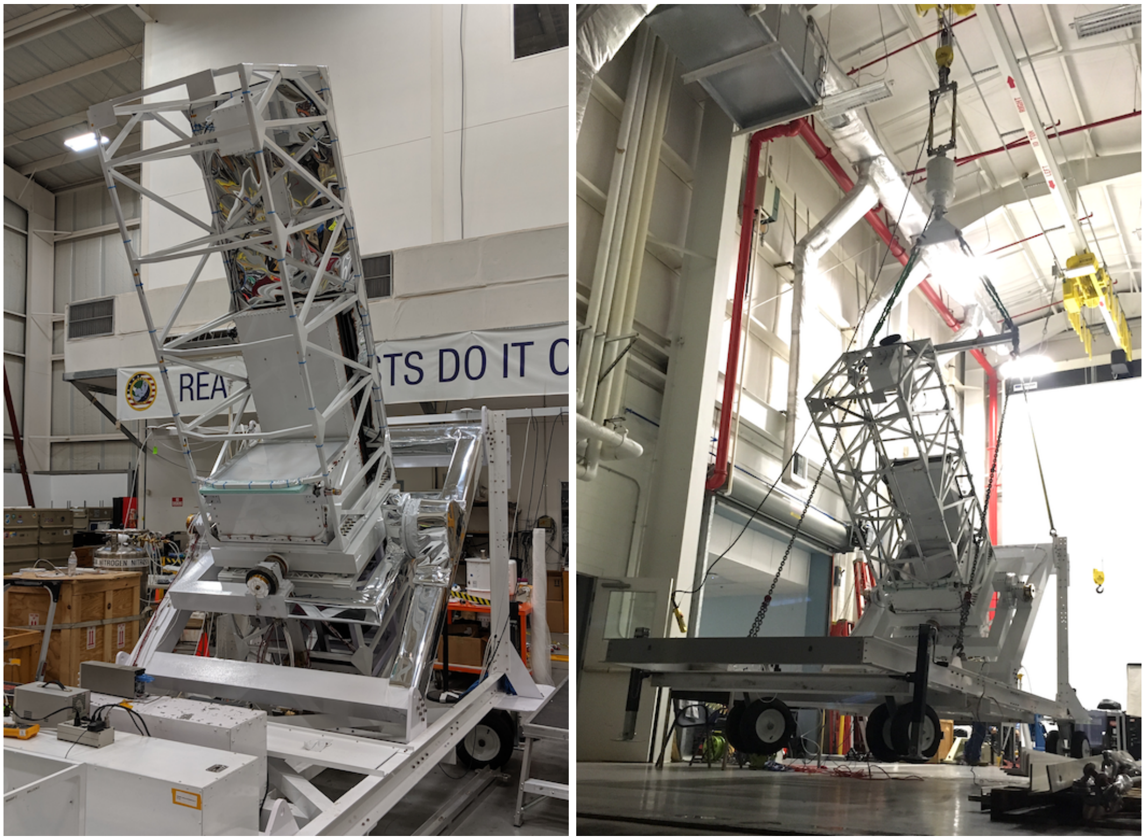
$$A_{cl} = \begin{bmatrix} 0 & 1 \\ -\mathbf{K}_p & -\mathbf{K}_d \end{bmatrix} \qquad \text{and} \qquad d = \begin{bmatrix} 0 \\ \tilde{W}^T\phi\left(\mathbf{z}\right) + \varepsilon \end{bmatrix}$$

and where d is bounded by $\|d\| \leq \|d_m\|$. By using the candidate Lyapunov function $L = \frac{1}{2}\mathbf{e}^T P\mathbf{e}$ and taking its time derivative using the definition of $\dot{\mathbf{e}}$ in Eq. (13) and where P satisfies the equation $A^T P + PA = -Q$, stability in a compact set about the origin can be shown if $\|\mathbf{e}\| \geq 2\frac{\lambda_{max}(P)d_m}{\lambda_{min}(Q)}$, demonstrating that both Lyapunov functions are UUB in $\|\mathbf{e}\|$, $\|\mathbf{e}_a\|$, and $\left\|\tilde{W}\right\|$ and thus shows Lyapunov stability of the proposed MSO method.

4. WASP HARDWARE IMPLEMENTATION

The initial verification and validation of the neural network algorithms for the pointing system were performed using high fidelity simulations through software-in-the-loop (SWIL) testing, where results showed significant improvements in performance as compared to the heritage control system. The focus of this paper is on the three-month hardware-in-the-loop (HWIL) testing campaign in preparation for science flight operations originally scheduled for Fall 2021. Due to unfavorable weather conditions during the short launch period and launch window, the science launch has been rescheduled for Fall 2022.

Hardware testing is characterised by two different configurations. The first configuration, ground testing, is conducted with the WASP platform being seated on the ground on either tires or jack stands. In this configuration, rotator and balloon flight train dynamics are not present in the environment and the system is reacting against the ground. This configuration is where the majority of testing was completed. The second configuration, suspended system flight functional, involves suspending the rotator and the WASP platform from a crane and lifting the entire system above the ground. This configuration results in a flight-like configuration with the system reacting against the inertia of the flight train. Due to the personnel required to perform lifting operations, this configuration is only tested a few times when the majority of systems are flight ready. In both of these configurations, the external environmental dynamics are considered heightened as compared to in-flight. Due to the low atmospheric density at float, the external dynamics are significantly reduced compared to ground. As a point of reference, it is typically not possible to point the WASP system outdoors due to even low level winds overpowering the torque authority of the pitch/yaw motor hubs. Suspending the entire WASP system above the ground, as in the second configuration, does provide significant changes to the dynamics, however, and closely resembles the anticipated dynamics in flight. The rotator maintains the centerbody centered within the external gondola structure while external disturbances and internal vibrations lead to a swaying motion of the entire system, thereby introducing roll angular velocities into the dynamics. The two configurations are illustrated in Figure 3 for the PICTURE-C mission.



(a) Ground Testing

(b) Suspended System Flight Functional

Figure 3. The WASP platform with the PICTURE-C mission science hardware mounted as the centerbody is shown in the ground testing configuration (a) at CSBF, and in suspended system flight functional configuration (b) at NASA WFF.

The three-month testing campaign was conducted from July through September, in preparation for a Fall 2021 launch opportunity from Fort Sumner, New Mexico. As such, tests were distributed over a period of three months between the NASA WFF location in Virginia and the Columbia Science Balloon Facility (CSBF) location in New Mexico. Data were selected from ten different test dates for analysis, and a subset of that data is presented here to demonstrate the implementation, tuning, and performance of the two neural network algorithms. Due to the distribution of tests, the performance on each test date is dependent on various and differing environmental conditions and only a general interpretation of results and conclusions could be made between test dates. Comparisons of specific tuning parameters are given for the same test dates, and several test dates compared the three control methods directly, allowing for more direct comparisons. Test date, location, and configuration are called out in the WASP Hardware Results section to clarify the discussion.

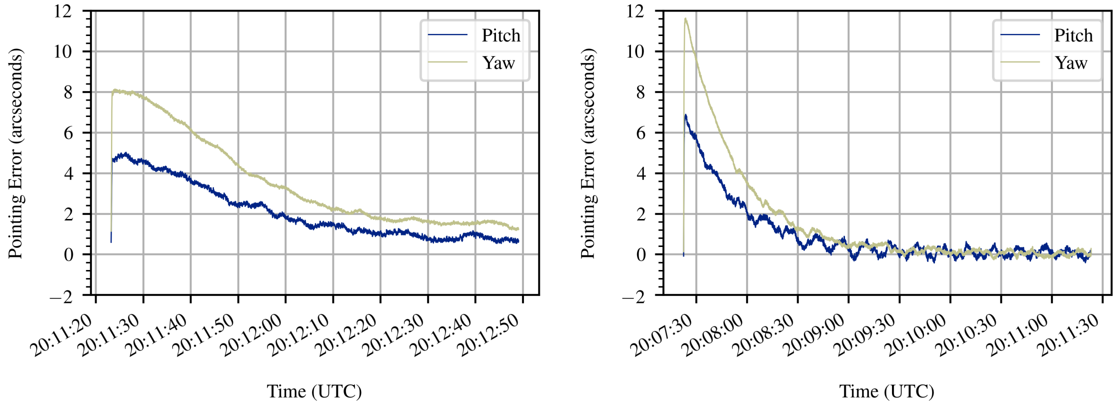
4.1. TUNING PARAMETERS

For the heritage PID system, tuning is primarily carried out through the selection of a closed-loop system bandwidth and damping ratio. The WASP platform bandwidth is approximated as $B_w = \frac{\omega_n}{2\pi}$, where the bandwidth is given in the frequency space (H_z). By default, the typical bandwidth is set to $B_w = 0.4\ H_z$ and the damping ratio is set to $\zeta = 0.9$. The inertia values were assumed to be $J_1 = 92$, $J_2 = 572$, and $J_3 = 554$ in slug-ft^2, corresponding to roll, pitch, and yaw axes, respectively. The final K_p, K_i, and K_d gains are then calculated using Equation (3). For MSO, the same equation is used but with the design parameter set to $\alpha = 0$, such that the K_p and K_d values in Equation (12) are given as $K_p = \frac{\omega_n^2}{J_i^{-1}}$ and $K_d = \frac{2\zeta\omega_n}{J_i^{-1}}$. For the SNNARC method, a relationship was formed between the linear gains K_p/K_d of the MSO method and the K_v/λ values of the filtered tracking error in Equation (6). The final tuning is given by two solutions: $K_v = \omega_n\zeta + \sqrt{\zeta^2 - 1}$ and $\lambda = \omega_n\zeta - \sqrt{\zeta^2 - 1}$ for $\zeta \geq 1$ and $K_v = \lambda = \omega_n\zeta$ for when $\zeta < 1$. As such, for all three control algorithms, the linear (and integral) portion of the controllers is tuned directly by setting the desired bandwidth, B_w, and damping ratio, ζ. In terms of directly comparing K_p/K_d (or equivalent) values between the MSO/SNNARC and PID methods, it was found that the values would be approximately equal at double the PID bandwidth (i.e. $K_{p,\text{ MSO/SNNARC}} \approx K_{p,\text{ PID}}$ when $B_{w,\text{ MSO/SNNARC}} = 2B_{w,\text{ PID}}$).

For the neural networks, the final basis vector, $\phi(\mathbf{z})$, was selected to be $\phi(\mathbf{z}) = [\theta_{e\beta}, \theta_{e\gamma}, \omega_\alpha, \omega_\beta, \omega_\gamma, H_\beta, H_\gamma]^T$ where α corresponds to the roll channel, β corresponds to the pitch channel, γ corresponds to the yaw channel, and $H_{\beta,\gamma}$ are the gimbal angles within the motor pitch/yaw hubs. The neural network weights for $\hat{W}$ (MSO/SNNARC) and $\hat{V}$ (SNNARC) were initialized as null matrices. For MSO, the learning rate is governed by Γ whereas for SNNARC the learning rate is governed by M and N. Both neural networks have a learning rate modification factor given by κ. Beyond these similarities, the remaining tuning parameters have no corollary between the various methods. For MSO, the only other tuning parameter is K_2, which governs the linear observer gain. For SNNARC, there is an additional robustifying controller portion with tuning parameters K_z and Θ_m.

4.2. INITIAL CONTROLLER CONVERGENCE BASED ON LEARNING RATE AND MODIFICATION FACTOR SELECTION

Initial testing of the two neural network algorithms began in July 2021 with ground tests at NASA WFF. As Figure 4 indicates, a ground test from July 16, 2021 showed that the initial configuration of the neural networks resulted in very slow convergence rates between the two methods. In both tests, the heritage PID system was first used to point the system to an inertial target and once steady state was achieved, the controller was switched to either MSO or SNNARC. The MSO test was terminated after approximately 1.5 minutes as it appeared that the system was converging with a steady state bias. In the SNNARC test, it was observed that steady state behavior was achieved after approximately three minutes, after which a total root-mean-square (RMS) error value of 0.223 arcseconds was achieved at steady state.



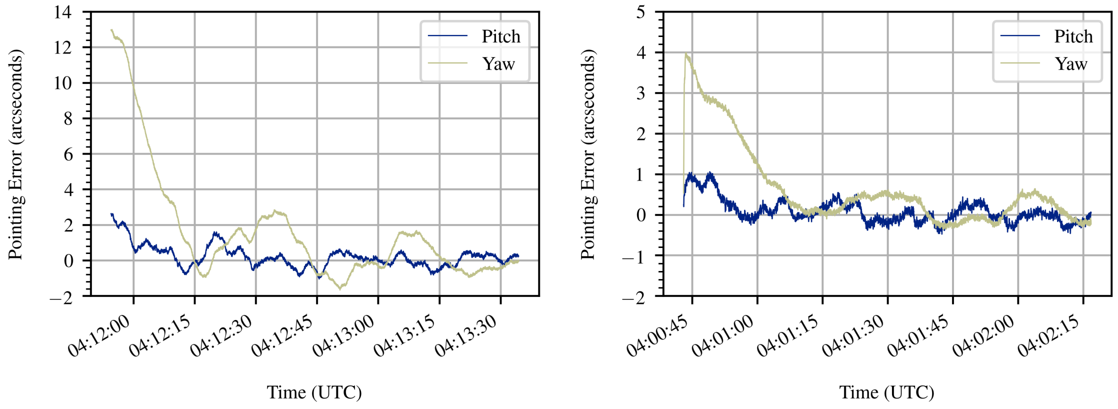
(a) MSO Initial Convergence

(b) SNNARC Initial Convergence

Figure 4. Initial convergence of the MSO method (a) and SNNARC method (b) are shown during non-suspended ground testing on July 16, 2021 at NASA WFF, demonstrating slow convergence of both methods and steady state bias in the MSO method.

As the MSO method utilizes only a one-layer network, the selection of the basis vector is especially critical. The pitch/yaw hub resolver angles were added to the basis vector, leading to the final version used by both neural network methods. Once the basis vector was updated, it was found that MSO no longer exhibited a steady state bias after converging. For tuning the convergence rate, the learning rate, M and N for SNNARC and Γ for MSO, were increased while the modification factor remained $\kappa < 1$. The MSO method also required a slight adjustment of its linear observer gain, K_2. It was found that as κ became smaller, the convergence rate increased but with a tendency for larger steady state errors while convergence rate increased as κ became large.

On July 22, 2021, a suspended system flight functional test was conducted, where pointing of the science instrument was demonstrated during an evening night test using the inertial star target, Polaris. The updated convergence behavior is shown in Figure 5, where it can be seen that the MSO method converged in a little over one minute while the SNNARC method reached steady state in a little under a minute. In both methods, the pitch channel converged rather quickly, while the yaw channel took a longer period of time. Once converged, the MSO algorithm had a steady state total RMS error value of



(a) MSO Tuned Convergence

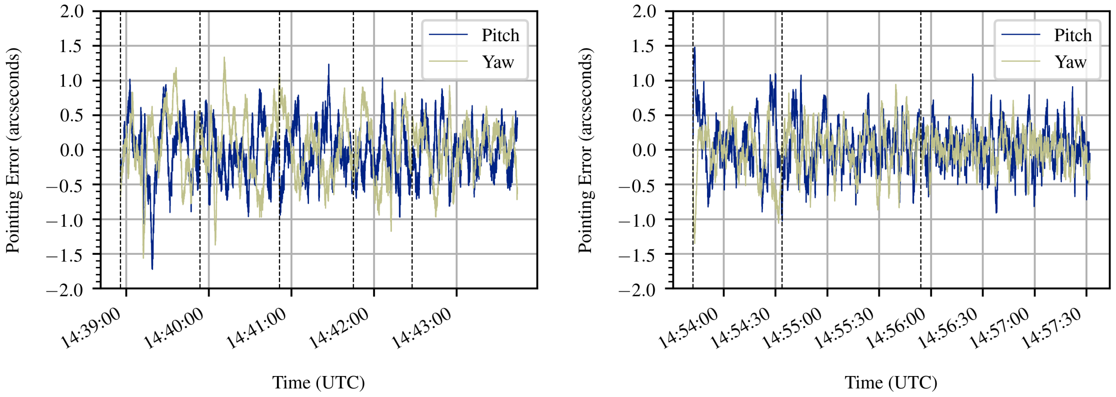
(b) SNNARC Tuned Convergence

Figure 5. Convergence of the MSO method (a) and SNNARC method (b) are shown during a suspended system flight functional on July 22, 2021 at NASA WFF, demonstrating the improved performance after updating to the final version of the basis vector, $\phi(\mathbf{z})$, and tuning the learning rates, M, N, and Γ, and modification factor, κ.

0.126 arcseconds while the SNNARC method achieved 0.198 arcseconds. Both methods outperformed the heritage PID system, which was only able to maintain a total RMS error value of 0.282 arcseconds. While the neural network methods outperformed the heritage PID system during this test, total system bandwidth was not yet considered and further performance testing was conducted at a later date to understand how pushing the system bandwidth affected the various methods.

4.3. STEADY STATE PERFORMANCE BASED ON LEARNING RATE SELECTION

While the learning rates and modification factor greatly affected the time for initial convergence, it was also desirable to understand how the learning rate affected steady state performance. Results from ground testing conducted on September 12, 2021 at the flight location at CSBF in Fort Sumner, New Mexico, are shown in Figure 6. During this test, the air conditioning in the building was operating, resulting in reduced overall pointing performance as compared to other test dates. The baseline controller parameters were



(a) MSO Learning Rate Adjustment

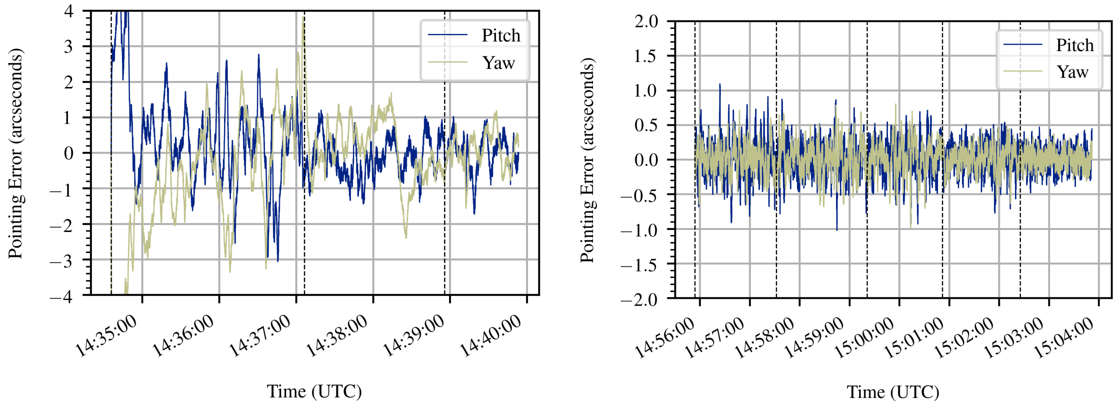
(b) SNNARC Learning Rate Adjustment

Figure 6. The learning rate of the MSO method (a) is increased from $\Gamma = 24 \rightarrow 42 \rightarrow 71 \rightarrow 92 \rightarrow 120$ and the SNNARC method (b) is increased from $M/N = 0.42 \rightarrow 0.8 \rightarrow 1.2$, where each change in learning rate is designated by a vertical dashed black line.

used and once the system was at steady state, the learning rate was slowly increased. It can be seen that as the learning rate is increased, there is only a marginal performance increase in the overall pointing behavior. It should be noted that in a more quiescent environment, increasing the learning rate exhibited a slightly higher improvement in steady state performance, though still not significant as compared to adjusting the bandwidth, as demonstrated next. This shows that beyond improving the convergence rate, there is not a large benefit in increasing the learning rate to improve steady state performance.

4.4. STEADY STATE PERFORMANCE BASED ON BANDWIDTH SELECTION

The largest factor in improving the overall pointing performance was through the increase of the overall system bandwidth. It should be noted that while this is simple in terms of the PID system, this approach is somewhat more nuanced and nebulous for the two neural network methodologies. As the systems are nonlinear and designed outside of the classical control domain, there are no direct analogies for system bandwidth. For the PID system, increasing the bandwidth results in changes to the K_p, K_i, and K_d gains, where



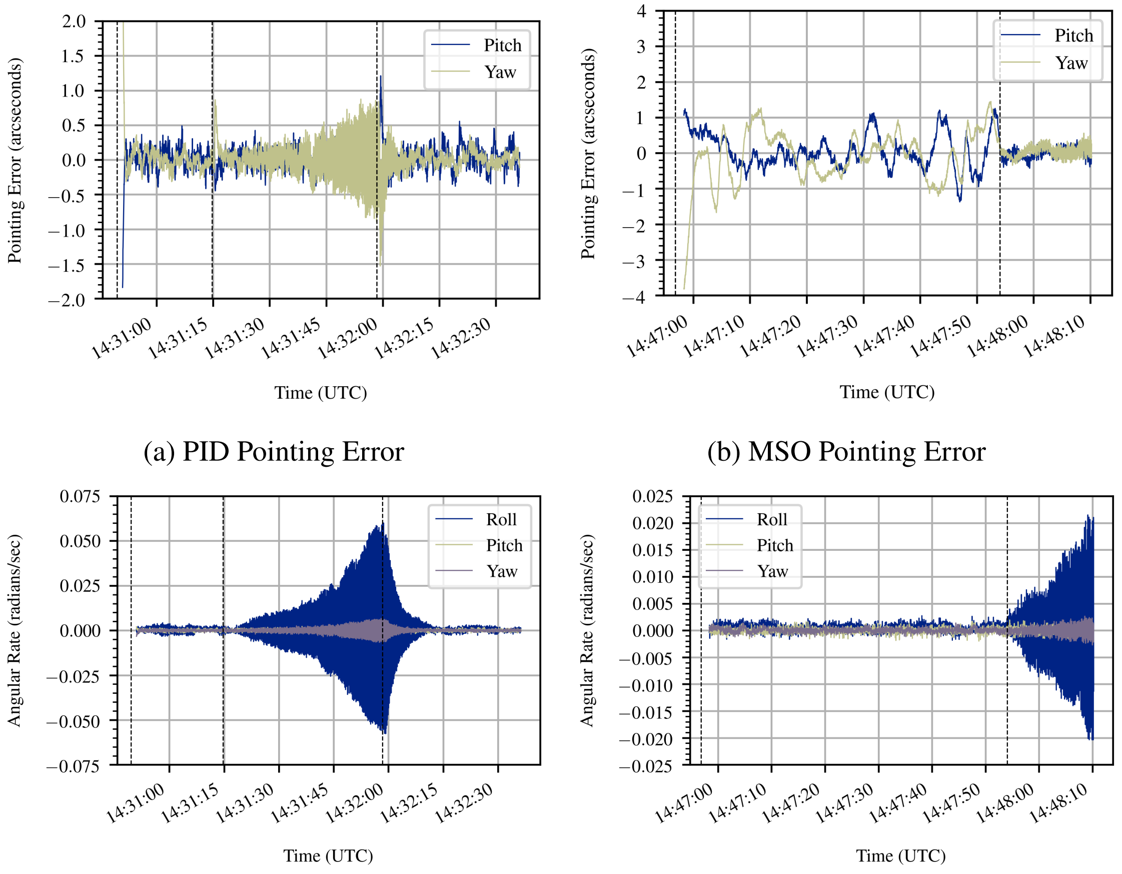
(a) MSO Bandwidth Tuning

(b) SNNARC Bandwidth Tuning

Figure 7. The system bandwidth is increased for the MSO method (a) from $B_w = 0.8 \longrightarrow 1.0 \longrightarrow 1.2\ H_z$ and for the SNNARC method (B) from $B_w = 0.8 \longrightarrow 0.9 \longrightarrow 1.0 \longrightarrow 1.2 \longrightarrow 1.4\ H_z$, where each change in bandwidth is designated by a vertical dashed black line.

for the neural network methods there are only increases in the equivalent K_p and K_d gains and the neural network gains remain constant. As such, even though a relationship was demonstrated between bandwidth selection and the equivalent K_p/K_d values, it cannot be said that the overall system bandwidths are the same. However, the trend in increasing and/or decreasing system bandwidth through tuning the gains is evident. Figure 7 shows that as the bandwidth is increased, the pointing performance is drastically improved for the two methodologies. These test data are also from the September 12, 2021 ground testing, where the disturbance environment was increased due to the active air conditioning system. A similar increase in bandwidth was implemented for the PID system, until the boundaries of stability were reached. The PID system achieved a total RMS error value of 0.193 arcseconds whereas the MSO only achieved 0.264 arcseconds, while the SNNARC method was able to achieve 0.169 arcseconds. An in-depth discussion of these results is presented in Section V.

From previous flight experiences using the heritage PID system, it is known that once the bandwidth is sufficiently high, instabilities in the system begin to occur. The controller begins reacting to the noise in the system as well as starting to interact with the structure, leading to an overall degradation of pointing performance and eventually the system becoming unstable. It was desired to understand when and how system instability would be presented using the neural network methods. The PID system was first tested to establish a baseline during the September 12, 2021 ground test with the air conditioning system active. The bandwidth of the system was slowly increased until instability presented, then the bandwidth was reduced to a stable set and stability reestablished. Once the system was at steady state, the control method was switched to the MSO method and the process repeated. The results in the pointing and angular rate data for the PID method and MSO method are shown in Figure 8, where the systems' behavior at the instability point is



(a) PID Pointing Error

(b) MSO Pointing Error

(c) PID Angular Rates

(d) MSO Angular Rates

Figure 8. The system bandwidth is increased for the PID method (a, c) and the MSO method (b, d) until the system becomes unstable showing the behavior of pointing (a, b) and angular rates (c, d) to give an understanding of how instability is presented. Each change in bandwidth is designated by a vertical dashed black line.

presented. While not shown, the SNNARC method behaved in a similar manner. It was found that as the bandwidth was increased, the first sign of instability began showing in the angular rates roll channel. The signal showed an increase in the noise and jitter, with a tendency for the signal to become slowly amplified. In the early stage of the symptoms, the angular rate jitter slowly increased and then stabilized, repeating this process a few times. At this stage, pointing initially remained stable. However, this borderline behavior was only briefly sustained before the growth in jitter became unbounded and instability was observed in the pointing behavior. As the WASP platform typically receives live data between approximately 10 to 50 Hz, it is generally possible to detect when instability begins occurring and allow for the operator to either switch controllers or lower the system bandwidth. The data show that the neural network methodologies exhibit a similar behavior to the heritage PID system in terms of how instability in the system is presented. This will assist operators during in-flight tuning of the control method and help in troubleshooting.

5. WASP HARDWARE RESULTS

To summarize the overall performance of the two neural networks in comparison to the heritage PID system, three specific test sets were analyzed. Two suspended system flight functional tests were used to determine expected flight performance, with one test conducted at NASA WFF, Virginia, on July 22, 2021 and the other test at CSBF, New Mexico, on September 7, 2021. Both tests were conducted during the night, utilizing the entire sensor suite including the star tracker. Pointing was conducted with the science team running their instrumentation at the same time. The environmental disturbance dynamics were generally quiescent (building air conditioning was disabled), though still likely higher as compared to expected flight conditions. The test from CSBF did have increased environment disturbances as compared to NASA WFF due to outdoor low winds level winds causing small drafts within the testing hanger. As the gain tuning was not yet finalized between the two test dates, results are compared to each other on only the date conducted. The third

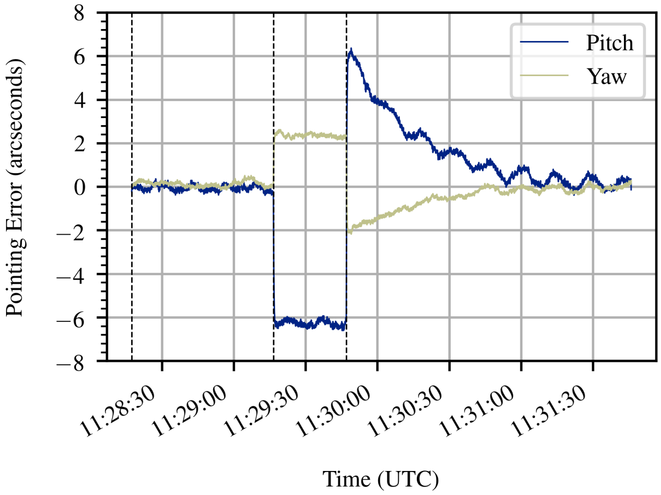
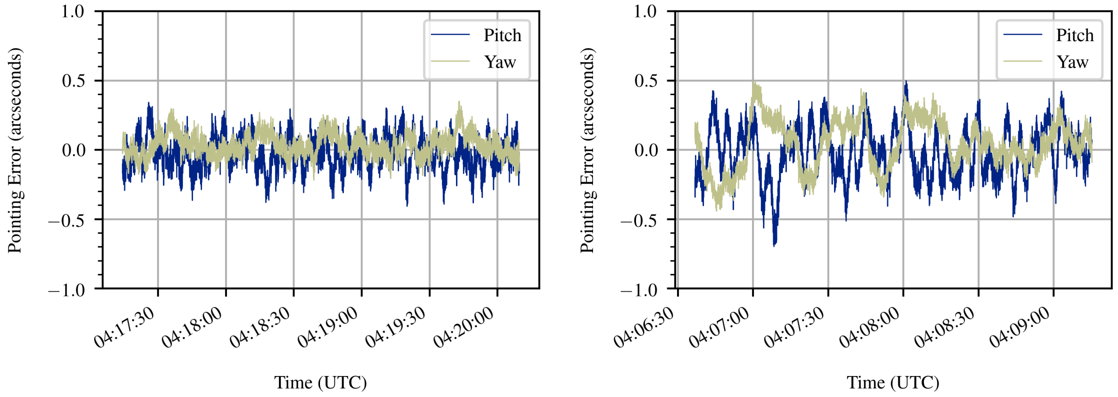


Figure 9. Disabling and enabling the neural network demonstrates that the neural network is correctly estimating the nonlinear dynamics and perturbations. Without the neural network a large steady state error is introduced.

test date used for performance analysis was on September 12, 2021, where ground testing was conducted with the building air conditioning actively running, giving a significantly increased disturbance environment.

Before presenting the overall performance of the neural network methods, it is useful to demonstrate the functionality of the neural network itself. It is known that errors in dynamics and/or unaccounted perturbations lead to a bias in a nonlinear system without an integral controller. To demonstrate that the neural network is properly estimating the dynamics, a case was run in which the neural network was allowed to reach steady state, and then the neural network portion of the controller was disabled, leaving only the linear portion enabled. The results are presented in Figure 9, where turning off the neural network causes a large bias in the steady state error, as expected. Once the neural network is enabled again, the system converges back to the original steady state.



(a) MSO Performance

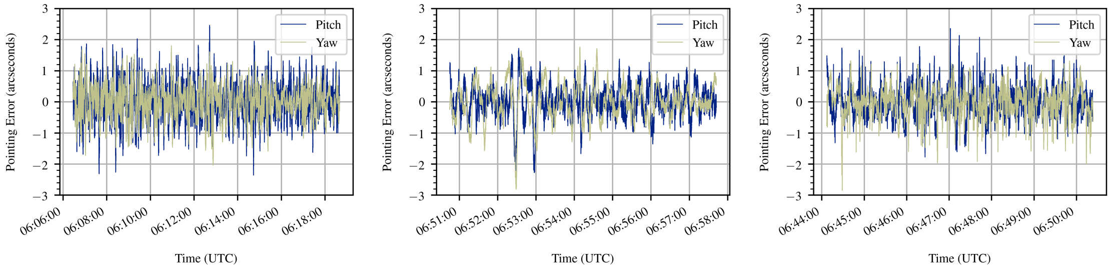
(b) SNNARC Performance

Figure 10. The performance of the MSO method (a) is compared to the performance of the SNNARC method (b) during a suspended system flight functional on July 22, 2021 at NASA WFF, demonstrating that the MSO method can achieve improved pointing stability as compared to the SNNARC method when both systems' bandwidths are set relatively high.

5.1. POINTING STABILITY

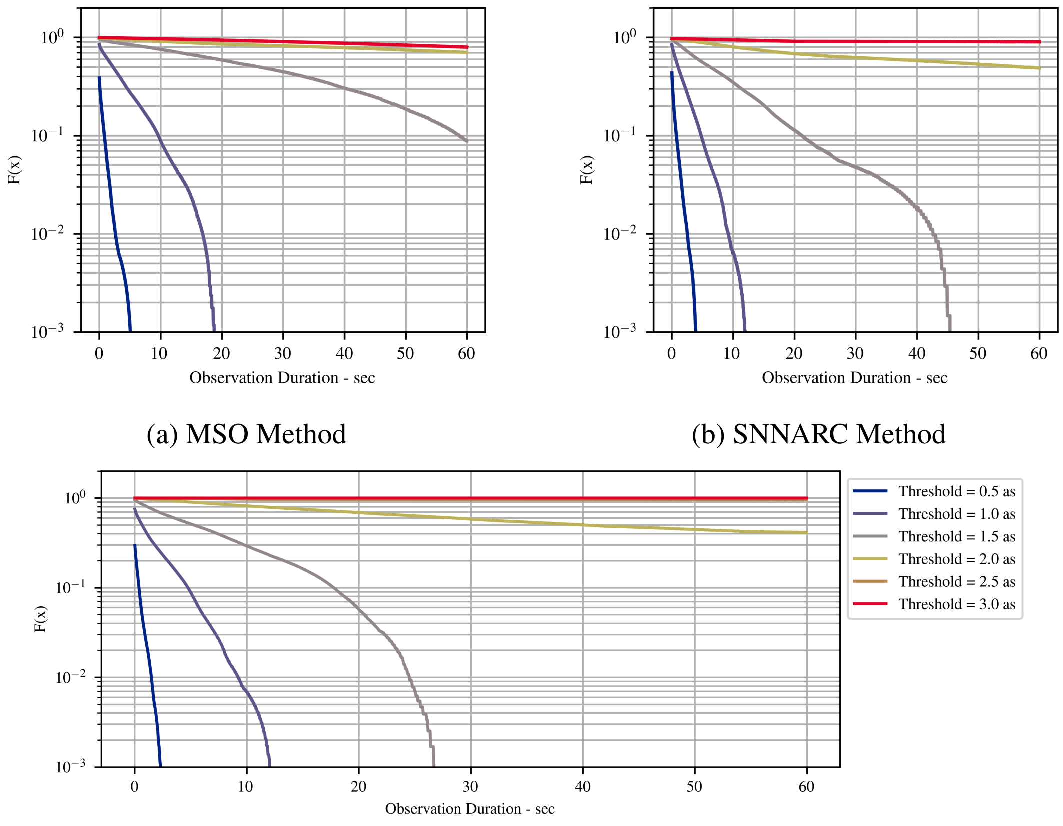
To demonstrate pointing stability, only the suspended hang tests were considered. During the first hang test on July 22, 2021, the bandwidth of the neural networks was set relatively high as compared to the PID system, so only the neural network controllers were compared to each other as shown in Figure 10. The MSO algorithm had a steady state total RMS error value of 0.126 arcseconds and the SNNARC method achieved 0.198 arcseconds. During the September 7, 2021 hang test, the bandwidths were approximately the same and kept low such that a more direct comparison between the three methods could be made, as shown in Figure 11. In this test, the PID system was able to maintain a total RMS error of approximately 0.689 arcseconds, while both the MSO and SNNARC methods achieved 0.605 arcseconds.

Though it appears that the MSO method exhibited similar performance to the SNNARC method during the second hang test, an interesting result is noticed when observation times are monitored instead of total RMS error. A period of 360 seconds of steady



(a) PID Performance (b) MSO Performance (c) SNNARC Performance

Figure 11. The performance of the PID method (a), MSO method (b), and SNNARC method (c) are compared during a suspended system flight functional on September 7, 2021 at CSBF, demonstrating that the neural network methods can outperform the PID system at low bandwidths. The MSO method exhibits a slight performance increase over the SNNARC method.



(a) MSO Method (b) SNNARC Method

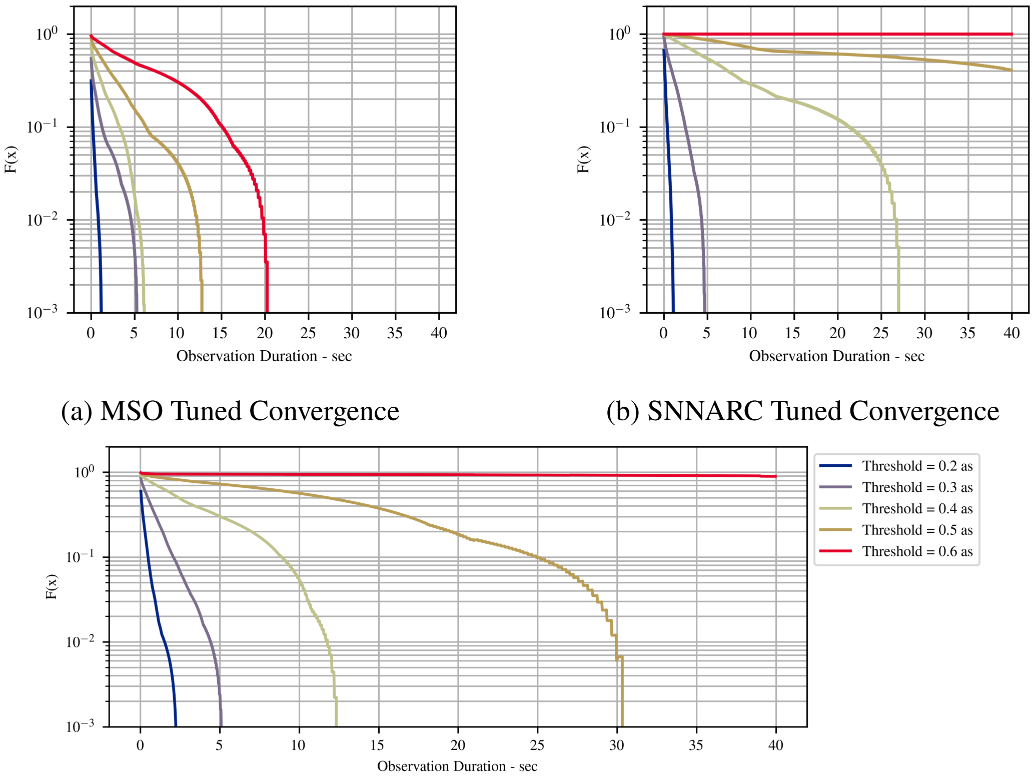
(c) PID Method

Figure 12. The probability, $F(x)$, to maintain observation lengths at various thresholds in total arcseconds of RMS error values is presented for the MSO (a), SNNARC (b), and PID (c) methods taken during a suspended system flight functional test on September 7, 2021 demonstrating that the neural network methods outperform the heritage PID system.

state performance was analyzed for the three methods, focusing on how long the control system could maintain the pointing within a certain threshold. The data are given by $F(x)$ (y-axis), which represents the probability that an observation length (x-axis) could be maintained before a violation occurs at various thresholds specified in arcseconds of total RMS error, as shown in Figure 12. It is interesting to note that at smaller thresholds, below 2.0 arcseconds, the MSO method is able to maintain significantly longer observation durations as compared to both the SNNARC and PID methods. However, when looking at values at and above 2.0 arcseconds, the probability of violating the threshold during an observation period is higher. Between the two suspended pointing tests, this demonstrates that while the MSO method is able to maintain a tighter pointing performance compared to the SNNARC method, it is more prone to brief excursions beyond desired thresholds. It was also found that by increasing the system bandwidth, the MSO method typically showed an even larger improvement in performance as compared to the SNNARC and PID methods, though this result was only during quiescent periods as discussed next. From these results, it is evident that the two neural networks are capable of outperforming the heritage PID system, with the MSO method perhaps having a slight edge over the SNNARC method when considering the average total RMS error.

5.2. HIGH DISTURBANCE ENVIRONMENT

From the high disturbance environment ground testing on September 12, 2021 in which the building air conditioning was allowed to run, it was found that while the PID system achieved a total RMS error value of 0.193 arcseconds and the SNNARC method achieved 0.169 arcseconds, the MSO method only achieved 0.264 arcseconds. This test was the last date where specific algorithm testing was conducted in preparation for the PICTURE-C launch opportunity. It is interesting to note that while the MSO method had always outperformed the PID and SNNARC algorithms during quiescent testing periods, it appears that in a highly dynamic environment (the air conditioning created strong drafts within the hanger) the performance begins to degrade more quickly as compared to SNNARC and PID. This is even more evident when considering the observation periods that could be achieved at various thresholds. It is clear from Figure 13 that while the SNNARC method continues to outperform the PID method in terms of observation lengths, the MSO performance is significantly degraded. This behavior was also observed on another occasion during the three month testing campaign. It is surmised that the two-layer network utilized by SNNARC is more capable of learning various disturbances without needing further modification of the basis vector. One-layer networks are very sensitive to the selection of the basis vector, and as such it is likely that the MSO method would require additional inputs into the basis vector to assist the learning in a highly dynamic environment.



(a) MSO Tuned Convergence

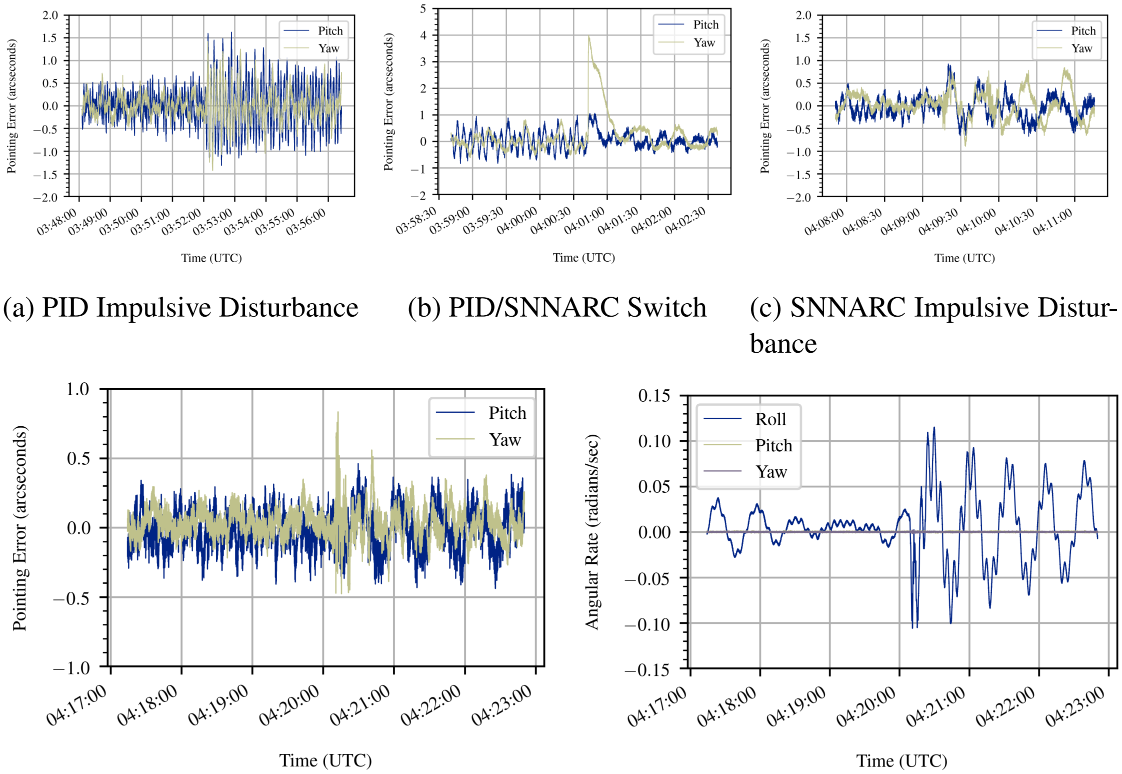
(b) SNNARC Tuned Convergence

(c) PID Tuned Convergence

Figure 13. The probability, $F(x)$, is presented to maintain observation lengths at various thresholds in total arcseconds of RMS error values for the MSO (a), SNNARC (b), and PID (c) methods in a high disturbance environment, taken from ground testing on September 12, 2021 demonstrating that while the SNNARC method continues to outperform the heritage PID system, the performance of the MSO method begins to degrade quickly.

5.3. INDUCED PENDULOUS DYNAMICS DISTURBANCE ENVIRONMENT

When the high-altitude balloon platform is at float, the balloon frequently experiences accelerations in response to changing float winds. This translates to changes in the pendulous dynamics of the WASP platform, the flight train, and the high-altitude balloon. To understand the behavior of the various control methods under this disturbance environment, tests were conducted at NASA WFF during a suspended system flight functional on July 22, 2021. Once the system was stabilized and maintained at steady state for a few minutes, an operator then gently pushed the external gondola, simulating the change in pendulous dynamics from a potential impulsive wind gust. The system pointing performance was monitored to see how well the centerbody could remain aligned with the desired inertial target, Polaris. The performance of the various methods is shown in Figure 14 for the PID, SNNARC, and MSO methods. The primary motion input into the system is observed in the roll channel angular rates, though this also translates into yaw and pitch disturbances in the pointing performance. It can be seen for the PID system that the pointing error jumps into the 1.5 arcsecond range and that significant jitter is introduced into the pitch channel. Though the PID system returned to steady state after a few minutes, the pitch channel continued to show some jitter. When the controller was switched to SNNARC at this point, the jitter in the pitch channel was removed as the system reached steady state. After steady state was achieved with the SNNARC method, an additional push was given to the external gondola to see how the SNNARC method performed. It is seen that the error only increased to about 1.0 arcseconds before the system began to converge back to steady state. A similar test for MSO was conducted that evening, which showed an even smaller pointing error and more rapid convergence. It is predicted that the MSO method, which, unlike the SNNARC method, monitors and predicts disturbances in the roll channel in addition to pitch and yaw, makes it more resilient to disturbances that may occur from roll dynamics. This test shows that both neural network methods are capable of withstanding disturbances resulting from the pendulous dynamics change from wind gusts and both



(a) PID Impulsive Disturbance

(b) PID/SNNARC Switch

(c) SNNARC Impulsive Disturbance

(d) MSO Impulsive Disturbance

(e) MSO Angular Rate

Figure 14. An impulse disturbance is introduced into the PID system (a) to baseline performance before switching to the SNNARC method (b), which further reduced the disturbance. Similar impulsive disturbances were introduced to the SNNARC method (c) and MSO method (d), where the the disturbance in the roll channel (e) is clearly seen.

appear to significantly outperform the heritage PID system, boding well for their in-flight performance at float environment. As the disturbance input into the system was given by a human operator, it is acknowledged that variations in the amount of impulsive force input into the system may have been present. However, this test was conducted multiple times by a skilled operator and as such this result can be generalized.

6. FUTURE WORK

The three-month testing campaign in preparation for the Fall 2021 flight operations for the PICTURE-C science mission showed that the two neural network methods may provide improved pointing performance to the WASP mission. The WASP platform was flight-ready in the middle of September and multiple launch attempts were made through early October. Due to unfavorable weather conditions during the short launch window, the PICTURE-C flight was canceled for 2021 and the flight was rescheduled for Fall 2022. However, a number of areas for improvement have been identified over the course of testing and these improvements are planned for implementation for the next flight opportunity. The next mission that will be flown is XL-Calibur, which is an X-ray polarimetry mission, and is currently in the integration phase at NASA WFF in preparation for a Spring/Summer 2022 flight from Sweden. This science instrument is significantly larger than PICTURE-C, with its inertia approximately an order of magnitude larger, as can be seen in Figure 15. While the full assembly is not yet complete and the system is not yet in its flight configuration, preliminary testing of the pointing system was conducted on October 25, 2021. During this testing, both the SNNARC and MSO methods were tested and compared to the PID system. Both methods were able to outperform the PID system, with the MSO method achieving the best performance with an impressive RMS error of only 0.048 arcseconds in pitch and 0.055 arcseconds in yaw. Not only has this shown the improved performance of the neural network methods, it shows that these control methodologies are viable for a variety of centerbody configurations.

One area for improvement identified is adding the ability to track a geodetic target. The current WASP software only provides a target reference position and assumes that the target inertial rates are zero. However, for a geodetic target the system will be following Earth's rotation rate, and as such non-zero target inertial angular rates are present. The heritage PID system is able to accommodate these non-zero rates through the integrator and adequate pointing is still achieved. However, the MSO and SNNARC methods require the

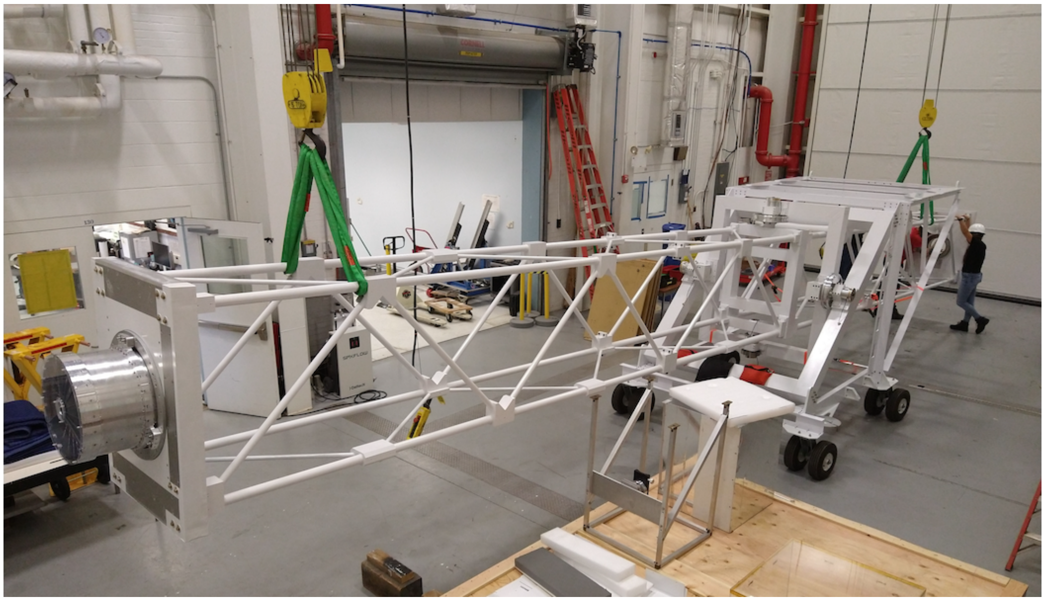


Figure 15. WASP XL-Calibur.

knowledge of the target angular velocities and as such a bias in pointing currently appears when tracking a geodetic target. An additional step in the flight software will be added to the two neural network methods to calculate the desired inertial angular rates so that the two methods can track a geodetic target.

Another area for improvement relates to changing targets. During missions such as PICTURE-C, it is desirable to track different inertial targets and to switch between them during the course of operations. The heritage PID system utilizes rate limiting in the controller as well as a saturation limit to the integrator to prevent integrator windup when switching targets. The MSO and SNNARC methods do not currently have this functionality, and as such large maneuvers executed when switching targets could lead to instability. This was especially noted in the SNNARC method, where the tracking error drives the neural network estimation. A sudden large error induced by switching targets led to instability in the neural network learning and the control system diverged. Though a temporary solution was found by reducing the learning rate to maintain system stability, a more elegant solution is desired. The MSO method, which learns based on observer error, was able to remain stable throughout such a maneuver. Initial testing did show instability, but this was traced to missing saturation limits being applied to the control method. As such, when the controller generated a torque beyond the capacity of the pitch/yaw motor hubs, the produced torque did not match the input torque into the observer and the system diverged. Once torque saturation was applied, the system remained converged even when the maximum torque was applied. By introducing a rate limit, the transient performance between targets could be improved without needing to reduce the learning rates.

During the initial pointing tests for the XL-Calibur mission, an operator accidentally bumped the centerbody and SNNARC, the control system running at the time, started to become unstable. Previous impulsive disturbance testing only applied a disturbance to the external gondola, and this behavior had not been noted in the past. It was found that this behavior was related to a high learning rate. Once the learning rate was reduced, the pointing remained stable even after intentional impulse disturbances were given to the centerbody. While this type of disturbance is not anticipated during flight, it is a useful aspect to study. When the learning rate was reduced the overall pointing performance remained similar, so future testing will examine the balance between initial neural network convergence, steady state behavior, and resilience to impulsive disturbances. As most missions have long periods of observation, initial convergence speed is not as critical as compared to steady state performance and disturbance rejection.

Unlike the SNNARC method, the MSO method only utilizes a one-layer neural network and is much more sensitive to the selection of a basis vector. This was particularly evident in the high environmental disturbance testing where the SNNARC method was able to outperform the MSO method. It is desirable to consider if adding a two-layer network, similar to SNNARC, will improve the overall performance of the MSO method. One potential draw back in adding additional layers to the neural network is an increase in required computational power. From prior testing, running the flight software with the PID method on the flight processor utilized only 24 percent of the CPU, while the MSO method used about 28 percent. The SNNARC method, which uses a two-layer network, used almost 50 percent of the CPU, nearly doubling the required computational power needed.

7. CONCLUSION

This research effort implemented, tuned, and tested two new control methodologies for the Wallops Arc-Second Pointer system for high-altitude balloon platforms. The SNNARC methodology, which uses a two-layer neural network, estimates nonlinear dynamics within the control space. The MSO methodology, which uses an observer and a one-layer neural network, estimates nonlinear dynamics outside the control space and uses input-output feedback linearization to control the system. The heritage PID control system and the two new methodologies were tested over a period of three months in preparation for a science mission originally scheduled to fly Fall 2021. It was found that the two neural network methods generally outperformed the heritage PID system. The MSO method typically exhibited the best pointing performance, particularly at higher bandwidths, in the majority of the tests conducted and is expected to be the superior method during flight at the 120,000 ft float altitude. It was found that the MSO method did suffer in performance when the disturbance environment was significant while the SNNARC method continued to outperform the PID system. Though such high disturbances are not anticipated in flight, this demonstrates where the two-layer neural network architecture from the SNNARC method may be superior to a one-layer neural network architecture. This research has shown that the addition of a neural network to a gimbal-controlled inertial pointing system suspended from a high-altitude balloon platform can provide improved performance as compared to traditional heritage pointing control systems. The testing campaign has identified a number of improvements that could be implemented into the two methodologies and these will be pursued as the WASP platform continues to be used for high altitude balloon missions.

FUNDING SOURCES

Portions of this research have been supported through the NASA Pathways Program and NASA Wallops Flight Facility.

ACKNOWLEDGMENTS

First and foremost, a heartfelt thank you to Dr. Balakrishnan for his major contributions and being a source of inspiration for this work. Special thanks are also extended to Jim Lanzi, Scott Heatwole, and Zach Peterson for their mentorship, support, and contributions to applying this research to NASA spaceflight systems. Thanks are given also to the Guidance, Navigation, and Control, and Mission System Engineering Branch (Code 598) and Balloon Program Office of NASA Wallops Flight Facility and the Attitude Control Systems Engineering Branch (Code 591) of NASA Goddard Space Flight Facility, as well as the NASA Pathways Program.

REFERENCES

[1] Stuchlik, D. W., "The Wallops Arc Second Pointer - A Balloon Borne Fine Pointing System," *Proceedings of the 2015 AIAA Balloon Systems Conference (June 2015, Dallas, TX)*, 2015, pp. 1–15. https://doi.org/10.2514/6.2015-3039.

[2] Stuchlik, D. W., and Lanzi, R., "The NASA Wallops Arc-Second Pointer (WASP) System for Precision Pointing of Scientific Balloon Instruments and Telescopes," *Proceedings of the 2017 AIAA Balloon Systems Conference (June 2017, Denver, CO)*, 2017, pp. 1–8. https://doi.org/10.2514/6.2017-3609.

[3] Fairbrother, D. A., "2017 NASA Balloon Program Update," *Proceedings of the 2017 AIAA Balloon Systems Conference (June 2017, Denver, CO)*, 2017, pp. 1–8. https://doi.org/10.2514/6.2017-3090.

[4] Jones-Wilson, L., Susca, S., Diaz, C., Chang, H., Duffy, E., Effinger, R., Lewis, D., Liewer, K., Lo, K., Ochoa, H., Perez, J., Rizvi, A., Seubert, C., Umsted, C., Borden, M., Clark, P., Massey, R., and Porter, M., "A Sub-Arcsecond Pointing Stability Fine Stage for a High Altitude Balloon Platform," *Proceedings of the 2017 IEEE Aerospace Conference (March 2017, Big Sky, MT)*, 2017, pp. 1–15. https://doi.org/10.1109/AERO.2017.7943590.

[5] Kopp, G., Smith, P., Belting, C., Castleman, Z., Drake, G., Espejo, J., Heuerman, K., Lanzi, J., and Stuchlik, D., "Radiometric flight results from the HyperSpectral Imager for Climate Science (HySICS)," *Geoscientific Instrumentation, Methods and Data Systems*, Vol. 6, 2017, pp. 169–191. https://doi.org/10.5194/gi-6-169-2017.

[6] Hurford, T. A., Mandell, A., Reddy, V., and Young, E., "Observatory for Planetary Investigations from the Stratosphere," *Proceedings of the 46th Lunar and Planetary Science Conference (March 2015, The Woodlands, TX)*, 2015, pp. 1–2.

[7] Abarr, Q., Awaki, H., Baring, M., Bose, R., De Geronimo, G., Dowkontt, P., Errando, M., Guarino, V., Hattori, K., Hayashida, K., Imazato, F., Ishida, M., Iyer, N., Kislat, F., Kiss, M., Kitaguchi, T., Krawczynski, H., Lisalda, L., Matake, H., Maeda, Y., Matsumoto, H., Mineta, T., Miyazawa, T., Mizuno, T., Okajima, T., Pearce, M., Rauch, B., Ryde, F., Shreves, C., Spooner, S., Stana, T.-A., Takahashi, H., Takeo, M., Tamagawa, T., Tamura, K., Tsunemi, H., Uchida, N., Uchida, Y., West, A., Wulf, E., and Yamamoto, R., "XL-Calibur – a Second-Generation Balloon-Borne Hard X-Ray Polarimetry Mission," *Astroparticle Physics*, Vol. 126, 2021, p. 102529. https://doi.org/10.1016/j.astropartphys.2020.102529.

[8] Gopalswamy, N., Newmark, J., Yashiro, S., Mäkelä, P., Reginald, N., Thakur, N., Gong, Q., Kim, Y. H., Cho, K. S., Choi, S. H., Baek, J. H., Bong, S. C., Yang, H. S., Park, J. Y., Kim, J. H., Park, Y. D., Lee, J. O., Kim, R. S., and Lim, E. K., "The Balloon-Borne Investigation of Temperature and Speed of Electrons in the Corona (BITSE): Mission Description and Preliminary Results," *Solar Physics*, Vol. 296, 2021. https://doi.org/10.1007/s11207-020-01751-8.

[9] Mendillo, C. B., Hewawasam, K., Howe, G. A., Martel, J., Cook, T. A., and Chakrabarti, S., "The PICTURE-C exoplanetary direct imaging balloon mission: first flight preparation," *Techniques and Instrumentation for Detection of Exoplanets IX, Proceedings of the 2019 International Society for Optics and Photonics Conference (August 2019, San Diego, CA)*, Vol. 11117, SPIE, 2019, pp. 101 – 111. https://doi.org/10.1117/12.2529710.

[10] Mendillo, C. B., Hewawasam, K., Martel, J., Potter, T., Cook, T. A., and Chakrabarti, S., "The PICTURE-C Exoplanetary Imaging Balloon Mission: First Flight Preparation," *Techniques and Instrumentation for Detection of Exoplanets IX, Proceedings of the 2021 International Society for Optics and Photonics Conference (August 2021, San Diego, CA)*, Vol. 11823, SPIE, 2021, pp. 125 – 133. https://doi.org/10.1117/12.2594749.

[11] Abdo, M., Vali, A. R., Toloei, A., and Arvan, M. R., "Research on the Cross-Coupling of a Two Axes Gimbal System with Dynamic Unbalance," *International Journal of Advanced Robotic Systems*, Vol. 10, 2013, p. 357. https://doi.org/10.5772/56963.

[12] Ekstrand, B., "Equations of Motion for a Two-Axes Gimbal System," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 37, No. 3, 2001, pp. 1083–1091. https://doi.org/10.1109/7.953259.

[13] Liu, S., Lu, T., Shang, T., and Xia, Q., "Dynamic Modeling and Coupling Characteristic Analysis of Two-Axis Rate Gyro Seeker," *International Journal of Aerospace Engineering*, Vol. 2018, 2018. https://doi.org/10.1155/2018/8513684.

[14] Romualdez, L. J., Damaren, C. J., Li, L., Galloway, M. N., Hartley, J. W., Netterfield, C. B., Clark, P., and Massey, R. J., "Precise Pointing and Stabilization Performance for the Balloon-Borne Imaging Testbed: 2015 Test Flight," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, Vol. 231, No. 4, 2017, pp. 713–727. https://doi.org/10.1177/0954410016641451.

[15] Markley, F. L., and Crassidis, J. L., *Fundamentals of Spacecraft Attitude Determination and Control*, Springer, First Edition, 2014.

[16] Kahlil, H. K., *Nonlinear Control*, Pearson, First Edition, 2015.

[17] Darling, J. E., Legrand, K. A., Galchenko, P., Pernicka, H. J., DeMars, K. J., Shirley, A. T., McCabe, J. S., Schmid, C. L., Haberberger, S. J., and Mundahl, A. J., "Development and Flight of a Stereoscopic Imager for Use in Spacecraft Close Proximity Operations," *Proceedings of the 39th Annual AAS Rocky Mountain Section Guidance and Control Conference (February 2016, Breckenridge, CO)*, Vol. 157, 2016, pp. 489–500.

[18] MacKunis, W., Leve, F., Patre, P., Fitz-Coy, N., and Dixon, W., "Adaptive Neural Network-Based Satellite Attitude Control in the Presence of CMG Uncertainty," *Aerospace Science and Technology*, Vol. 54, 2016, pp. 218–228. https://doi.org/10.1016/j.ast.2016.04.022.

[19] Davis, J., Galchenko, P., Jennings, D., and Pernicka, H. J., "Development and Validation of a GNC Algorithm Using a Stereoscopic Imaging Sensors in Close Proximity Operations," *Proceedings of the 2017 AAS/AIAA Astrodynamics Specialist Conference (August 2017, Stevenson, WA)*, Vol. 162, 2018, pp. 3149–3161.

[20] Galchenko, P., and Pernicka, H. J., "Precision Control of Microsatellite Swarms Using Plasmonic Force Propulsion," *Proceedings of the 2018 AAS/AIAA Astrodynamics Specialist Conference (August 2018, Snowbird, UT)*, Vol. 167, 2018, pp. 935–954.

[21] Jennings, D., Davis, J., Galchenko, P., and Pernicka, H. J., "Validation of a GNC Algorithm Using a Stereoscopic Imaging Sensor to Conduct Close Proximity Operations," *Proceedings of the 41st Annual AAS Rocky Mountain Section Guidance and Control Conference (2018, Breckenridge, CO)*, Vol. 164, 2018, pp. 47–58.

[22] Galchenko, P., and Pernicka, H. J., "Precision Control of Microsatellite Swarms Using Plasmonic Force Propulsion," *Proceedings of the 2018 AAS/AIAA Astrodynamics Specialist Conference (August 2018, Snowbird, UT)*, Vol. 167, 2019, pp. 935–954.

[23] Ghafoor, A., Galchenko, P., Balakrishnan, S. N., Pernicka, H., and Yucelen, T., "ETNAC Design Enabling Formation Flight at Liberation Points," *Proceedings of the 2019 American Control Conference (July 2019, Philadelphia, PA)*, 2019, pp. 3689–3694. https://doi.org/10.23919/ACC.2019.8814922.

[24] Galchenko, P., Pernicka, H. J., and Balakrishnan, S. N., “Pointing System Design for the COronal Diagnostic Experiment (CODEX) using a Modified State Observer and a Neural Network Controller,” *Proceedings of the 2020 AAS/AIAA Astrodynamics Specialist Conference (August 2020, South Lake Tahoe, CA)*, 2021, pp. 645–660.

[25] Lewis, F. L., Jagannathan, S., and Yesildirek, A., *Neural Network Control of Robot Manipulators and Nonlinear Systems*, Taylor and Francis, 1999.

[26] Rajagopal, K., Mannava, A., Balakrishnan, S., Nguyen, N., and Krishnakumar, K., “Neuroadaptive Model Following Controller Design for Non-Affine and Non-Square Aircraft Systems,” *Proceedings of the 2009 AIAA Guidance, Navigation, and Control Conference (August 2009, Chicago, IL)*, 2009, pp. 1–21. https://doi.org/10.2514/6.2009-5737.

[27] Mannava, A., Balakrishnan, S., Tang, L., and Landers, R., “Optimal Tracking Control of Motion Systems,” *IEEE Transactions on Control Systems Technology*, Vol. 20, 2012, pp. 1548–1558. https://doi.org/10.1109/TCST.2011.2168608.

[28] Harl, N., Rajagopal, K., and Balakrishnan, S. N., “Neural Network Based Modified State Observer for Orbit Uncertainty Estimation,” *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 4, 2013, pp. 1194–1209. https://doi.org/10.2514/1.55711.

III. NEURAL NETWORK CONTROL SCHEMES ENABLING DEEP SPACE SMALL SPACECRAFT DISTRIBUTED SYSTEM MISSIONS

Pavel Galchenko[5] and Henry Pernicka[6]
Missouri University of Science and Technology, Rolla, MO, 65409-0050

ABSTRACT

Two neural network control schemes are introduced in the context of small spacecraft precision formation flight missions in deep space environments, with the goal of enabling distributed system missions. The first neural network method uses tracking error to estimate system perturbations while the second neural network method uses an observer for estimation and input-output feedback linearization for control. Two spacecraft in precision formation flight were simulated in a libration point orbit and achieved the desired science objectives of a notional advanced science mission concept. The two neural network schemes provide improved position and attitude control performance while offering significant savings in propellant consumption as compared to the traditional proportional-integral-derivative control method.

NOMENCLATURE

x, y, z = spacecraft position states in CR3BP [km]

$\dot{x}, \dot{y}, \dot{z}$ = spacecraft velocity states in CR3BP [km/s]

u_x, u_y, u_z = translation control accelerations [N]

ω_s = angular velocity of CR3BP rotating frame [rad/s]

[5] Ph.D. Candidate, Department of Mechanical and Aerospace Engineering, Missouri University of Science and Technology, 400 W 13th St, Rolla, MO 65409-0050.

[6] Curators' Distinguished Teaching Professor of Aerospace Engineering, Department of Mechanical and Aerospace Engineering, Missouri University of Science and Technology, 400 W 13th St, Rolla, MO 65409-0050.

r_1, r_2	=	distance of primary/secondary bodies to spacecraft [km]
r_{B1}, r_{B2}	=	distance of primary/secondary bodies to barycenter [km]
μ_1, μ_2	=	gravitational parameter of primary/secondary bodies [m^3/s^2]
q_1, q_2, q_3, q_4	=	spacecraft attitude quaternion
$\theta_\alpha, \theta_\beta, \theta_\gamma$	=	spacecraft roll, pitch, and yaw rotations [rad]
$\omega\alpha, \omega\beta, \omega\gamma$	=	spacecraft roll, pitch, and yaw angular velocities [rad/s]
$u_\alpha, u_\beta, u_\gamma$	=	roll, pitch, and yaw control torques [Nm]
$f_{\mathbf{x}}(\mathbf{x})$	=	known system dynamics
$g_{\mathbf{x}}$	=	control mapping to dynamics
$\delta(\mathbf{x})$	=	known disturbance dynamics
J	=	system inertia matrix [kg m^2]
P_{sun}	=	solar radiation pressure at spacecraft position
$\Im_{\mathrm{sun}}$	=	solar constant at 1.0 AU
$r_{\mathrm{s}\rightarrow\odot}$	=	distance between spacecraft and Sun [AU]
$\hat{\mathbf{r}}_{\mathrm{s}\rightarrow\odot}$	=	spacecraft-to-Sun unit vector
$R^i_{\mathrm{diff}}, R^i_{\mathrm{spec}}$	=	coefficient of diffuse/spectral reflection
$M_{s/c}$	=	mass of the spacecraft [kg]
$\mathbf{F}^i_{\mathrm{SRP}}, \mathbf{L}^i_{\mathrm{SRP}}$	=	force [N] and torque [Nm] disturbance from SRP
A^i	=	area of the spacecraft i^{th} side [km^2]
$\mathbf{n}^i_B$	=	normal outward unit vector of the spacecraft i^{th} side
$\mathbf{r}^i_{\mathrm{press}}$	=	center of mass to the center of pressure vector [km]
$\mathbf{x}_{d1}, \mathbf{x}_{d2}$	=	desired position/orientation [km or quaternion]
$\mathbf{x}_{d1}, \mathbf{x}_{d2}$	=	desired velocity or angular rate [km/s or rad/s]
f_d	=	desired dynamics
$\mathbf{e} = [\ \mathbf{e}_1 \quad \mathbf{e}_2\ ]^T$	=	error states
A_{cl}	=	closed-loop dynamics
K_p	=	proportional gain [(km/s^2)/km] or [Nm/rad]

K_i	=	integral gain [(km/s^2)/(km-sec)] or [Nm/(rad-sec)]
K_d	=	derivative gain [(km/s^2)/(km/sec)]] or [Nm/(rad/sec)] gains
ω_n	=	controller natural frequency [rad/s]
ζ	=	damping ratio
α	=	tuning parameter
$\mathcal{F}_{\mathbf{x}}, \hat{\mathcal{F}}_{\mathbf{x}}$	=	true and estimated dynamics
$\Delta(\mathbf{x})$	=	unmodeled perturbations/dynamics
σ, σ'	=	activation function and its derivative
$\phi\left(\mathbf{z}\right)$	=	basis vector
ε	=	bounded neural network estimation error
$W^T, V^T, \hat{W}^T, \hat{V}^T$	=	true and estimated neural network weighting matrices
$\tilde{W}, \tilde{V}$	=	neural network weight matrices estimation error
λ	=	gain coefficient
$\mathbf{u}_r$, K_z	=	robustifying control and gain
$\tilde{\mathcal{F}}_{\mathbf{x}}$	=	dynamics estimation error
M, N	=	neural network adaptation gain matrices
κ	=	design parameter
L	=	Lyapunov function
$tr\left\{\right\}$	=	trace operator
K_2	=	linear observer gain matrix
$\mathbf{e}_a$	=	observer estimation error
b_{1c}, b_{2c}, b_{3c}	=	leader/chief spacecraft body axes
b_{1f}, b_{2f}, b_{3f}	=	follower spacecraft body axes

1. INTRODUCTION

Advances in hardware capabilities of the small spacecraft platform have begun to enable deep space missions typically reserved for large monolithic spacecraft [1]. Small spacecraft are able to play an increasing role in advanced mission concepts for distributed system mission architectures with cost, size, mass, and power savings as compared to traditional large monolithic spacecraft [2]. To achieve the objectives of these missions, often a number of small spacecraft are needed to maintain a tight flight formation for scientific observations, requiring precision translation and attitude control. With the maturity of the small spacecraft platform and the recent development of micropropulsion systems designed for such architectures, the required tight formation tolerances can now be achieved [3]. This study addresses the design of robust and precise control solutions to enable the use of these technologies in precision formation flight.

1.1. LITERATURE REVIEW

A number of advanced mission concepts require precision formation flight (PFF) in a deep space environment for distributed system mission architectures. Planet finding missions such as the Terrestrial Planet Finder (TPF) [4] and Darwin [5] seek to create infrared interferometers using multiple spacecraft in PFF to emulate powerful telescopes to search for distant planets. The Micro-Arcsecond X-Ray Interferometry Mission (MAXIM) [6] seeks to observe and study black hole phenomena while the Laser Interferometer Space Antenna mission [7] seeks to study gravitational waves. Other missions such as the NASA Stellar Imager [8] seek to use small spacecraft in PFF to create a virtual telescope for observation of stellar surfaces. Each of these unique missions intends to utilize swarms of spacecraft in PFF to enable advanced mission concepts that would otherwise be infeasible for large monolithic spacecraft due to size, mass, and power requirements.

Three-body dynamics, often described by the Circular Restricted Three-Body Problem (CR3BP) equations, give rise to libration points (locations relative to two primary bodies where dynamic forces/accelerations acting on a spacecraft are in equilibrium). These points offer unique advantages to deep space missions because they allow for spacecraft to enter and remain in orbit at these locations with only modest station-keeping effort, facilitating long duration scientific missions. However, missions like MAXIM and Stellar Imager require relative position control in the millimeter to submillimeter range [9, 10]. To achieve these tolerances, micropropulsion systems must be capable of producing thrust values in the nano- to millinewton ranges. Marchand and Howell studied a variety of control strategies to quantify propulsion requirements for advanced missions such as MAXIM and TPF [11]. Using the CR3BP model as well as an ephemeris model, it was found that thrust levels ranged from nano- to millinewtons for large monolithic spacecraft, as well as requiring nearly continuous control when attempting to achieve submillimeter position control. While studies by Howell and Marchand found the potential existence of natural formations at libration points [12], additional development is needed before continuous or impulsive control strategies can be replaced by such formations.

Previous studies have explored continuous and impulsive controllers for PFF through linear, nonlinear, and adaptive control techniques. An optimal nonlinear controller technique, Θ-D, was used by Xin et al. to bring relative position error into the submillimeter range [13, 14]. Li used a combination of linear quadratic regulator control with neural network learning to accommodate the nonlinearities in the basic dynamics and achieve range error levels in the order of tens of millimeters [15]. By using nonlinear control and robust adaptive methods, Xu was able to keep formation errors in the sub-kilometer range by estimating the spacecraft mass and bounds of the disturbances experienced [16]. Similar adaptive techniques used by Queiroz were able to estimate the mass and disturbances to reduce relative errors into the submeter range [17]. Work by Gurfil with nonlinear control techniques along with neural networks reduced relative errors into the submillimeter range [18]. Other continuous techniques include distributed adaptive synchronization control laws [19], Hamiltonian structure-preserving (HSP) controllers [20], and control systems using solar sails [21, 22]. While these methods show promise for long duration missions, relative position errors are typically in the meter range, exceeding typical requirements for PFF missions. While continuous controllers provide acceptable relative position performance, jitter and propellant budgets onboard the spacecraft are increased compared to impulsive methods. Work by Qi used an impulsive control strategy, which was able to maintain the spacecraft in a bounded relative position error corridor as small as 100 centimeters [23]. However, this case was limited to CR3BP dynamics without consideration of other disturbances outside of thruster errors. Ghafoor and Galchenko showed that an error corridor of 1.0 millimeters can be achieved using an event-triggered neural network control system using notional thrusters, but did not consider contributions from attitude dynamics [24].

Kane conducted initial research into the stability of attitude dynamics in the CR3BP, showing that stability is similar to that of two-body dynamics when considering the triangular equilibrium points. However, the collinear libration points have decreased stability margins, especially when considering the L_1 and L_2 equilibrium points [25], assuming the typical convention where L_1 is between the two primary bodies, L_2 is near the secondary (less massive) primary body, and L_3 is near the first primary body. These points are of special interest for distributed system missions, as missions such as LISA, TPF, and Stellar Imager seek to utilize Lissajous and halo orbits naturally found at these equilibrium points. Guzzeti expanded the research on stability, developing a general framework for attitude dynamics in the CR3BP as well as identifying trajectory design techniques using attitude dynamics [26, 27]. While variances in orbital position from perturbations resulted in minimal effects on attitude dynamics, relatively small variations in attitude from perturbations induced large variations in orbital dynamics. As the CR3BP often exhibits chaotic behavior, instabilities in spacecraft control can be quickly introduced if attitude dynamics are not considered. While most work on formation control at libration points has focused on only translational control, this research addresses both translation and attitude control in the CR3BP using two neural network control schemes with the intent to enable small spacecraft distributed system mission architectures in a deep space environment.

1.2. SCHOLARLY CONTRIBUTIONS

This research offers several contributions in the field of precision formation flight control in the CR3BP:

1. SNNARC, a two-layer neural network controller scheme, is reformulated for both the translational and rotational dynamics of a small spacecraft in the CR3BP.
2. MSO, originally a one-layer neural network observer and an input-output feedback linearization controller, is expanded to a two-layer neural network observer and is reformulated for both the translational and rotational dynamics of a small spacecraft in the CR3BP.
3. Simulation results show significant improvements in position and attitude performance metrics, along with cost savings in propellant consumption, using the two neural networks schemes as compared to traditional control techniques.

1.3. ORGANIZATION

This paper is organized as follows. In Section 2, a brief background on the dynamics and control theory is presented. Section 3 introduces neural networks and the design of two neural network control schemes. Section 4 discusses the problem formulation, simulation environment, and controller tuning. Section 5 presents simulation results of the two neural network control schemes compared to a classic proportional-integral-derivative control system. Section 6 summarizes the results and presents concluding remarks.

2. BACKGROUND

In this study, a notional deep space mission is considered at the Sun-Earth/Moon L_2 libration point. The translational and rotational dynamics and relevant perturbations in this environment that were considered are presented, along with a brief overview of relevant control theory.

2.1. TRANSLATIONAL DYNAMICS

In deep space mission analysis, a three-body dynamic model is often used, where two primaries are the dominate gravitational masses and the third mass is the spacecraft of interest. The general three-body problem remains unsolved analytically, and as such simplifying assumptions are typically made. First, it is assumed that the (infinitesimal) mass of the spacecraft does not produce a gravitational force sufficient to affect the motion of the primary gravitational bodies. Second, the two primary gravitational bodies are in circular orbits about their center of mass, the barycenter, which lies between the two primary bodies. By defining a rotating frame with the origin at the barycenter such that the $\hat{x}_s$ axis is oriented toward the secondary body and the $\hat{y}_s$ axis is normal to the $\hat{x}_s$ axis within the orbital plane, a synodic coordinate frame is formed. By formulating the dynamics in terms of this rotating frame, the classic CR3BP is formulated [28] and can be rewritten into a two-vector state system as

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \mathbf{x}_2 \\ \dot{\mathbf{x}}_2 &= f_{\mathbf{x}}(\mathbf{x}_1, \mathbf{x}_2) + \delta(\mathbf{x}) + g_{\mathbf{x}}\mathbf{u} \end{aligned} \tag{1}$$

where

$$f_{\mathbf{x}}(\mathbf{x}_1, \mathbf{x}_2) = \begin{bmatrix} 2\omega_s\dot{y} + \omega_s^2 x - \frac{\mu_1(x+r_{B1})}{r_1^3} - \frac{\mu_2\left(x-r_{B2}\right)}{r_2^3} \\ -2\omega_s\dot{x} + \omega_s^2 y - \frac{\mu_1 y}{r_1^3} - \frac{\mu_2 y}{r_2^3} \\ -\frac{\mu_1 z}{r_1^3} - \frac{\mu_2 z}{r_2^3} \end{bmatrix} \quad \mathbf{x}_1 = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad \mathbf{x}_2 = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \quad \mathbf{u} = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix}$$

$$r_1 = \left[\left(x + r_{B1}\right)^2 + y^2 + z^2\right]^{\frac{1}{2}} \qquad r_2 = \left[\left(x - r_{B2}\right)^2 + y^2 + z^2\right]^{\frac{1}{2}}$$

and where μ_1 and μ_2 are the gravitational parameters of the primary and secondary bodies, respectively, ω_s is the angular velocity of the rotating frame, r_{B1} is the distance from the barycenter to the primary body and r_{B2} is the distance from the barycenter to the secondary body, $\delta(\mathbf{x})$ is some known disturbance that perturbs the system, $g_{\mathbf{x}}$ maps the controller dynamics to the system, and $\mathbf{u}$ is the control acceleration. A graphical representation of the CR3BP is shown in Figure 1.

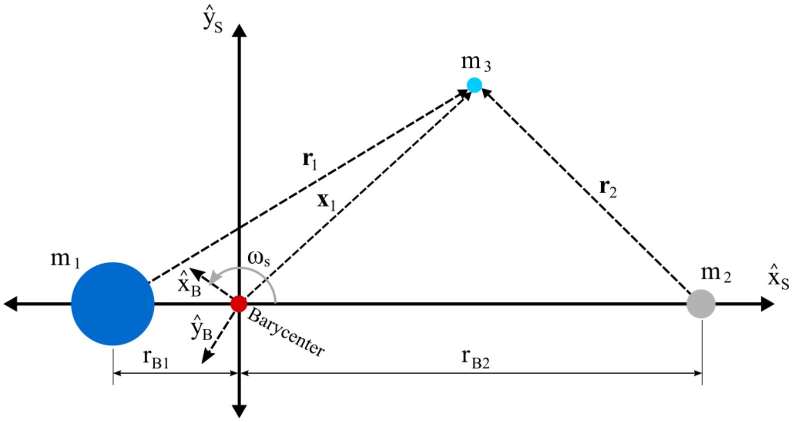


Figure 1. The rotating synodic coordinate frame, $[\hat{x}_s\ \hat{y}_s\ \hat{z}_s]$ is defined such that it rotates about the inertially fixed barycenter frame, $[\hat{x}_B\ \hat{y}_B\ \hat{z}_B]$, which allows for the simplification of the equations of motion for the spacecraft.

2.2. ROTATIONAL DYNAMICS

By using a quaternion attitude representation, the typical unconstrained second-order three degrees of freedom attitude dynamics model [29] can be given by a two-vector state system as

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \frac{1}{2}\Xi(\mathbf{x}_1)\mathbf{x}_2 \\ \dot{\mathbf{x}}_2 &= f_{\mathbf{x}}(\mathbf{x}_2) + \delta(\mathbf{x}) + g_{\mathbf{x}}\mathbf{u} \end{aligned} \qquad \text{where} \qquad \Xi(\mathbf{x}_1) = \begin{bmatrix} q_4 & -q_3 & q_2 \\ q_3 & q_4 & -q_1 \\ -q_2 & q_1 & q_4 \\ -q_1 & -q_2 & -q_3 \end{bmatrix} \tag{2}$$

and where $f_{\mathbf{x}}(\mathbf{x}_2) = -J^{-1}[\mathbf{x}_2 \times]J\mathbf{x}_2$, $\mathbf{x}_1 = [\, q_1 \quad q_2 \quad q_3 \quad q_4 \,]^{\mathrm{T}}$, $\mathbf{x}_2 = \left[\, \omega_\alpha \quad \omega_\beta \quad \omega_\gamma \,\right]^{\mathrm{T}}$, $\mathbf{u} = [\, u_\alpha \quad u_\beta \quad u_\gamma \,]^{\mathrm{T}}$, and where J is the inertia matrix (assumed constant) of the (rigid) body with respect to the center of mass and in terms of the body-fixed axes, $\mathbf{x}_1$ and $\mathbf{x}_2$ are the quaternion and angular velocity vectors, respectively, of the body with respect to the inertial frame, $\delta(\mathbf{x})$ is some known disturbance dynamics, $g_{\mathbf{x}}$ maps the controller dynamics to the system, and $\mathbf{u}$ is the applied torque, i.e. the control input. It should be noted that J^{-1} is also embedded into the disturbance dynamics $\delta(\mathbf{x})$ and the mapping of the controller dynamics, $g_{\mathbf{x}}$, in Eq. (2) to simplify the formulation.

2.3. SOLAR RADIATION PRESSURE DYNAMICS

Spacecraft trajectories in deep space can be significantly perturbed by solar radiation pressure (SRP) [28, 29]. One model for SRP is given by

$$P_{\mathrm{sun}} = \frac{\mathfrak{I}_{\mathrm{sun}}}{c\, r^2_{\mathrm{s}\rightarrow\odot}} \tag{3}$$

where $\Im_{\mathrm{sun}}$ is the solar constant at one astronomical unit, which ranges between 1,361 $\mathrm{W/m^2}$ and 1,363 $\mathrm{W/m^2}$ based on the solar cycle, c is the speed of light, and $r_{\mathrm{s}\rightarrow\odot}$ is the distance between the spacecraft and Sun in astronomical units. The spacecraft is modeled as having six uniform sides and the force generated by SRP from Eq. (3) on the i^{th} side is quantified by

$$\mathbf{F}^i_{\mathrm{SRP}} = -P_{\mathrm{sun}} A^i \left[2\left(\frac{R^i_{\mathrm{diff}}}{3} + R^i_{\mathrm{spec}} \cos\theta^i_{\mathrm{SRP}} \right) \mathbf{n}^i_B + (1 - R^i_{\mathrm{spec}}) \hat{\mathbf{r}}_{\mathrm{s}\rightarrow\odot} \right] \max(\cos\theta^{\mathrm{i}}_{\mathrm{SRP}}, 0) \tag{4}$$

where A^i is the area of the i^{th} side, R^i_{diff} is the coefficient of diffuse reflection, R^i_{spec} is the coefficient of spectral reflection, $\mathbf{n}^i_B$ is the normal outward unit vector of the i^{th} side, $\hat{\mathbf{r}}_{\mathrm{s}\rightarrow\odot}$ is the spacecraft-to-Sun unit vector expressed in terms of the body frame, and $\cos\theta^i_{\mathrm{SRP}} = \mathbf{n}^i_B \cdot \hat{\mathbf{r}}_{\mathrm{s}\rightarrow\odot}$. The addition of the $\max(\cos\theta^{\mathrm{i}}_{\mathrm{SRP}}, 0)$ term eliminates forces generated on the sides where the normal vector is orientated away from the Sun [28, 29].

The disturbance forces from Eq. (4) acting on the sides of the spacecraft from the SRP perturbations are then summed and the disturbance acceleration is given as

$$\delta(\mathbf{x}) = \begin{bmatrix} 0 & 0 & 0 & \sum_{i=1}^{N} F^i_{\mathrm{SRP},x} & \sum_{i=1}^{N} F^i_{\mathrm{SRP},y} & \sum_{i=1}^{N} F^i_{\mathrm{SRP},z} \end{bmatrix}^T M^{-1}_{s/c} \tag{5}$$

where N is the number of sides modeled on the spacecraft and $M_{s/c}$ is the mass of the spacecraft. In this study, the spacecraft is modeled as a rectangular prism where $N = 6$.

To quantify how SRP disturbance forces (using Eq. (4)) perturb the attitude of the spacecraft, the torque generated by SRP is calculated as

$$\mathbf{L}_{\mathrm{SRP}} = \sum_{i=1}^{6} \mathbf{r}^i_{\mathrm{press}} \times \mathbf{F}^i_{\mathrm{SRP}} \tag{6}$$

where $\mathbf{r}^i_{\text{press}}$ is the vector from the spacecraft center of mass to the center of pressure of the i^{th} side of the spacecraft [29]. It should be noted that for this SRP disturbance formulation, the translational and rotational dynamics are effectively coupled. Changes in attitude affect the magnitude and direction of the translational disturbance, while changes in translation affect the magnitude and direction of rotational disturbances.

2.4. CONTROL THEORY OVERVIEW

To develop a control algorithm for translational and attitude control, the dynamics are rewritten in terms of the error between the current and desired reference system states. The desired states, $\mathbf{x}_d$, are defined as $\mathbf{x}_d = [\ \mathbf{x}_{d1} \quad \mathbf{x}_{d2}\ ]^T$, where $\mathbf{x}_{d1}$ is a vector of the desired position or attitude quaternion and $\mathbf{x}_{d2}$ is a vector of the desired velocity or rotational rates. The reference system dynamics are given as $\dot{\mathbf{x}}_{d1} = \mathbf{x}_{d2}$ and $\dot{\mathbf{x}}_{d2} = f_d$, where f_d is a set of desired dynamics for the reference system. Note that for a regulatory reference system, $\mathbf{x}_{d2} = 0$ and $f_d = 0$. The error states for translational and attitude control are then defined as $\mathbf{e} = [\ \mathbf{e}_1 \quad \mathbf{e}_2\ ]^T$, where $\mathbf{e}_2 = \mathbf{x}_2 - \mathbf{x}_{d2}$ for both systems.

For translation, the error state for position is given by $\mathbf{e}_1 = \mathbf{x}_1 - \mathbf{x}_{d1}$, such that taking the derivative of the error vectors gives $\dot{\mathbf{e}} = [\ \dot{\mathbf{e}}_1 \quad \dot{\mathbf{e}}_2\ ]^T$ with

$$\begin{aligned} \dot{\mathbf{e}}_1 &= \mathbf{e}_2 \\ \dot{\mathbf{e}}_2 &= f_{\mathbf{x}}(\mathbf{x}) - f_d + \delta(\mathbf{x}) + g_{\mathbf{x}}\mathbf{u} \end{aligned} \tag{7}$$

defining the error dynamics used for the translational controller formulations.

For the quaternion attitude controller, $\mathbf{e}_1$ is defined as $\delta\mathbf{x}_1$, which is given as

$$\begin{aligned} \delta\mathbf{x}_{1,1:3} &= \Xi^T(\mathbf{x}_{d1})\mathbf{x}_1 \\ \delta x_{1,4} &= \mathbf{x}_1^T\mathbf{x}_{d1} \end{aligned} \qquad \text{where} \qquad \Xi^T(\mathbf{x}_{d1}) = \begin{bmatrix} q_{d4} & -q_{d3} & q_{d2} \\ q_{d3} & q_{d4} & -q_{d1} \\ -q_{d2} & q_{d1} & q_{d4} \\ -q_{d1} & -q_{d2} & -q_{d3} \end{bmatrix} \tag{8}$$

such that taking the derivative of $\mathbf{e}_1$ gives

$$\delta\dot{\mathbf{x}}_1 = \frac{1}{2}\Omega\left(\mathbf{x}_2\right)\delta\mathbf{x}_1 \qquad \text{where} \qquad \Omega\left(\mathbf{x}_2\right) = \begin{bmatrix} -\left[\mathbf{x}_2\times\right] & \mathbf{x}_2 \\ -\mathbf{x}_2^T & 0 \end{bmatrix} \tag{9}$$

where $[\mathbf{x}_2\times]$ is the skew symmetric cross product matrix of the angular velocities, $\mathbf{x}_2$. Using the definition of $\mathbf{e}_2$ and its derivative, the error dynamics for the rotational system can now be given as

$$\begin{aligned} \dot{\mathbf{e}}_1 &= \frac{1}{2}\Omega\left(\mathbf{x}_2\right)\mathbf{e}_1 \\ \dot{\mathbf{e}}_2 &= f_{\mathbf{x}}(\mathbf{x}_2) - f_d + \delta(\mathbf{x}) + g_{\mathbf{x}}\mathbf{u} \end{aligned} \tag{10}$$

defining the error dynamics used for the rotational controller formulations. While the error kinematics between the two systems differ somewhat, it is shown in the literature that using the definitions of $\mathbf{e}_1$ and $\mathbf{e}_2$ of the respective systems, stability can be shown with a candidate Lyapunov function using input-output feedback linearization [29]. While the remainder of the paper expresses the error dynamics in terms of Eq. (7), differences in the attitude formulation are mentioned as necessary.

It is known that a linear closed-loop system can be shown to be asymptotically stable if the closed-loop matrix is Hurwitz [30]. By using input-output feedback linearization and carefully designing a nonlinear controller in terms of the error dynamics in Eq. (7), it is possible to make the nonlinear system behave in a linear manner. A controller is chosen such that $\mathbf{u} = \frac{1}{g_{\mathbf{x}}}\left(-f_x(\mathbf{x}) + f_d - \delta(\mathbf{x}) - \mathbf{K}_p\mathbf{e}_1 - \mathbf{K}_d\mathbf{e}_2\right)$, where $\mathbf{K}_p$ and $\mathbf{K}_d$ are proportional and derivative controller gain vectors, respectively, and $|g_{\mathbf{x}}| > 0$. The closed-loop error dynamics in Eq. (7) are reduced to $\dot{\mathbf{e}} = A_{cl}\,\mathbf{e}$, where $A_{cl} = \begin{bmatrix} 0 & 1 \\ -\mathbf{K}_p & -\mathbf{K}_d \end{bmatrix}$, such that the origin of the unperturbed nonlinear system will be asymptotically stable if A_{cl} is Hurwitz. This method of nonlinear control is the basis from which the neural network controllers are designed.

3. NEURAL NETWORK DESIGN

In formulating the input-output feedback linearization controller, it was assumed that the model of $f_x(\mathbf{x})$ is "truth" and that $\delta(\mathbf{x})$ describes all other contributing disturbance dynamics to the system. As inaccuracies are inherent to all models and all disturbances cannot be accounted for, consideration of unmodeled dynamics must be made. While controlling nonlinear systems in the presence of unmodeled dynamics is an extensively studied subject [24, 29–33], this research focuses on the application of neural networks as they offer some unique advantages over other adaptive control methods.

The true dynamics can be expressed as $\mathcal{F}_{\mathbf{x}} = f_x(\mathbf{x}) + \delta(\mathbf{x}) + \Delta(\mathbf{x})$, where $f_x(\mathbf{x})$ and $\delta(\mathbf{x})$ are as described in Eq. (1) and Eq. (2) and $\Delta(\mathbf{x})$ is a set of all other unmodeled dynamics. Using neural networks the true dynamics can be represented by $\mathcal{F}_{\mathbf{x}} = W^T \sigma\left(V^T \phi\left(\mathbf{z}\right)\right) + \varepsilon$ for a two-layer network, where σ is some activation function, $\phi\left(\mathbf{z}\right)$ is the basis vector (a vector of inputs into the neural network), and W^T and V^T are the true weights that give the value of $\mathcal{F}_{\mathbf{x}}$ with some bounded approximation error, ε, through the universal function approximation properties of neural networks [34]. While a one-layer neural network can be used, a two-layer neural network has the benefit of satisfying the persistence of excitation condition [34]. As the true weighting matrices are unknown, the estimated dynamics, $\hat{\mathcal{F}}_{\mathbf{x}}$, are given as $\hat{\mathcal{F}}_{\mathbf{x}} = \hat{W}^T \sigma\left(\hat{V}^T \phi\left(\mathbf{z}\right)\right)$ for a two-layer network, where $\hat{W}^T$ and $\hat{V}^T$ are the approximated weights.

The two-vector state system from Eq. (1) can be rewritten into a common form as

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \mathbf{x}_2 \\ \dot{\mathbf{x}}_2 &= \mathcal{F}_{\mathbf{x}} + g_{\mathbf{x}}\mathbf{u} \end{aligned} \tag{11}$$

along with the error dynamics from Eq. (7) as

$$\begin{aligned} \dot{\mathbf{e}}_1 &= \mathbf{e}_2 \\ \dot{\mathbf{e}}_2 &= \mathcal{F}_{\mathbf{x}} - f_d + g_{\mathbf{x}}\mathbf{u} \end{aligned} \tag{12}$$

forming the foundation of the two neural network methodologies studied in this research. The first method operates in terms of the error dynamics, Eq. (12), and as such the neural network learning occurs in the control space, where the learning is driven by the tracking error. The second method introduces an observer for the state dynamics, Eq. (11), and as such the neural network learning occurs outside the control space, where the learning is driven by the observer error. Both methods can be described by a Lyapunov candidate function and are shown to be Ultimately Upper Bounded (UUB), demonstrating Lyapunov stability.

3.1. SNNARC FORMULATION

The first method, originally developed for robotic manipulators [34], is formulated from Galchenko's work on the Subarcsecond Neural Network Attitude Reference Controller (SNNARC) [33]. Whereas the previous formulation utilized a filtered tracking error and was applied to a specific attitude control problem, this updated formulation uses the original error dynamics from Eq. (12) and generates a control input in terms of a traditional proportional derivative formulation. The control input is given by

$$\mathbf{u} = \frac{1}{g_{\mathbf{x}}}\left(-\hat{\mathcal{F}}_{\mathbf{x}} + f_d - K\mathbf{e} + \mathbf{u}_r\right) \quad \text{where} \quad \mathbf{u}_r = -K_z\left(\left\|\hat{\Theta}\right\| + \Theta_m\right)\mathbf{e} \tag{13}$$

and where $|g_{\mathbf{x}}| > 0$ and K is a diagonal matrix such that $K = \begin{bmatrix} K_p & 0 \\ 0 & K_d \end{bmatrix}$, where K_p and K_d correspond to proportional and derivative gains. For the rotational system, the error, $\mathbf{e}$, is given by $\mathbf{e} = \begin{bmatrix} \mathbf{e}_{1,1:3} & \mathbf{e}_2 \end{bmatrix}^T$. A robustifying control component, $\mathbf{u}_r$, is added where K_z is some positive controller gain matrix and the weighting matrices are defined as $\Theta = \begin{bmatrix} V & 0 \\ 0 & W \end{bmatrix}$, $\hat{\Theta} = \begin{bmatrix} \hat{V} & 0 \\ 0 & \hat{W} \end{bmatrix}$, and $\|\Theta\| \leq \Theta_m$. Substituting the control function from Eq. (13) in the error dynamics from Eq. (12), the closed-loop system becomes

$$\dot{\mathbf{e}} = -K\mathbf{e} + \tilde{\mathcal{F}}_{\mathbf{x}} + \mathbf{u}_r \tag{14}$$

where $\tilde{\mathcal{F}}_{\mathbf{x}}$ is the estimation error defined by $\tilde{\mathcal{F}}_{\mathbf{x}} = \mathcal{F}_{\mathbf{x}} - \hat{\mathcal{F}}_{\mathbf{x}}$, or by $\tilde{\mathcal{F}}_{\mathbf{x}} = \tilde{W}^T \sigma \left(\tilde{V}^T \phi \left(\mathbf{z} \right) \right) + \varepsilon$ in terms of weights. The neural network weight update laws are selected as

$$\begin{aligned} \dot{\hat{W}} &= M \left(\hat{\sigma} - \hat{\sigma}' \hat{V}^T \phi \left(\mathbf{z} \right) \right) \mathbf{e}^T - \kappa \left\| \mathbf{e} \right\| M \hat{W} \\ \dot{\hat{V}} &= N \phi \left(\mathbf{z} \right) \mathbf{e}^T \hat{W}^T \hat{\sigma}' - \kappa \left\| \mathbf{e} \right\| N \hat{V} \end{aligned} \tag{15}$$

where $\phi \left(\mathbf{z} \right)$ is the basis vector, M and N are positive definite matrices that give the learning rate, $\hat{\sigma} = \sigma \left(\hat{V}^T \phi \left(\mathbf{z} \right) \right)$, $\hat{\sigma}' = \sigma' \left(\hat{V}^T \phi \left(\mathbf{z} \right) \right)$, and κ is a modification factor such that $\kappa > 0$.

To demonstrate the stability of the system, the candidate Lyapunov function is selected as $L = \frac{1}{2} \mathbf{e}^T \mathbf{e} + \frac{1}{2} tr \left\{ \tilde{W}^T M^{-1} \tilde{W} \right\} + \frac{1}{2} tr \left\{ \tilde{V}^T N^{-1} \tilde{V} \right\}$, where $tr \left\{ \right\}$ is the trace operator. By taking the time derivative of the Lyapunov function and using the definitions of $\mathbf{e}$ and the closed-loop form of $\dot{\mathbf{e}}$ in Eq. (14), and the weight update law in Eq. (15), stability in a compact set about the origin can be shown utilizing methods similar to those in the literature [30, 33]. This demonstrates that the Lyapunov function is UUB in both $\left\| \mathbf{e} \right\|$ and $\left\| \tilde{\Theta} \right\|$ and thus shows Lyapunov stability of the reformulated SNNARC method.

3.2. MSO FORMULATION

The second method, originally developed for estimating unmodeled dynamics [35–37] and used by Galchenko for attitude control of a gimbaled pointing system [33], is the Modified State Observer (MSO). The method utilizes an observer to estimate the dynamics and then uses the estimated dynamics in an input-output feedback linearization controller. While the original implementation for attitude control only utilized a one-layer neural network and a single state vector for estimation (that being the angular rate states), the updated formulation utilizes the full state for both translational and attitude dynamics as well as expanding the neural network estimator to a two-layer neural network. An observer for the state dynamics in Eq. (11) can be given as

$$\begin{aligned} \dot{\hat{\mathbf{x}}}_1 &= \hat{\mathbf{x}}_2 \\ \dot{\hat{\mathbf{x}}}_2 &= \hat{\mathcal{F}}_{\mathbf{x}} + g_{\mathbf{x}}\mathbf{u} + K_2\left[\lambda\left(\mathbf{x}_1 - \hat{\mathbf{x}}_1\right) + \mathbf{x}_2 - \hat{\mathbf{x}}_2\right] \end{aligned} \tag{16}$$

where $\hat{\mathbf{x}}$ is the estimated state, K_2 is some gain matrix, and λ is some scalar gain such that $\|K_2\| > \|\lambda I_{3\times3}\|$, where $I_{3\times3}$ is an identity matrix. Note that for the rotational system, $(\mathbf{x}_1 - \hat{\mathbf{x}}_1)$ is defined using $\boldsymbol{\delta}\hat{\mathbf{x}}_{1,1:3} = \Xi^T(\hat{\mathbf{x}}_1)\mathbf{x}_1$ as in Eq. (8). The dynamics, $\hat{\mathcal{F}}_{\mathbf{x}}$, can be approximated by a two-layer neural network in a manner similar to SNNARC. The state from Eq. (11) and the observer in Eq. (16) are rewritten in terms of observer error, $\mathbf{e}_a = \lambda\left(\mathbf{x}_1 - \hat{\mathbf{x}}_1\right) + \mathbf{x}_2 - \hat{\mathbf{x}}_2$, such that the observer closed-loop error dynamics are given as

$$\dot{\mathbf{e}}_a = -\lambda K_2\left(\mathbf{x}_1 - \hat{\mathbf{x}}_1\right) - \left(K_2 - \lambda I_{3\times3}\right)\left(\mathbf{x}_2 - \hat{\mathbf{x}}_2\right) + \tilde{\mathcal{F}}_{\mathbf{x}} \tag{17}$$

where $\tilde{\mathcal{F}}_{\mathbf{x}}$ is the estimation error defined by $\tilde{\mathcal{F}}_{\mathbf{x}} = \mathcal{F}_{\mathbf{x}} - \hat{\mathcal{F}}_{\mathbf{x}}$, or by $\tilde{\mathcal{F}}_{\mathbf{x}} = \tilde{W}^T\sigma\left(\tilde{V}^T\phi\left(\mathbf{z}\right)\right) + \varepsilon$ in terms of weights. The neural network weight update laws are selected as

$$\begin{aligned} \dot{\hat{W}} &= M\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\phi\left(\mathbf{z}\right)\right)\mathbf{e}_a^T - \kappa\left\|\mathbf{e}_a\right\| M\hat{W} \\ \dot{\hat{V}} &= N\phi\left(\mathbf{z}\right)\mathbf{e}_a^T\hat{W}^T\hat{\sigma}' - \kappa\left\|\mathbf{e}_a\right\| N\hat{V} \end{aligned} \tag{18}$$

where $\phi\left(\mathbf{z}\right)$ is the basis vector, M and N are positive definite matrices that give the learning rate, $\hat{\sigma} = \sigma\left(\hat{V}^T\phi\left(\mathbf{z}\right)\right)$, $\hat{\sigma}' = \sigma'\left(\hat{V}^T\phi\left(\mathbf{z}\right)\right)$, and κ is a modification factor such that $\kappa > 0$.

The stability of the system in Eq. (17) can be shown with the candidate Lyapunov function $L = \frac{1}{2}\mathbf{e}_a^T\mathbf{e}_a + \frac{1}{2}tr\left\{\tilde{W}^T M^{-1}\tilde{W}\right\} + \frac{1}{2}tr\left\{\tilde{V}^T N^{-1}\tilde{V}\right\}$, where $tr\left\{\right\}$ is the trace operator. By taking the time derivative of the Lyapunov function and using the definitions of $\mathbf{e}_a$ and $\dot{\mathbf{e}}_a$ in Eq. (17), and the weight update law in Eq. (18), stability in a compact set about the origin can be shown utilizing methods similar to those in the literature [30, 33]. This demonstrates that the Lyapunov function is UUB in both $\|\mathbf{e}_a\|$ and $\left\|\tilde{\Theta}\right\|$ and thus shows Lyapunov stability of the observer in the proposed MSO method.

Once the nonlinear dynamics, $\hat{\mathcal{F}}_{\mathbf{x}}$, are estimated by the neural network, an input-output feedback linearization controller given by

$$\mathbf{u} = \frac{1}{g_{\mathbf{x}}}\left(-\hat{\mathcal{F}}_{\mathbf{x}} + f_d - \mathbf{K}_p\mathbf{e}_1 - \mathbf{K}_d\mathbf{e}_2\right) \tag{19}$$

can be applied to the two-vector system in Eq. (12) such that the closed-loop system dynamics become

$$\begin{aligned} \dot{\mathbf{e}}_1 &= \mathbf{e}_2 \\ \dot{\mathbf{e}}_2 &= -\mathbf{K}_p\mathbf{e}_1 - \mathbf{K}_d\mathbf{e}_2 + \tilde{W}^T\sigma\left(\tilde{V}^T\phi\left(\mathbf{z}\right)\right) + \varepsilon \end{aligned} \quad \longrightarrow \quad \dot{\mathbf{e}} = A_{cl}\,\mathbf{e} + d$$
$$\text{where} \quad A_{cl} = \begin{bmatrix} 0 & 1 \\ -\mathbf{K}_p & -\mathbf{K}_d \end{bmatrix} \quad \text{and} \quad d = \begin{bmatrix} 0 \\ \tilde{W}^T\sigma\left(\tilde{V}^T\phi\left(\mathbf{z}\right)\right) + \varepsilon \end{bmatrix} \tag{20}$$

where d is bounded by $\|d\| \leq \|d_m\|$. By using the candidate Lyapunov function $L = \frac{1}{2}\mathbf{e}^T P\mathbf{e}$ and taking its time derivative using the definition of $\dot{\mathbf{e}}$ in Eq. (20) and where P satisfies the equation $A^T P + PA = -Q$, stability in a compact set about the origin can be shown if $\|\mathbf{e}\| \geq 2\frac{\lambda_{max}(P)d_m}{\lambda_{min}(Q)}$, demonstrating that both Lyapunov functions are UUB in $\|\mathbf{e}\|$, $\|\mathbf{e}_a\|$, and $\left\|\tilde{\Theta}\right\|$ and thus shows Lyapunov stability of the proposed MSO method.

4. PROBLEM FORMULATION

A number of distributed system mission concepts can typically be described by a virtual structure. The typical virtual structure has a central node, defined as the reference node (i.e., the leader), and a number of nodes defined by fixed locations from the reference node. The shape of the virtual structure can be defined as desired; for example, the Stellar Imager mission proposes a parabolic virtual structure to emulate a parabolic telescope mirror [8]. This structure remains fixed over periods of science observations and its reference node follows the natural motion of some desired trajectory, such as a halo orbit. Each spacecraft can then be assigned to a relative node by which they maintain their position. By maintaining their position, formation flight is realized. The virtual structure concept is shown in Figure 2, where n_{ref} is the reference node and where n_1, n_2, ... are relative nodes at fixed locations, given by r_1, r_2, ..., respectively, from the reference node. As the structure is virtual, the shape can change based on science requirements and objectives; e.g., the Stellar Imager mission proposes various separation distances to vary the focal length of the virtual parabolic telescope [38].

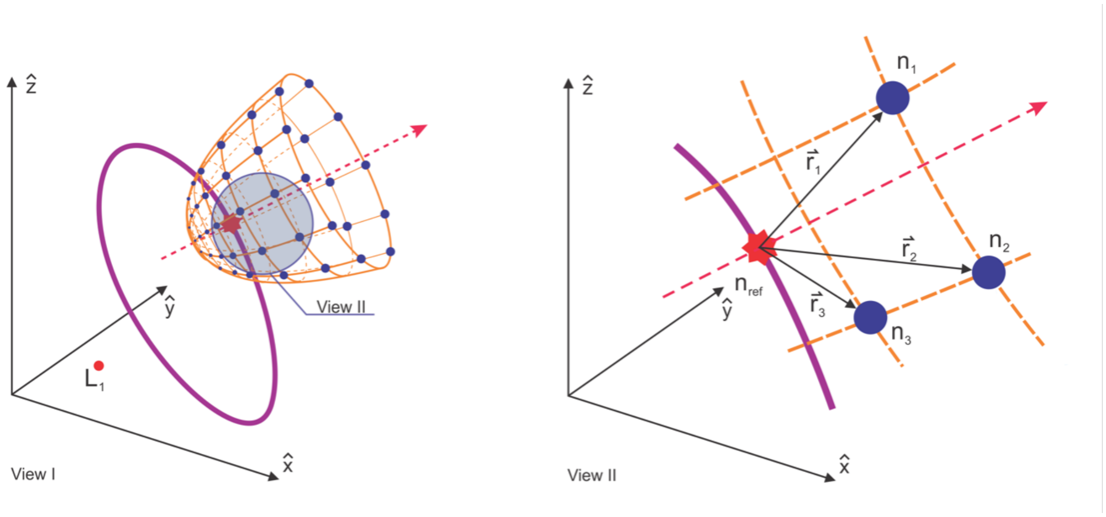


Figure 2. The fixed virtual structure (View I) is defined with the reference node on the halo trajectory, with relative nodes (View II) placed at fixed locations from the reference node. By placing and maintaining spacecraft at these relative nodes, formation flight is realized.

The case study defined considers the Stellar Imager mission concept [38] to formulate a realistic mission baseline. Formation flight at the Sun-Earth/Moon L_2 libration point is considered, with a halo orbit as the reference trajectory. A reference virtual node, defined as the leader of a formation of an arbitrary number of spacecraft, is placed along the reference trajectory. The leader spacecraft stationkeeps its position at this reference node. For the remaining spacecraft in the formation, the relative nodes are fixed vectors from the leader spacecraft. As such, each spacecraft in the virtual structure follows the motion of the leader instead of the reference node. However, the virtual structure matches the nominal definition if the control objective is achieved. The follower spacecraft will maintain some defined relative position from the leader spacecraft. Following the Stellar Imager design, a relative node was placed at a fixed separation of 5000 meters along the rotating $\hat{x}$-axis and 250 meters along the $\hat{y}$-axis and $\hat{z}$-axis from the leader spacecraft. This formation can be easily expanded to any number of spacecraft, each with their own relative position from the leader spacecraft, such that a virtual structure is formed. Using a leader spacecraft and a single follower spacecraft to demonstrate the relative position and attitude control is sufficient to show the concept feasibility, and it is a key objective of this study enabled by the two neural network schemes proposed.

4.1. REFERENCE TRAJECTORY DESIGN

The halo orbit is a natural orbit that exists within the context of the CR3BP, with stationkeeping methods needing to be applied occasionally to maintain the orbit. While the orbit is "unnatural" in higher fidelity models such as the elliptic restricted three-body problem or an ephemeris-based model, a separate dynamical representation of the reference trajectory is not needed in this study because the system dynamics are modeled as "truth" in the CR3BP. As such, the reference trajectory is determined using the CR3BP EOMs given by

$$\begin{aligned} \dot{\mathbf{x}}_{r_1} &= \mathbf{x}_{r_2} \\ \dot{\mathbf{x}}_{r_2} &= f_r(\mathbf{x}_{r_1}, \mathbf{x}_{r_2}) \end{aligned} \qquad \text{where} \qquad \mathbf{x}_{r_1} = \begin{bmatrix} x_r \\ y_r \\ z_r \end{bmatrix} \quad \mathbf{x}_{r_2} = \begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{z}_r \end{bmatrix}$$

$$\text{and} \quad f_r(\mathbf{x}_{r_1}, \mathbf{x}_{r_2}) = \begin{bmatrix} 2\omega_s \dot{y}_r + \omega_s^2 x_r - \frac{\mu_1 (x_r + r_{B1})}{r_{r_1}^3} - \frac{\mu_2 \left(x_r - r_{B2}\right)}{r_{r_2}^3} \\ -2\omega_s \dot{x}_r + \omega_s^2 y_r - \frac{\mu_1 y_r}{r_{r_1}^3} - \frac{\mu_2 y_r}{r_{r_2}^3} \\ -\frac{\mu_1 z_r}{r_{r_1}^3} - \frac{\mu_2 z_r}{r_{r_2}^3} \end{bmatrix} \tag{21}$$

such that the propagated dynamics of the spacecraft only differ in the addition of a control input and using the true state instead of the reference. The initial conditions for the reference trajectory are numerically computed using the well-known differential corrector approach that generates a halo orbit when propagated forward in time.

This study considers a naturally occurring halo orbit in the CR3BP at the Sun-Earth/Moon L$_2$ point. The error dynamics for the leader spacecraft is given as

$$\mathbf{e}_1 = \mathbf{x}_1 - \mathbf{x}_{r_1} \qquad \text{and} \qquad \mathbf{e}_2 = \mathbf{x}_2 - \mathbf{x}_{r_2} \tag{22}$$

where $\mathbf{x}_r$ is the state of the reference node of the virtual structure in terms of the rotating barycentric reference frame. The error dynamics for the i^{th} follower spacecraft is given as

$$\mathbf{e}_1^i = \mathbf{x}_1^i - \mathbf{x}_{r_1}^l - \mathbf{r}_{r_1}^i \qquad \text{and} \qquad \mathbf{e}_2^i = \mathbf{x}_2^i - \mathbf{x}_{r_2}^l \tag{23}$$

where $\mathbf{x}_r^l$ is the state of the leader spacecrat and $\mathbf{r}_{r_1}^i$ is the fixed position vector from the leader spacecraft to the i^{th} relative node of the virtual structure.

4.2. SPACECRAFT AND THRUSTER MODEL

The spacecraft considered in this study are modeled after the preliminary design presented in the Stellar Imager final report [38]. For a typical spacecraft, the center of pressure is generally close to the geometrical centroid or designed to be near the center of mass [39]. However, the design of the spacecraft for Stellar Imager requires large solar panels and deployable communication arrays, which can significantly affect the center of pressure location. Spacecraft models for the leader and follower used in this study are shown in Figure 3, where the body-fixed $\hat{b}_{1c}$-axis of the leader/chief spacecraft is always pointed away from the Sun, while the body-fixed $\hat{b}_{1f}$-axis of the follower spacecraft is continuously oriented towards the leader spacecraft. The center of pressure is defined by the geometry of each spacecraft's model, while the center of mass is located at the geometrical center of the primary hubs (yellow in hue). An additional 50 millimeter deviation to the center of mass is added from the centroid along each i^{th} spacecraft's side to account for typical design tolerances. Depending on the spacecraft orientation relative to the Sun, the center of pressure to center of mass distance can be as high as 0.2 meters for the follower spacecraft. The force and torque from Eq. (5) and Eq. (6), respectively, generates a conservatively high force and torque disturbance to demonstrate the efficacy of the proposed neural network schemes.

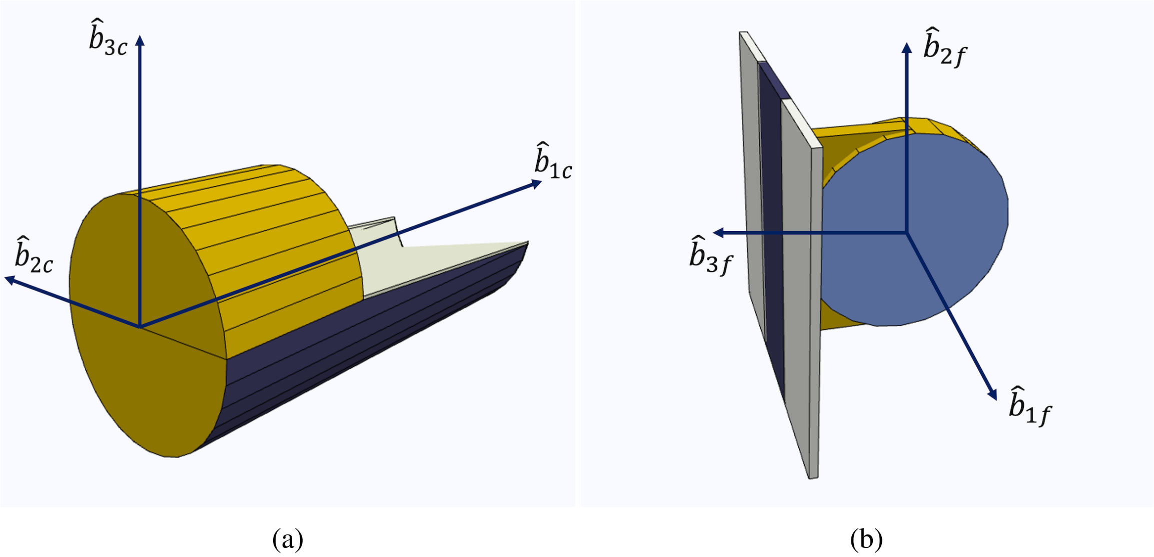


Figure 3. The leader spacecraft (a) points its primary body $\hat{b}_{1c}$-axis anti-Sun while the follower spacecraft (b) points its primary $\hat{b}_{1f}$-axis at the leader spacecraft and rotates about $\hat{b}_{1f}$-axis such that the secondary $\hat{b}_{3f}$-axis points towards the Sun as closely as possible.

Marchand and Howell have shown that for tight precision control, thrusters need to be capable of producing thrust in the nano to millinewtons range [11]. Thruster systems are in development that have the potential of producing thrust in the nano to micronewton range (such as plasmonic force propulsion [31, 32, 40, 41]), but this study considers the 10s to 100s of micronewtons range as such technology is readily available and has flight heritage [3]. Here, a single thruster capable of producing between 10 to 350 micronewtons of thrust was implemented in simulations, where the control effort either becomes saturated if it exceeds the thruster capability or does not actuate if the desired control is below the minimum thrust value. The proposed Stellar Imager design [38] used a configuration of eight thrusters to provide both net-zero torque for translational control and net-zero force for attitude control. While the baseline mission also considers momentum wheels, this study shows that the mission objectives can be achieved using only the micronewton thrusters.

4.3. BASELINE PID CONTROLLER DESIGN

To show the viability and benefits of using the two neural network schemes, a PID control system is defined as a baseline for performance comparisons. The system can then be described by the plant $P(s) = M_{s/c}^{-1}/s^2$, where by choosing a PID control input, the closed-loop characteristic equation is given by $s^3 + [M_{s/c}^{-1}K_d]s^2 + [M_{s/c}^{-1}K_p]s + [M_{s/c}^{-1}K_i]$, where K_p, K_i, and K_d are proportional, integrator, and derivative gains, respectively. Using the typical (ζ, ω_n) parametrization and fixing the third pole at $-\alpha\omega_n$, the desired closed-loop characteristic equation is given by $s^3 + [(2\zeta + \alpha)\omega_n]s^2 + [(2\zeta\alpha + 1)\omega_n^2]s + [\alpha\omega_n^3]$ where ω_n is the natural frequency of the system, ζ is the damping ratio, and α is a design parameter. The controller gains can now be selected to match the design characteristic equation with

$$K_p = (2\zeta\alpha + 1)\omega_n^2 M_{s/c} \qquad K_i = \alpha\omega_n^3 M_{s/c} \qquad K_d = (2\zeta + \alpha)\omega_n M_{s/c} \tag{24}$$

where by fixing the design parameter at $\alpha = 1$, the values for ζ and ω_n are tuned to give desired performance characteristics. It is noted that for $\alpha = 0$, the gain tuning becomes that of a proportional-derivative (PD) controller for the same system.

4.4. PID TUNING

The baseline PID system was tuned until the desired performance was achieved, with relative position errors being submillimeter and attitude errors in the arcsecond range. It was found that by fixing two parameters at $\alpha = 1$ and $\zeta = 0.9$, that desired performance could be achieved by tuning only the natural frequency, ω_n. For the leader spacecraft, the natural frequency was $\omega_n = 4 \times 10^{-4}$ for translation and $\omega_n = 4 \times 10^{-2}$ for attitude. For the follower spacecraft, the natural frequency was given by $\omega_n = 4 \times 10^{-3}$ for translation and $\omega_n = 4 \times 10^{-2}$ for attitude. Using these parameters and the tuning laws defined in Eq. (24), the K_p, K_i, and K_d gains are selected.

4.5. SNNARC TUNING

To approximate a similar bandwidth between the various control methods, the two neural networks tuning process was based on the PID tuning method. For each spacecraft, the α, ζ, and ω_n parameters were selected as in the PID method to get the equivalent K_p and K_d gains. The basis vector for the translational system was selected as $\phi(\mathbf{z}) = [x \quad y \quad z \quad \dot{x} \quad \dot{y} \quad \dot{z}]^T$. The basis vector for the attitude system was selected as $\phi(\mathbf{z}) = \begin{bmatrix}\theta_\alpha & \theta_\beta & \theta_\gamma & \omega_\alpha & \omega_\beta & \omega_\gamma\end{bmatrix}^T$ where α corresponds to the roll channel, β corresponds to the pitch channel, and γ corresponds to the yaw channel. The neural network weights for $\hat{W}$ and $\hat{V}$ were initialized as null matrices. While the input and output dimensions of the weights were fixed, a two-layer neural network allows for an infinite number of hidden nodes. For this application, it was found that ten hidden nodes gave sufficient performance without significantly affecting the CPU performance. The *sigmoid* activation function was chosen, where $\sigma(x) = \frac{1}{1+e^{-x}}$.

The neural network portion was then tuned until sufficient performance was achieved. For the leader spacecraft, the translational gains were set to $M = 8 \times 10^{-4}$, $N = 8 \times 10^{-4}$, $\kappa = 0.1$, $\Theta_m = 1 \times 10^{-10}$, and $K_z = 0.1$, while for attitude they were set to $M = 1 \times 10^{-7}$, $N = 1 \times 10^{-7}$, $\kappa = 0.1$, $\Theta_m = 1 \times 10^{-10}$, and $K_z = 0.1$. For the follower spacecraft, the translational gains were set to $M = 2 \times 10^{-4}$, $N = 2 \times 10^{-4}$, $\kappa = 0.1$, $\Theta_m = 1 \times 10^{-10}$, and $K_z = 0.1$, while for attitude they were set to $M = 1 \times 10^{-8}$, $N = 1 \times 10^{-8}$, $\kappa = 0.1$, $\Theta_m = 1 \times 10^{-10}$, and $K_z = 0.1$.

4.6. MSO TUNING

The tuning of the MSO neural network scheme's input-output feedback linearization portion was matched with the PID method, were α, ζ, and ω_n were kept constant such that the K_p and K_d values matched for both spacecraft between the PID and MSO methods. For the attitude controller, the K_i gain was slowed by a factor of 10 to allow the observer to learn before the integrator error could accumulate. At this stage, both methods produced identical results with the observer disabled.

The MSO neural network portion of the algorithm was setup in a similar manner to SNNARC so that a meaningful comparison could be made. As such, the same basis vector, $\phi(\mathbf{z})$, was used for both translation and attitude and the neural network weights, $\hat{W}$ and $\hat{V}$, were initialized as null matrices with ten hidden nodes. The neural network portion was then tuned until the estimated state converged with the measured state. The estimated dynamics were then fed into the controller, completing the tuning process for the MSO scheme. For the leader spacecraft, the translation gains were set to $K_2 = 1 \times 10^{-1}$, $M = 1 \times 10^{-2}$, $N = 1 \times 10^{-2}$, $\kappa = 1 \times 10^{-2}$, and $\lambda = 1 \times 10^{-3}$, while for attitude they were set to $K_2 = 1 \times 10^{-2}$ $M = 2 \times 10^{-8}$, $N = 2 \times 10^{-8}$, $\kappa = 1 \times 10^{-2}$, and $\lambda = 1 \times 10^{-3}$. For the follower spacecraft, the translation gains were set to $K_2 = 1 \times 10^{-1}$, $M = 1 \times 10^{-2}$, $N = 1 \times 10^{-2}$, $\kappa = 1 \times 10^{-2}$, and $\lambda = 1 \times 10^{-3}$, while for attitude they were set to $K_2 = 1 \times 10^{-1}$ $M = 1 \times 10^{-8}$, $N = 1 \times 10^{-8}$, $\kappa = 1 \times 10^{-2}$, and $\lambda = 1 \times 10^{-3}$.

5. SIMULATION RESULTS

To demonstrate the functionality of the two neural network schemes, the trajectories of the leader spacecraft, following the reference node, and the follower spacecraft, following the leader spacecraft, were simulated for the duration of one complete halo orbit (approximately 182 days). Even though the force generated by SRP is only on the order of tens of nanonewtons, this disturbance causes a large deviation in both the leader and follower spacecraft trajectories when the control system was not active, as shown in Figure 4. It is interesting to note (and not surprisingly) that even though the two spacecraft are in relative close proximity to each other at the start of the uncontrolled simulation (approximately 5 km separation), the differential forces experienced due to different spacecraft masses and geometries result in completely different trajectories after a single orbit. This helps illustrate the chaotic dynamics of this nonlinear system and demonstrates the need for precision control methods to achieve submillimeter control needed for advanced mission concepts.

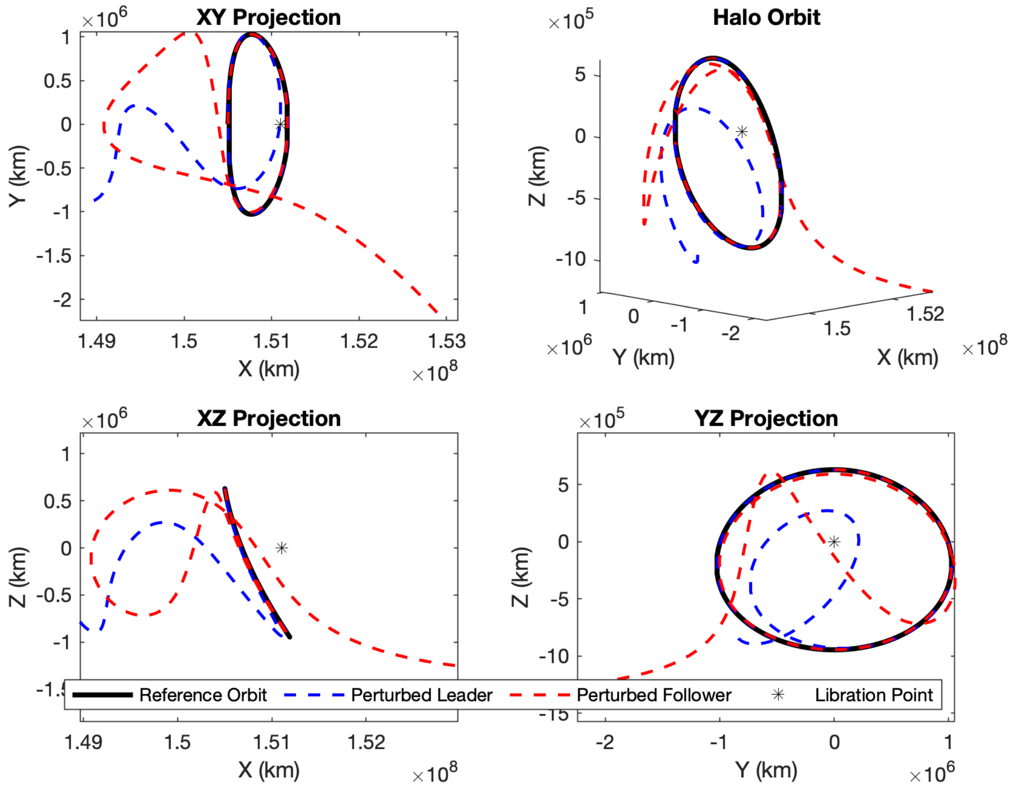
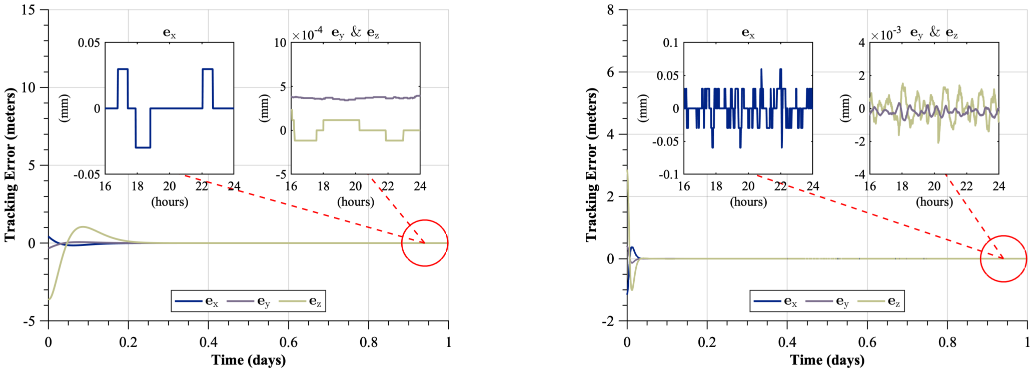


Figure 4. The leader spacecraft (blue) and follower spacecraft (red) with inactive control systems results in diverging trajectories compared to the desired halo trajectory (black) due to SRP.

To evaluate the various control schemes, identical initial conditions were used with the control tuning described. Both spacecraft were initiated with an initial error of approximately five meters from the desired relative position and up to five degrees error in roll, pitch, and yaw as compared to desired. Both spacecraft trajectories were simulated for a duration of one day, a time period in which all methods were able to achieve steady state performance. In addition to measuring the translation and pointing performance, the propellant consumption was also computed.

5.1. PID RESULTS

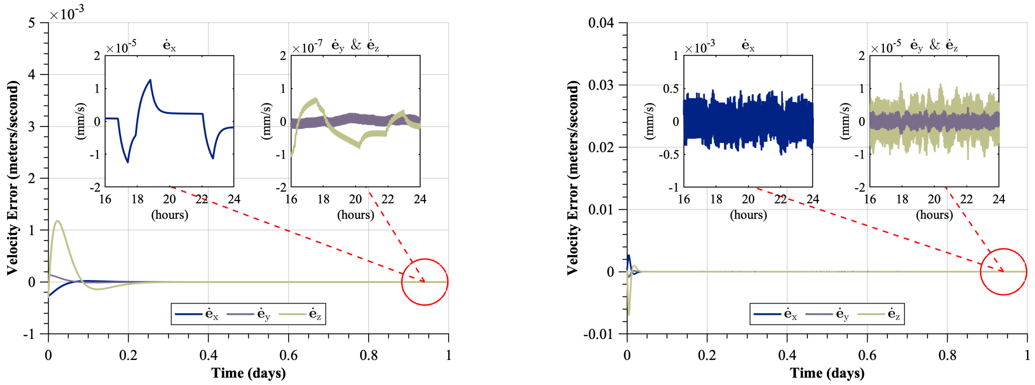
The simulation results when implementing the PID control scheme are presented first to establish a baseline for comparison. The performance of both spacecraft for position and velocity are presented in Figure 5 and Figure 6, respectively. It can be seen that both



(a)

(b)

Figure 5. Relative position error performance for the leader spacecraft (a) and follower spacecraft (b) using the PID control system.

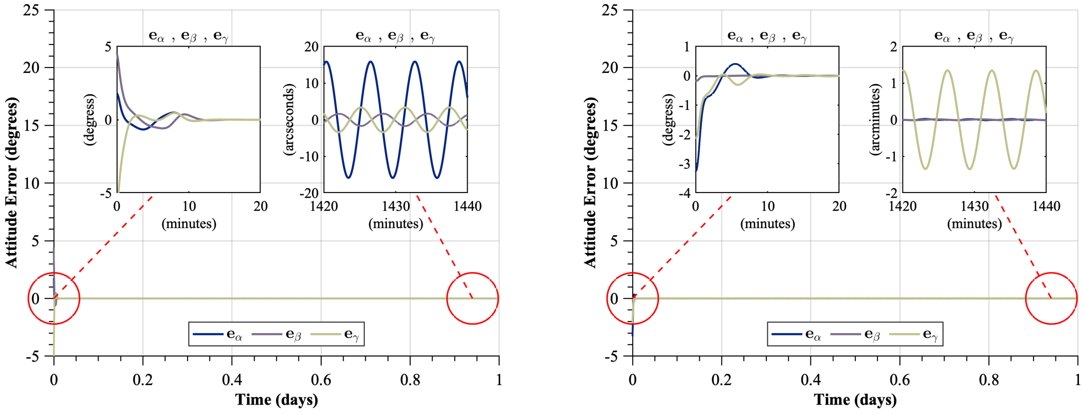


(a)

(b)

Figure 6. Relative velocity error performance for the leader spacecraft (a) and follower spacecraft (b) using the PID control system.

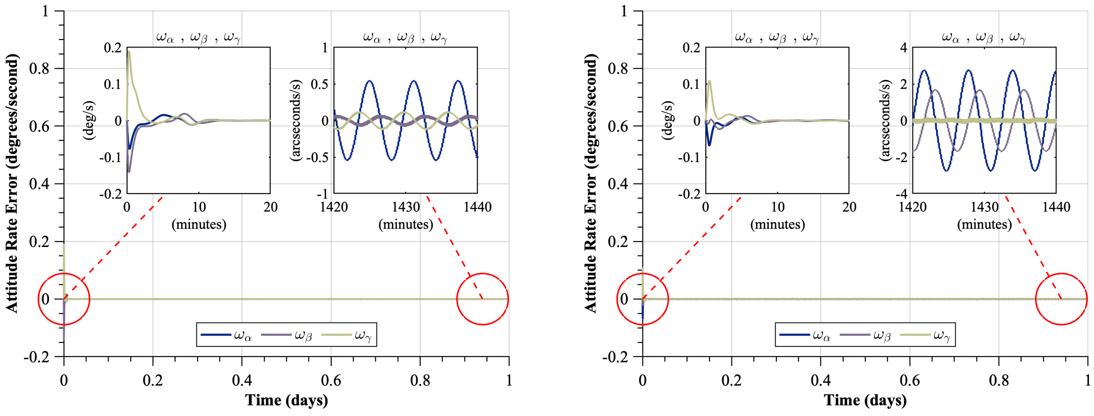
spacecraft can maintain a submillimeter relative position using the PID system. The numerical tolerance of the MATLAB simulation platform is quickly reached for the translational control along the $\hat{x}$-axis. This is due to the scale of the CR3BP primaries. The position along the $\hat{x}$-axis is on the order of 1×10^8 kilometers and as MATLAB retains only 16 digits of precision, the minimum measurable distance is approximately 0.02 millimeters. As such, the results appear step-like when assessing the relative position control of the two spacecraft in Figure 5. Though numerical precision is reached, some important conclusions can still be reached. First, though it is shown that submillimeter control can be reached with the classical PID system, it can be seen that there remains a significant amount of jitter in the velocity states. The total ΔV consumed for translation control over a ten day period was 4.0922×10^{-2} m/s for the leader spacecraft and 1.4460×10^{-1} m/s for the follower spacecraft.



(a) (b)

Figure 7. Attitude error performance for the leader spacecraft (a) and follower spacecraft (b) using the PID control system.

The performance of both spacecraft for attitude and angular velocities are presented in Figure 7 and Figure 8, respectively. It can be seen that the leader spacecraft is able to achieve tens of arcseconds of error range, while the follower spacecraft could only maintain an arcminute error range. The total ΔV consumed for attitude control over the one day period was 5.6465×10^{-3} m/s for the leader spacecraft and 1.0983×10^{-1} m/s for the follower spacecraft.



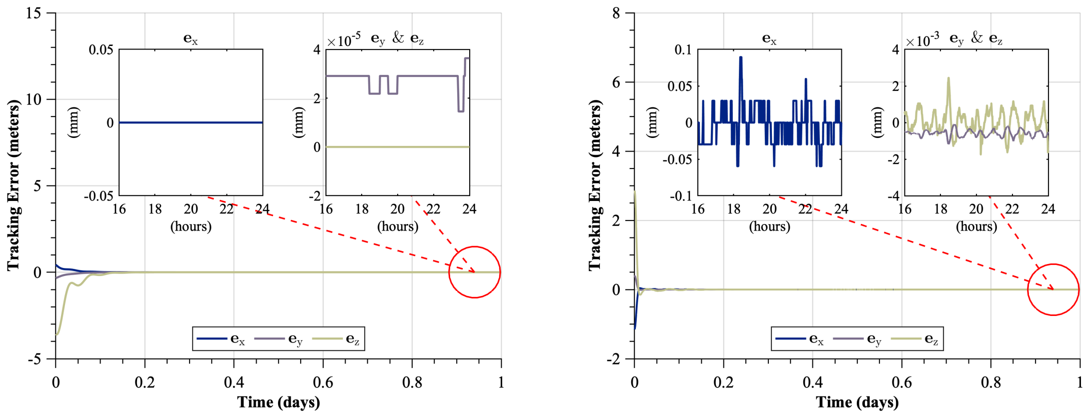
(a)

(b)

Figure 8. Attitude rate error performance for the leader spacecraft (a) and follower spacecraft (b) using the PID control system.

5.2. SNNARC RESULTS

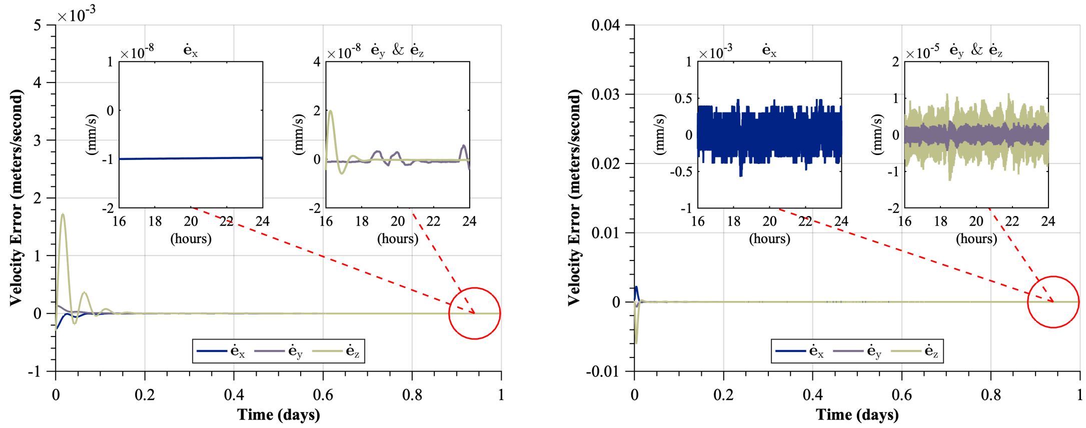
The simulation results when implementing the SNNARC control scheme are presented for both spacecraft for position and velocity in Figure 9 and Figure 10, respectively. It can be seen that both spacecraft can maintain submillimeter relative position and reach the numerical precision limits of the simulation framework, similar to the PID system. The total ΔV consumed for translational control over a ten day period was 4.3297×10^{-2} m/s for the leader spacecraft and 1.4190×10^{-1} m/s for the follower spacecraft. While the translational performance and ΔV results appear similar to the PID method, it was expected that the SNNARC method would have improved performance if the precision of the simulation could be increased. This is suggested by the velocity states in Figure 10, where the SNNARC method has a reduced amount of jitter in the leader spacecraft as compared to the PID method in Figure 6.



(a)

(b)

Figure 9. Relative position error performance for the leader spacecraft (a) and follower spacecraft (b) using the SNNARC control system.

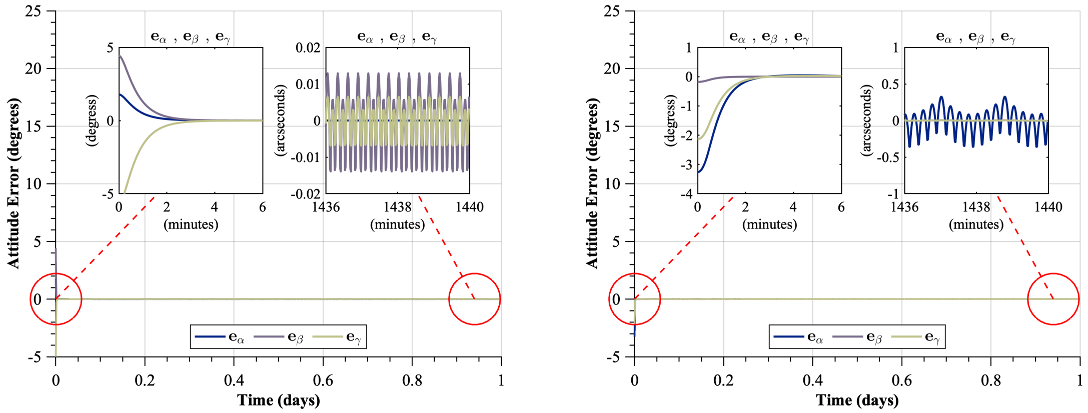


(a)

(b)

Figure 10. Relative velocity error performance for the leader spacecraft (a) and follower spacecraft (b) using the SNNARC control system.

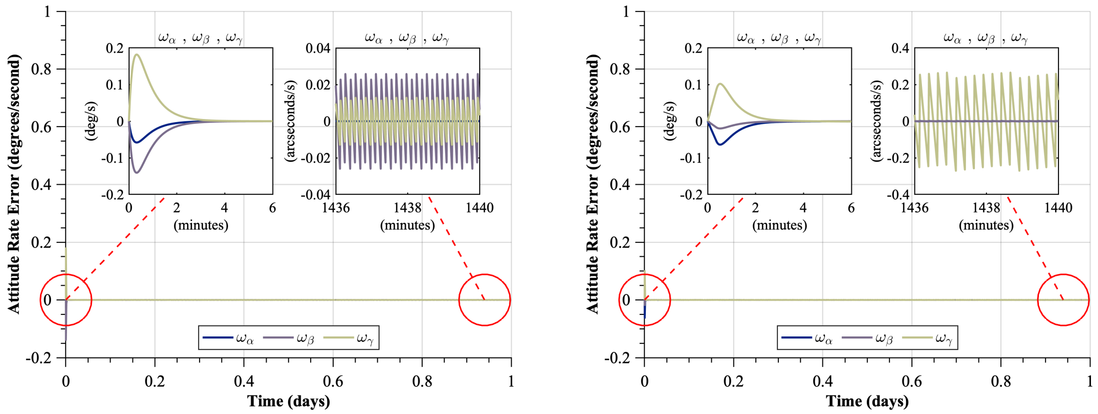
The difference in performance is clearly observable in the attitude control performance, where numerical precision tolerances were not encountered. The performance of both spacecraft in attitude and angular velocities is presented in Figure 11 and Figure 12, respectively. It can be seen that both the leader and follower spacecraft are able to achieve subarcseconds of error, significantly outperforming the PID method. The total ΔV consumed for attitude control over a ten day period was 4.8545×10^{-3} m/s for the leader spacecraft and 3.7213×10^{-2} m/s for the follower spacecraft, a significant reduction in propellant consumption compared to the PID method.



(a)

(b)

Figure 11. Attitude error performance for the leader spacecraft (a) and follower spacecraft (b) using the SNNARC control system.

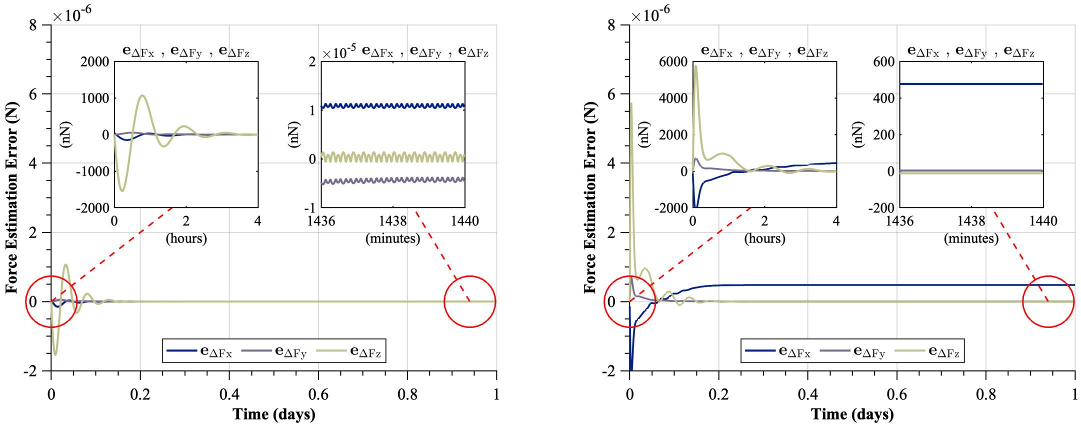


(a)

(b)

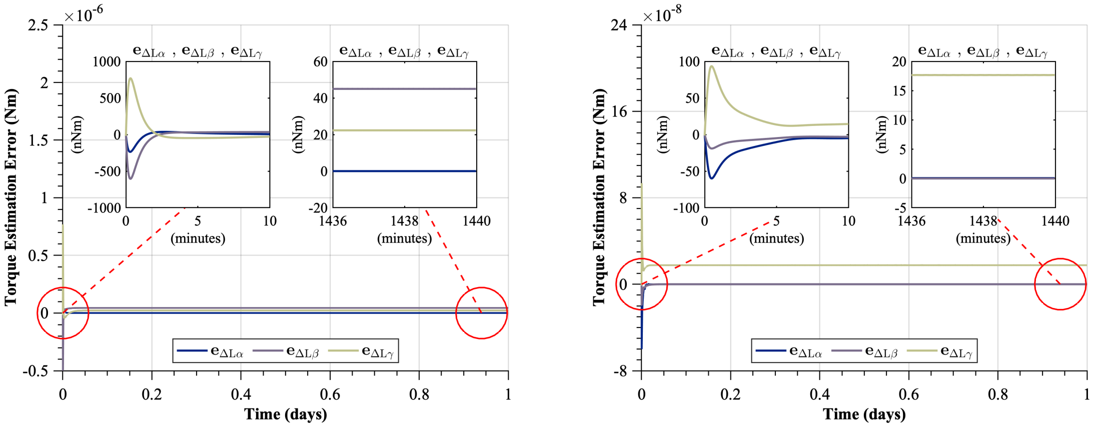
Figure 12. Attitude rate error performance for the leader spacecraft (a) and follower spacecraft (b) using the SNNARC control system.

The disturbance estimation performance results from the neural network portion of the SNNARC control scheme are presented for both spacecraft for force and torque perturbations in Figure 13 and Figure 14, respectively. It is seen that, in general, the perturbation estimation does not appear to match the expected values other than for the



(a) (b)

Figure 13. Force perturbation estimation error performance for the leader spacecraft (a) and follower spacecraft (b) using the SNNARC neural network.



(a) (b)

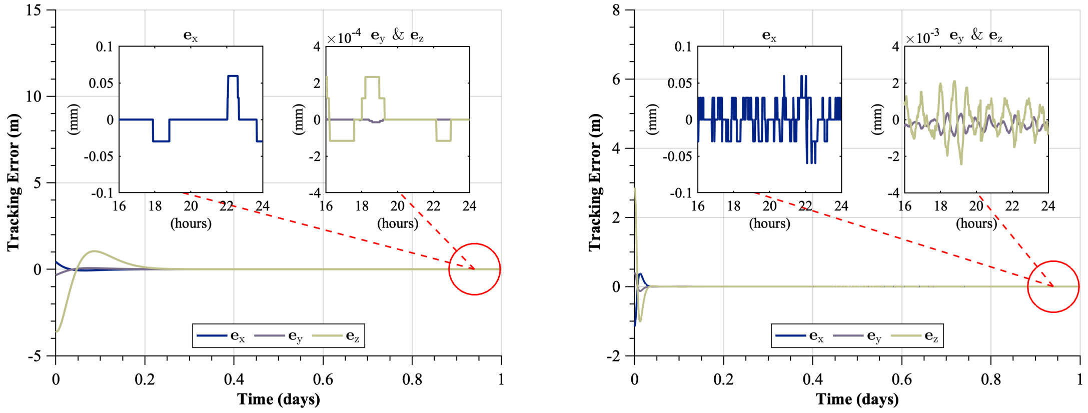
Figure 14. Torque perturbation estimation error performance for the leader spacecraft (a) and follower spacecraft (b) using the SNNARC neural network.

position of the leader spacecraft. This is the result of how the leader and reference trajectories are defined. For the leader spacecraft, its desired reference dynamics match the spacecraft's simulated states when the error is null. However, this is not the case for the follower spacecraft. Because the reference dynamics are defined by a fixed distance from the leader spacecraft, the "natural" velocity from those dynamics would result in the spacecraft deviating from the desired trajectory. As these are the reference dynamics that are utilized, the neural network compensates for the mismatch. This is because the SNNARC methodology learns based on the tracking error. As such, the bias introduced is fixed by estimating it as an additional perturbation. This is also the case for the attitude perturbation estimation observed in Figure 14, where the assumed desired attitude dynamics and desired angular velocities are null. However, there can be small variances in true desired angular velocities, near null, that are in part coupled with the performance of the translational controller. The neural network, being fed the attitude tracking error, is able to compensate for the bias by estimating an additional perturbation in attitude. This phenomenon is further explored for both translation and attitude using the MSO control scheme, where an integrator corrects for the bias while the observer is better able to estimate the perturbations.

5.3. MSO RESULTS

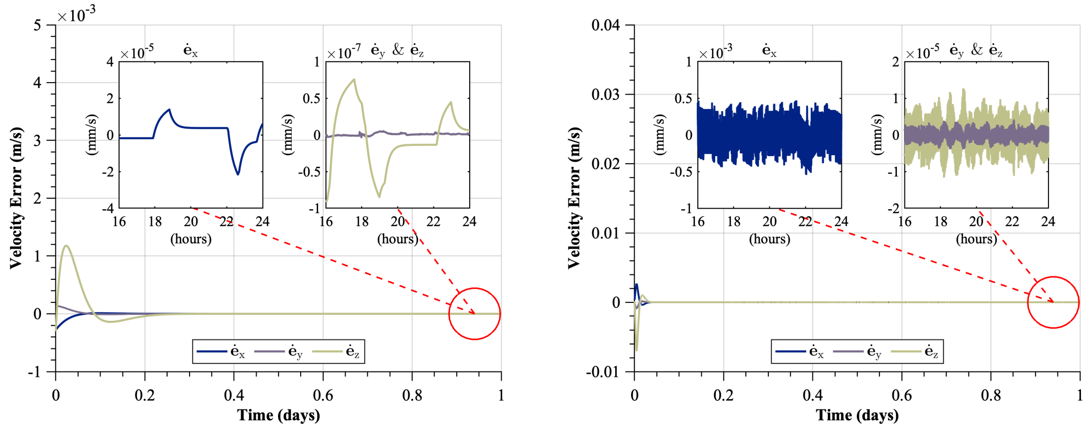
The simulations results when implementing the MSO control scheme are presented for both spacecraft for position and velocity in Figure 15 and Figure 16, respectively. It can be seen that both spacecraft maintain submillimeter relative position and reach the numerical precision limits of the simulation framework just as the PID and SNNARC methods did. The total ΔV consumed for translational control over a ten day period was 4.0922×10^{-2} m/s for the leader spacecraft and 1.4458×10^{-1} m/s for the follower spacecraft, showing the results are comparable to the PID and SNNARC methods. Similar to the SNNARC method, the resulting velocity states, as shown in Figure 16, exhibit reduced jitter as compared to the

PID method in Figure 6. So while the overall relative position results are similar to the PID method, it is anticipated that the results would show improved performance once numerical precision issues could be resolved.



(a) (b)

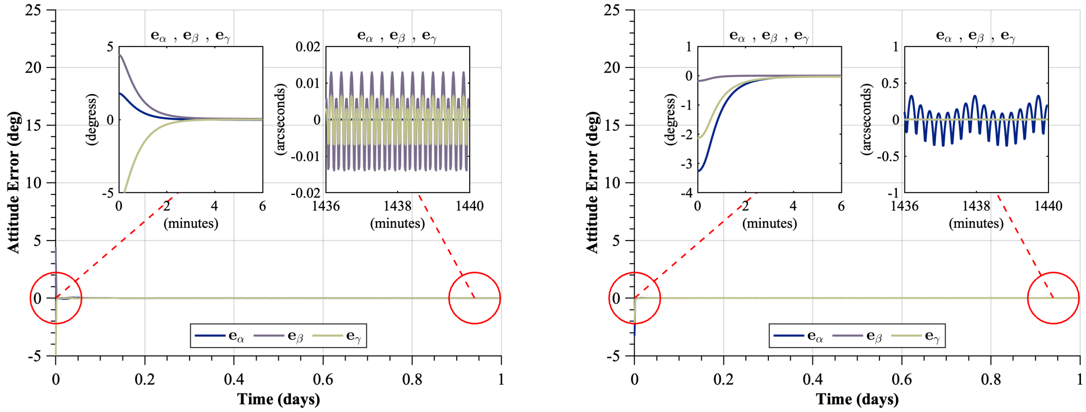
Figure 15. Relative position error performance for the leader spacecraft (a) and follower spacecraft (b) using the MSO control system.



(a) (b)

Figure 16. Relative velocity error performance for the leader spacecraft (a) and follower spacecraft (b) using the MSO control system.

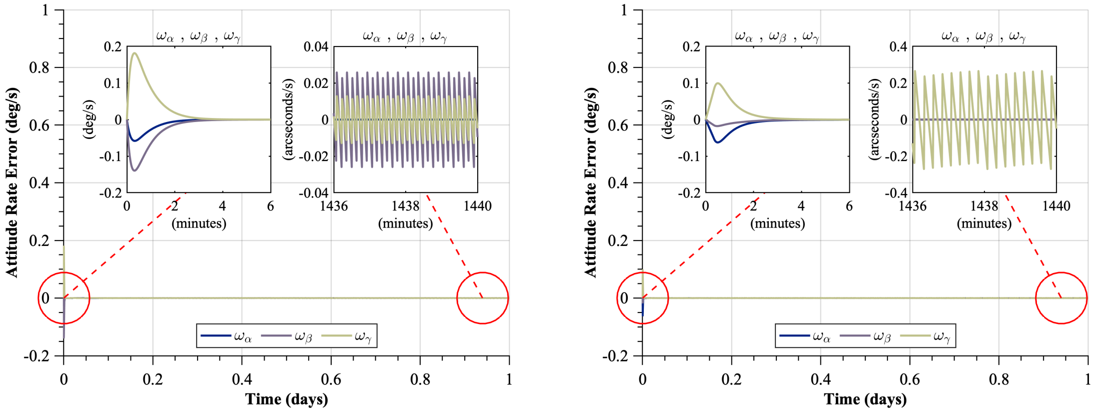
The performance in attitude control for the MSO method is nearly identical to the SNNARC methodology. The performance of both spacecraft for attitude and angular velocities is presented in Figure 17 and Figure 18, respectively. It can be seen that both the leader and follower spacecraft achieve subarcseconds of error, significantly outperforming the PID method. The total ΔV consumed for attitude control over a ten day period was 4.8503×10^{-3} m/s for the leader spacecraft and 3.7137×10^{-2} m/s for the follower spacecraft, a significant reduction in propellant consumption compared to the PID method and similar to the SNNARC method.



(a)

(b)

Figure 17. Attitude error performance for the leader spacecraft (a) and follower spacecraft (b) using the MSO control system.

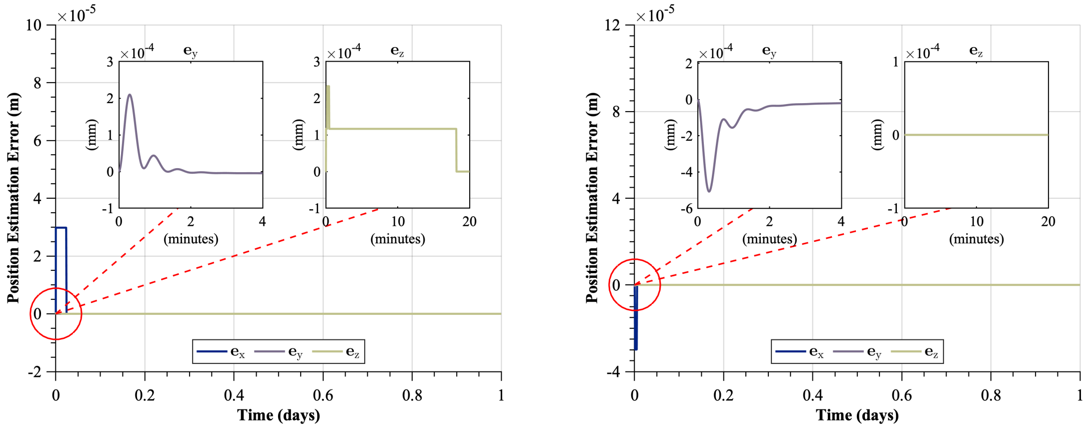


(a)

(b)

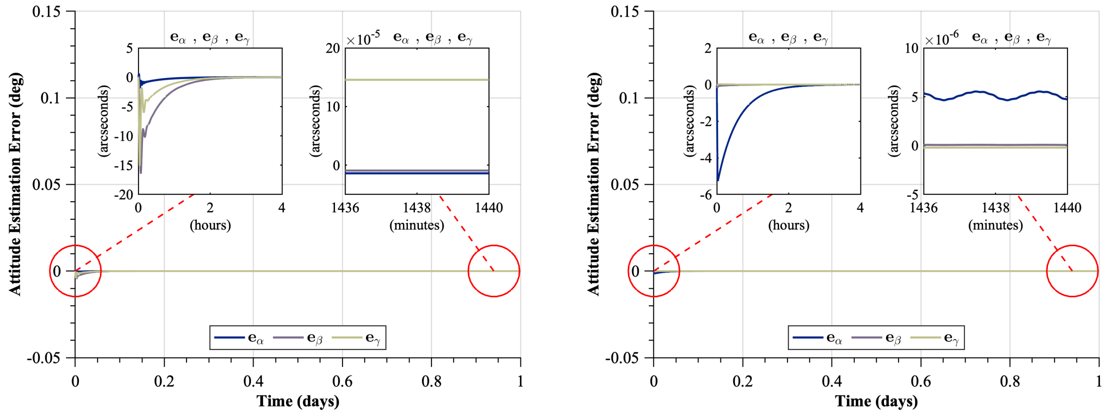
Figure 18. Attitude rate error performance for the leader spacecraft (a) and follower spacecraft (b) using the MSO control system.

The state estimation performance results from the observer portion of the MSO control scheme are presented for both spacecraft for position and attitude in Figure 19 and Figure 20, respectively. It can be seen for the position estimation that the precision limits of the simulation environment are quickly reached and after approximately 20 minutes, steady state shows no estimation error within the allowable numerical precision for both spacecraft. For the attitude state estimation, steady state is reached within approximately four hours with estimation errors in the 2×10^{-6} arcseconds range.



(a) (b)

Figure 19. Position estimation error performance for the leader spacecraft (a) and follower spacecraft (b) using the MSO neural network.

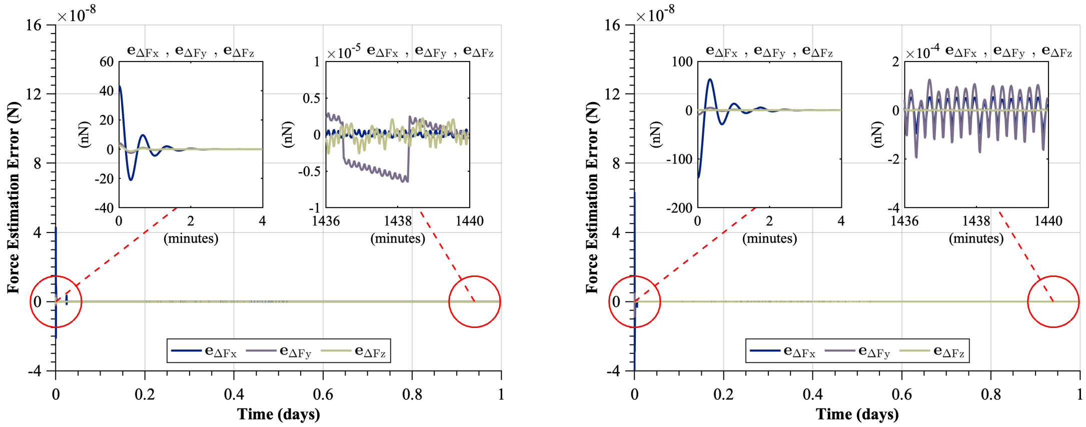


(a) (b)

Figure 20. Attitude estimation error performance for the leader spacecraft (a) and follower spacecraft (b) using the MSO neural network.

To achieve such tight tolerances, the estimated dynamics are propagated using a fourth-order Rung-Kutta (RK4) method. While using an Euler's integration method worked appropriately for attitude dynamics, the chaotic nature of the CR3BP required a higher tolerance integration method to achieve satisfactory estimation performance. As such, an RK4 integration method was used for propagating both the estimated translational and estimated attitude dynamics.

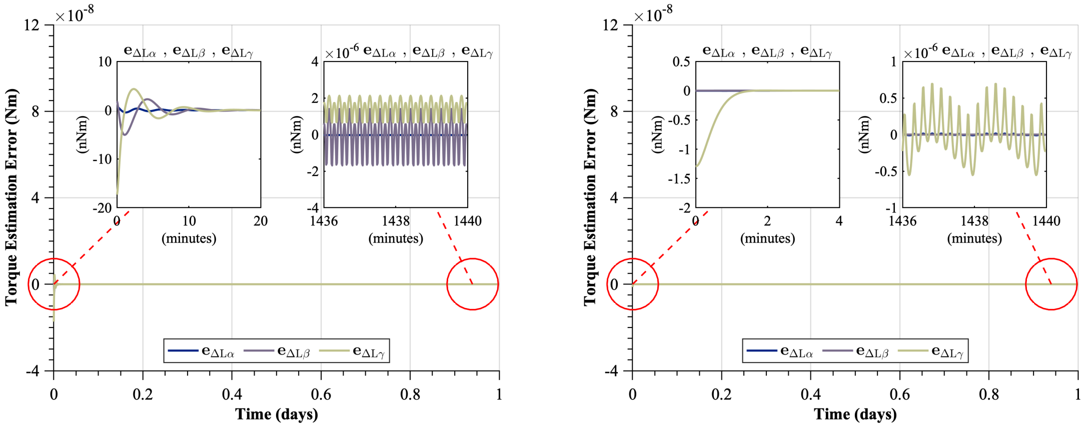
The disturbance estimation performance results from the observer portion of the MSO control scheme are presented for both spacecraft for force and torque perturbations in Figure 21 and Figure 22, respectively. The force and torque estimation converges to steady state within approximately four minutes for both spacecraft, though the estimation of the torque disturbance for the leader spacecraft took approximately 20 minutes. Both spacecraft exhibited similar steady state performance, with the force disturbance estimation being within 1×10^{-5} nanonewtons for the leader spacecraft and 2×10^{-4} nanonewtons for the follower spacecraft, while the torque disturbance estimation was within 2×10^{-6} nanonewtons for both spacecraft. It is interesting to note that while the MSO control scheme was able to estimate perturbations, the method required an integral control portion to compensate for a bias introduced between the state dynamics and desired dynamics. The neural network in the SNNARC method was able to automatically compensate for this bias at the expense of estimating a perturbation that deviated from the truth. Even with this distinction, both methods were nearly identical in position and attitude control performance.



(a)

(b)

Figure 21. Force perturbation estimation error performance for the leader spacecraft (a) and follower spacecraft (b) using the MSO neural network.



(a) (b)

Figure 22. Torque perturbation estimation error performance for the leader spacecraft (a) and follower spacecraft (b) using the MSO neural network.

6. CONCLUSION

Two neural network schemes were introduced, updated, and shown to achieved desired science performance objectives in both position and attitude control for an advanced mission concept modeled after the Stellar Imager mission. The SNNARC methodology was updated by replacing the filtered tracking error with a proportional-derivative approach to the linear tuning gains, allowing for a more traditional approach to controller gain tuning. The MSO methodology was updated from a one-layer neural network to a two-layer neural network, state estimation was expanded to full-state, and integrator action was added to accommodate biases in reference state and reference dynamics formulation. Both neural network methodologies outperformed the traditional PID method, even though system bandwidth was approximately the same. The SNNARC methodology allowed for a simpler approach to achieving the control objective, where the estimator was able to accommodate the perturbations and bias in reference state dynamics at the same time. However, this method can prevent perturbation reconstruction. The MSO methodology allowed for a complementary control design approach, where the perturbation estimation augments a traditional PID approach to improve overall control performance. The method, however, required an RK4 integrator for state estimation, which could increase onboard computation requirements. Not only did the two neural network schemes outperform the PID method in terms of control objectives, both were able to provide significant cost savings in terms of propellant consumption. It is noted that, while not specifically simulated, it is expected that similar results would be observed at the other collinear libration points, for other primary systems, in other multi-body regimes, and in higher fidelity models. Both neural network methods are viable methods for performing precision formation flight and attitude control in a deep space environment, with both presenting potential advantages over each other depending on the mission requirements.

FUNDING SOURCES

Portions of this research were supported through the NASA Pathways Program and NASA Wallops Flight Facility. Additionally, the authors thank the Missouri University of Science and Technology for partially sponsoring this research.

ACKNOWLEDGMENTS

First and foremost, a heartfelt thank you to Dr. Balakrishnan for his major contributions and being a source of inspiration for this work. Special thanks are also extended to Jim Lanzi, Scott Heatwole, and Zach Peterson for their mentorship, support, and contributions to applying this research to NASA spaceflight systems. Thanks are given also to the Guidance, Navigation, and Control, and Mission System Engineering Branch (Code 598) and Balloon Program Office of NASA Wallops Flight Facility and the Attitude Control Systems Engineering Branch (Code 591) of NASA Goddard Space Flight Facility, as well as the NASA Pathways Program.

REFERENCES

[1] Selva, D., and Krejci, D., "A survey and assessment of the capabilities of Cubesats for Earth observation," *Acta Astronautica*, Vol. 74, 2012, pp. 50–68. https://doi.org/https://doi.org/10.1016/j.actaastro.2011.12.014.

[2] Le Moigne, J., "Distributed Spacecraft Missions (DSM) Technology Development at NASA Goddard Space Flight Center," *2018 IEEE International Geoscience and Remote Sensing Symposium (July 2018, Valencia, Spain)*, 2018, pp. 293–296. https://doi.org/10.1109/IGARSS.2018.8519065.

[3] Yost, B., and Weston, S., "State-of-the-Art of Small Spacecraft Technology," Tech. rep., Small Spacecraft Systems Virtual Institute, Ames Research Center, Moffett Field, California, 2017.

[4] Noecker, C., and Kilston, S., "Terrestrial Planet Finder: the search for life elsewhere," *1999 IEEE Aerospace Conference. Proceedings (Cat. No.99TH8403)*, Vol. 4, 1999, pp. 49–57. https://doi.org/10.1109/AERO.1999.792078.

[5] Cockell, C. S., Herbst, T., Léger, A., and et. all, "Darwin—an experimental astronomy mission to search for extrasolar planets," *Experimental Astronomy*, Vol. 23, No. 1, 2009, pp. 435–461. https://doi.org/10.1007/s10686-008-9121-x.

[6] Gendreau, K. C., Cash, W. C., Shipley, A. F., and White, N., "MAXIM Pathfinder x-ray interferometry mission," *X-Ray and Gamma-Ray Telescopes and Instruments for Astronomy, Proceedings of the 2019 International Society for Optics and Photonics Conference (March 2000, Munich, Germany)*, Vol. 4851, 2003, pp. 353–364. https://doi.org/10.1117/12.461316.

[7] Shaddock, D. A., "An Overview of the Laser Interferometer Space Antenna," *Publications of the Astronomical Society of Australia*, Vol. 26, No. 2, 2009, p. 128–132. https://doi.org/10.1071/AS08059.

[8] Carpenter, K. G., Schrijver, C. J., Karovska, M., and SI Mission Concept Development Team, "The Stellar Imager (SI) project: a deep space UV/Optical Interferometer (UVOI) to observe the Universe at 0.1 milli-arcsec angular resolution," *Astrophysics and Space Science*, Vol. 320, 2009, pp. 217–223. https://doi.org/10.1007/s10509-008-9815-6.

[9] Gendreau, K. C., Cash, W. C., Gorenstein, P., Windt, D. L., Kaaret, P., and Reynolds, C., "MAXIM: the black hole imager," *UV and Gamma-Ray Space Telescope Systems, Proceedings of the 2019 International Society for Optics and Photonics Conference (June 2004, Glasgow, United Kingdom)*, Vol. 5488, 2004, pp. 394–402. https://doi.org/10.1117/12.551250.

[10] Carpenter, K. G., Schrijver, C. J., and Karovska, M., "The Stellar Imager (SI) vision mission," *Advances in Stellar Interferometry, Proceedings of the 2019 International Society for Optics and Photonics Conference (May 2006, Orlando, Florida)*, Vol. 6268, 2006, pp. 610–621. https://doi.org/10.1117/12.669713.

[11] Marchand, B. G., and Howell, K. C., "Control Strategies for Formation Flight In the Vicinity of the Libration Points," *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 6, 2005, pp. 1210–1219. https://doi.org/10.2514/1.11016.

[12] Howell, K. C., and Marchand, B. G., "Natural and non-natural spacecraft formations near the L1 and L2 libration points in the Sun–Earth/Moon ephemeris system," *Dynamical Systems*, Vol. 20, No. 1, 2005, pp. 149–173. https://doi.org/10.1080/1468936042000298224.

[13] Xin, M., Balakrishnan, S. N., and Pernicka, H. J., "Multiple spacecraft formation control with Θ-D method," *IET Control Theory & Applications*, Vol. 1, 2007, pp. 485–493. https://doi.org/10.1049/iet-cta_20050410.

[14] Xin, M., Balakrishnan, S. N., and Pernicka, H. J., "Position and Attitude Control of Deep-Space Spacecraft Formation Flying Via Virtual Structure and Θ-D Technique," *Journal of Dynamic Systems, Measurement, and Control*, Vol. 129, No. 5, 2007, pp. 689–698. https://doi.org/10.1115/1.2764509.

[15] Li, P., Cui, P., and Cui, H., "An improved nonlinear control strategy for deep space formation flying spacecraft," *Acta Mechanica Sinica*, Vol. 25, No. 6, 2009, pp. 847–856. https://doi.org/10.1007/s10409-009-0274-2.

[16] Xu, M., Zhou, N., and Wang, J., "Robust adaptive strategy for stationkeeping of Halo orbit," *2012 24th Chinese Control and Decision Conference (May 2012, Taiyuan, China)*, 2012, pp. 3086–3091. https://doi.org/10.1109/CCDC.2012.6244486.

[17] Queiroz, M. S. d., Kapila, V., and Yan, Q., "Adaptive Nonlinear Control of Multiple Spacecraft Formation Flying," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 3, 2000, pp. 385–390. https://doi.org/10.2514/2.4549.

[18] Gurfil, P., Idan, M., and Kasdin, N. J., "Neurocontrol of Spacecraft Formation Flying in the Elliptic Restricted Three-Body Problem," *AIAA Guidance, Navigation, and Control Conference and Exhibit (August 2002, Monterey, California)*, 2002, pp. 1–11. https://doi.org/10.2514/6.2002-4962.

[19] Wang, W., Mengali, G., Quarta, A. A., and Yuan, J., "Distributed adaptive synchronization for multiple spacecraft formation flying around Lagrange point orbits," *Aerospace Science and Technology*, Vol. 74, 2018, pp. 93–103. https://doi.org/10.1016/j.ast.2018.01.007.

[20] Xu, M., Liang, Y., and Fu, X., "Formation flying on quasi-halo orbits in restricted Sun–Earth/Moon system," *Aerospace Science and Technology*, Vol. 67, 2017, pp. 118–125. https://doi.org/https://doi.org/10.1016/j.ast.2017.03.038.

[21] Shahid, K., and Kumar, K. D., "Formation Control at the Sun-Earth L2 Libration Point Using Solar Radiation Pressure," *Journal of Spacecraft and Rockets*, Vol. 47, No. 4, 2010, pp. 614–626. https://doi.org/10.2514/1.47342.

[22] Wang, W., Baoyin, H., Mengali, G., and Quarta, A. A., "Solar sail cooperative formation flying around L2-type artificial equilibrium points," *Acta Astronautica*, Vol. 169, 2020, pp. 224–239. https://doi.org/10.1016/j.actaastro.2019.10.028.

[23] Qi, R., Xu, S., and Xu, M., "Impulsive Control for Formation Flight About Libration Points," *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 2, 2012, pp. 484–496. https://doi.org/10.2514/1.54383.

[24] Ghafoor, A., Galchenko, P., Balakrishnan, S. N., Pernicka, H., and Yucelen, T., "ETNAC Design Enabling Formation Flight at Liberation Points," *Proceedings of the 2019 American Control Conference (July 2019, Philadelphia, PA)*, 2019, pp. 3689–3694. https://doi.org/10.23919/ACC.2019.8814922.

[25] Kane, T. R., and Marsh, E. L., "Attitude stability of a symmetric satellite at the equilibrium points in the restricted three-body problem," *Celestial mechanics*, Vol. 4, No. 1, 1971, pp. 78–90. https://doi.org/10.1007/BF01230323.

[26] Guzzetti, D., and Howell, K. C., "Natural periodic orbit-attitude behaviors for rigid bodies in three-body periodic orbits," *Acta Astronautica*, Vol. 130, 2017, pp. 97–113. https://doi.org/10.1016/j.actaastro.2016.06.025.

[27] Guzzetti, D., and Howell, K. C., "Attitude dynamics in the circular restricted three-body problem," *Astrodynamics*, Vol. 2, No. 2, 2018, pp. 87–119. https://doi.org/10.1007/s42064-017-0012-7.

[28] Vallado, D. A., *Fundamentals of Astrodynamics and Applications*, Microcosm Press, Fourth Edition, 2013.

[29] Markley, F. L., and Crassidis, J. L., *Fundamentals of Spacecraft Attitude Determination and Control*, Springer, First Edition, 2014.

[30] Kahlil, H. K., *Nonlinear Control*, Pearson, First Edition, 2015.

[31] Galchenko, P., and Pernicka, H. J., "Precision Formation Flying and Spacecraft Pointing Using Plasmonic Force Propulsion," *Proceedings of the 2017 AAS/AIAA Astrodynamics Specialist Conference (August 2017, Stevenson, WA)*, Vol. 162, 2017, pp. 2985–2997.

[32] Galchenko, P., and Pernicka, H. J., "Precision Control of Microsatellite Swarms Using Plasmonic Force Propulsion," *Proceedings of the 2018 AAS/AIAA Astrodynamics Specialist Conference (August 2018, Snowbird, UT)*, Vol. 167, 2018, pp. 935–954.

[33] Galchenko, P., Pernicka, H. J., and Balakrishnan, S. N., "Pointing System Design for the COronal Diagnostic Experiment (CODEX) using a Modified State Observer and a Neural Network Controller," *Proceedings of the 2020 AAS/AIAA Astrodynamics Specialist Conference (August 2020, South Lake Tahoe, CA)*, 2021, pp. 645–660.

[34] Lewis, F. L., Jagannathan, S., and Yesildirek, A., *Neural Network Control of Robot Manipulators and Nonlinear Systems*, Taylor and Francis, 1999.

[35] Rajagopal, K., Mannava, A., Balakrishnan, S., Nguyen, N., and Krishnakumar, K., "Neuroadaptive Model Following Controller Design for Non-Affine and Non-Square Aircraft Systems," *Proceedings of the 2009 AIAA Guidance, Navigation, and Control Conference (August 2009, Chicago, IL)*, 2009, pp. 1–21. https://doi.org/10.2514/6.2009-5737.

[36] Mannava, A., Balakrishnan, S., Tang, L., and Landers, R., "Optimal Tracking Control of Motion Systems," *IEEE Transactions on Control Systems Technology*, Vol. 20, 2012, pp. 1548–1558. https://doi.org/10.1109/TCST.2011.2168608.

[37] Harl, N., Rajagopal, K., and Balakrishnan, S. N., "Neural Network Based Modified State Observer for Orbit Uncertainty Estimation," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 4, 2013, pp. 1194–1209. https://doi.org/10.2514/1.55711.

[38] Carpenter, K. G., Schrijver, C. J., and Karovska, M., "SI – The Stellar Imager," Tech. rep., Vision Mission Study Report, NASA Goddard Space Flight Facility, 2017.

[39] Wertz, J. R., Everett, D. F., and Puschell, J. J., *Space Mission Engineering: The New SMAD*, Microcosm Press, First Edition, 2011.

[40] Rovey, J. L., Yang, X., Friz, P. D., Hu, C., and Glascock, M. S., "Plasmonic force propulsion revolutionizes nano/picosatellite capability," Tech. rep., NASA Innovative Advanced Concepts Phase I Final Report, 2014.

[41] Rovey, J. L., Friz, P. D., Hu, C., Glascock, M. S., and Yang, X., "Plasmonic Force Space Propulsion," *Journal of Spacecraft and Rockets*, Vol. 52, No. 4, 2015, pp. 1163–1168. https://doi.org/10.2514/1.A33155.

SECTION

2. FUTURE WORK

With the successful testing of the two neural networks on the Wallops Arc-Second Pointer, as well as demonstrating the functionality on advanced mission concepts such as formation control of large virtual structures in a deep space environment, a significant amount of development is planned for future work. As the work is ongoing, presented below is a subset of the recent accomplishments and future applications.

2.1. XL-CALIBUR

The next high-altitude balloon-borne mission that will be flown by NASA Wallops Flight Facility (WFF) is XL-Calibur, which is an X-ray polarimetry mission that will study high-energy emissions from deep space sources such as black-hole systems and neutron stars. The mission finished the initial integration phase in December 2021 at NASA WFF in preparation for a Spring/Summer 2022 flight from Sweden. This science instrument is significantly larger than any previous WASP flight, with its inertia approximately an order of magnitude larger as compared to previous missions. The XL-Calibur and WASP platform are shown in Figure 2.1 during the initial integration phase at NASA WFF.

During the integration period, the two neural network methods were tested and refined on the flight hardware. For both the SNNARC and MSO methods, a desired velocity derivation was added to enable tracking a geodetic target. Both methods were able to track the geodetic target with sub-arcsecond precision without any observable steady-state bias, and both were able to outperform the heritage control system. The MSO method was able to achieve the best performance with an impressive RMS error of only 0.048 arcseconds in pitch and 0.055 arcseconds in yaw during one of the testing dates. The MSO algorithm was also revised to add the addition of a velocity limit to the control system. By refining how the proportional and derivative gains are applied, a limit on the angular velocity could be achieved. This enables using high gains for learning while maintaining a slew maneuver without compromising learning or pointing performance. In particular, with both the velocity limit enabled and the desired velocity tracking enabled for geodetic targets, the MSO method was able to slew with near zero bias, whereas the heritage control system exhibited a large lagging bias during slews. Not only does this demonstrate the improved performance of the neural network methods over the heritage approach, it also shows that these control methodologies are viable for a variety of configurations for the WASP platform.

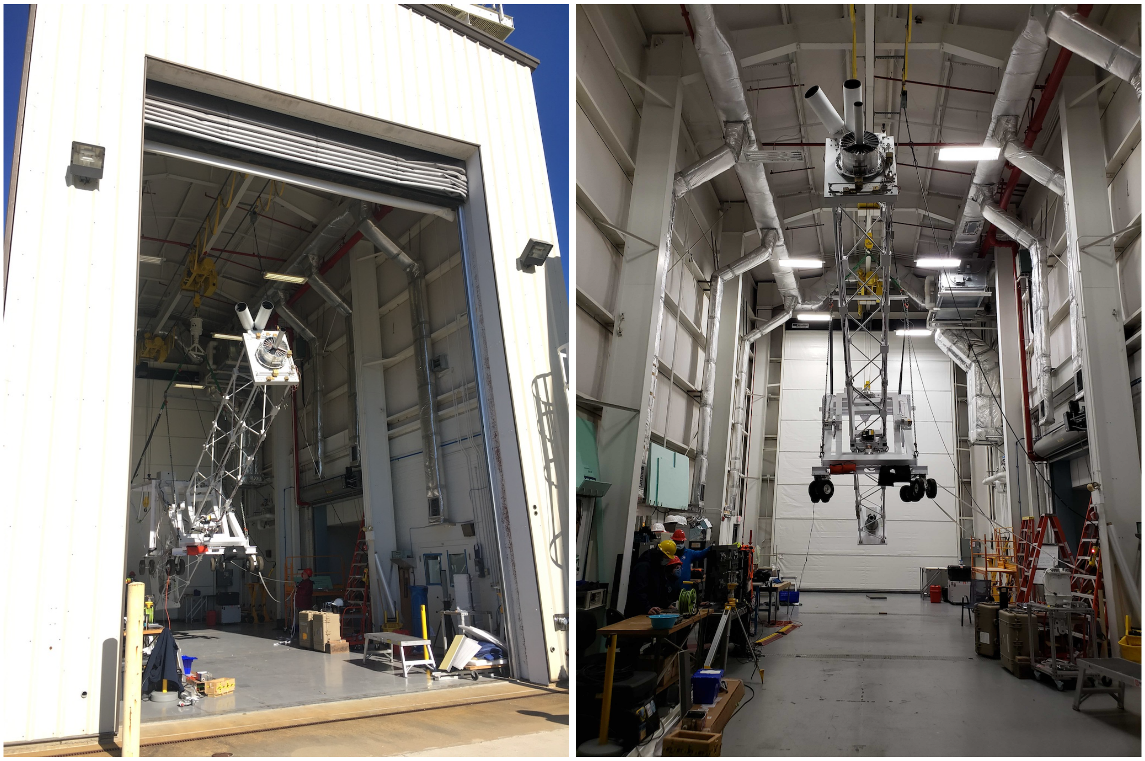


Figure 2.1. The WASP platform with the XL-Calibur mission science hardware mounted as the centerbody is shown in suspended system flight functional configuration at NASA WFF.

2.2. PICTURE-C

Due to the delay in the science flight of the PICTURE-C mission, a new flight opportunity is planed for Fall 2022. Having demonstrated the benefits of velocity limiting on the MSO methodology, it is desirable to also implement this for the SNNARC methodology. The removal of the filtered tracking error when demonstrating the SNNARC methodology for the two spacecraft in a a deep space environment has enabled applying a similar approach. The linear gain tuning of SNNARC is now similar to the MSO method, and as such a similar approach to velocity limiting can now be implemented. Testing with XL-Calibur has shown that currently, performing a slew using the SNNARC method causes instability. It is predicted that velocity limiting will help prevent instability in the learning, which is driven by error. The PICTURE-C mission plans to observe multiple inertial stellar targets and would benefit from improved pointing stability offered by the two neural network methods. Currently, the operational planning requires switching the control law from SNNARC, if that is being used, prior to performing a slew and then switching back to SNNARC once on target again. This introduces the potential of human-factor errors, so enabling SNNARC to be able to perform slews is greatly desirable. This work is planned to be completed for the Picture-C mission during Summer 2022.

2.3. CODEX

The COronal Diagnostic Experiment (CODEX), which is scheduled to launch to the International Space Station (ISS) in 2023, seeks to improve the performance of its pointing system utilizing the two neural network methods presented in this research [32]. It will be mounted to the ExPRESS Logistics Carrier (ELC 3), as shown in Figure 2.2, which was added to the ISS in May 2011. The CODEX mission seeks to study coronal mass ejections

(CME), which are abrupt and explosive ejections of coronal plasma from the surface of the Sun. These CMEs can interact with Earth's magnetic field causing geomagnetic storms, potentially harming spacecraft and terrestrial infrastructure.

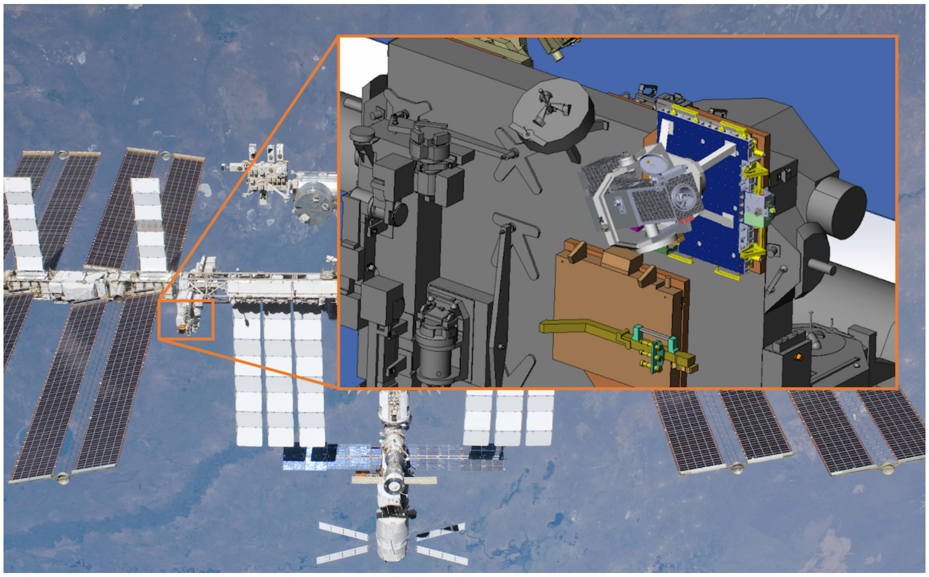
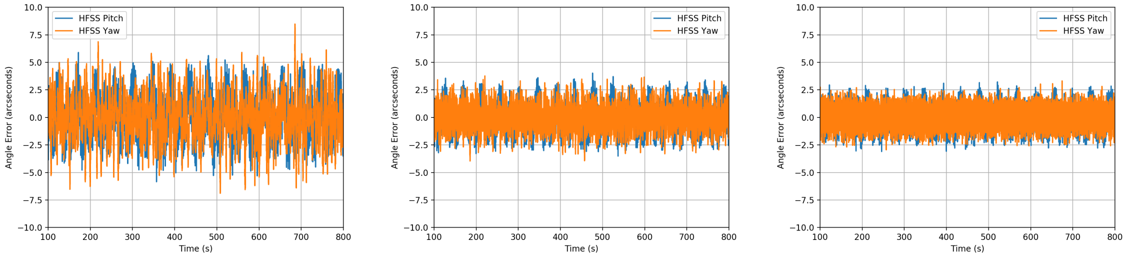


Figure 2.2. CAD Representation of CODEX mounted to the ISS on the ExPRESS Logistics Carrier.

While the NASA team has extensive experience in developing and operating pointing systems for a high-altitude balloon platform, the transition to spaceflight offers new challenges that must be considered. For example, one source of disturbance occurs from the ISS's own control system used to hold a local vertical, local horizon (LVLH) orientation. This motion of the ISS, in addition to general vibrations of the ISS mechanical structure, translate as motion disturbances to the CODEX platform. To meet budget and schedule constraints, several changes to the platform were also introduced, including the removal of the inertial measurement unit (IMU) and rotating shafts in the azimuth and elevation motor hubs. The rotating shafts allowed the motor hubs to operate without considerations of static friction, which must now be accounted for in the new design. Similarly, the removal of the IMU requires filtering to derive the body rates of the telescope.

A traditional PID controller has been baselined for the mission, which has worked well for the team in previous pointing systems for the WASP platform. The team is considering the addition of either the SNNARC and/or MSO methodology to supplement the control system as both a technology demonstration and potential substitute if scientific objectives can be improved. The flight software for the control system has already been written and testing with software-in-the-loop (SWIL) simulations has been successfully conducted, showing significant improvements in pointing performance as compared to the heritage controller, as shown in Figure 2.3. Furthermore, the software has been compiled to the flight hardware and hardware-in-the-loop (HWIL) testing has begun. As this spaceflight system is higher risk, it is desired to further test the two neural network methods in a flight-like environment, and as such the WASP platform serves as the perfect candidate. The hardware testing and in flight performance characterizations will offer valuable lessons-learned to help further improve and refine the two neural network methodologies.



(a) Heritage PID Controller (b) SNNARC Performance (c) MSO Performance

Figure 2.3. Both the SNNARC (b) and MSO (c) methods provide significant improvements in steady state performance as compared to the heritage PID (a) control system in CODEX SWIL testing.

2.4. DISTRIBUTED SYSTEM MISSIONS

Enabling Distributed System Mission (DSM) architectures for advanced mission concepts has become a priority for NASA Goddard Space Flight Center (GSFC). In fact, investigating and enabling formation control for SmallSat constellations for the DSM architecture has led to funding through the Internal Research & Development program at GSFC.

While the work is still ongoing, development has already begun in enabling a variety of mission and control architectures in a high fidelity environment. As part of that work, a version of the MSO neural network was tested in the General Mission Analysis Tool (GMAT), where the guidance and control system algorithm used the CR3BP model presented in Paper III. However, due to the GMAT high-fidelity simulation environment, the “truth” environment is n-body, and as such, the desired halo orbit no longer naturally occurs. Initial testing of the neural network method has shown that even in an n-body environment, where even the libration point itself is time variant, the neural network successfully learned the dynamics and the control system was able to maintain precision formation flight. The development work has transitioned to using NASA 42, as this high-fidelity simulation environment offers the ability of simulating attitude dynamics in addition to translational dynamics, a capability GMAT does not currently have. As part of the work planned, it is desired to include a variety of control strategies to the development, including the two neural network methods presented here. Furthermore, the NASA 42 software allows for the modeling of sensors, navigation algorithms, and actuator hardware such as thrusters and reaction wheels. All these capabilities can further enable high-fidelity simulations for spacecraft utilizing the DSM architecture, where the two neural networks presented can be greatly beneficial.

3. SUMMARY AND CONCLUSIONS

This research has shown that advanced mission concepts can be achieved for spaceflight systems using neural networks. Two neural network methodologies were synthesized, implemented, tested, and refined in both simulation and hardware testing, showing not only the viability of the method, but also the benefits as compared to other heritage control methods. The SNNARC methodology was introduced with both a filtered tracking error and a traditional proportional-derivative formulation, while using a two-layer neural network to estimate nonlinear dynamics within the control space, to control both translational and rotational dynamics of a spaceflight system. The MSO methodology introduced both a one-layer and two-layer neural network that operated within an observer to estimate the nonlinear dynamics outside the control space, and then used input-output feedback linearization to control both translational and rotational dynamics of the spaceflight system. Lyapunov-based proofs were presented that showed both control systems were ultimately upper bounded. Simulations showed that both neural network methodologies offered improved attitude control of the Wallops Arc-Second Pointer as well as offering improved position and attitude control of two spacecraft in formation control in a deep space environment. Finally, hardware testing of the two neural network methods on the WASP platform, a high-altitude balloon-borne spaceflight system, showed that the methods outperformed the heritage control system. Through the use of neural network methods, control performance can be improved for spaceflight systems when their dynamics are unknown or poorly understood and/or modelled.

REFERENCES

[1] Gril, G., “GLCM Guidance System,” *Association of Air Force Missileers*, Vol. 5, No. 4, 1997, p. 5.

[2] Lippmann, R., “An introduction to computing with neural nets,” *IEEE ASSP Magazine*, Vol. 4, No. 2, 1987, pp. 4–22. https://doi.org/10.1109/MASSP.1987.1165576.

[3] Lewis, F. L., Jagannathan, S., and Yesildirek, A., *Neural Network Control of Robot Manipulators and Nonlinear Systems*, Taylor and Francis, 1999.

[4] Simpson, P., *Artificial Neural Systems: Foundations, Paradigms, Applications, and Implementations*, Neural networks, Elsevier Science Limited, 1990.

[5] Hush, D., and Horne, B., “Progress in supervised neural networks,” *IEEE Signal Processing Magazine*, Vol. 10, No. 1, 1993, pp. 8–39. https://doi.org/10.1109/79.180705.

[6] Garde, G. J., and Fairbrother, D. A., “The NASA Balloon Program - Positioning For the Future,” *Proceedings of the 2015 AIAA Balloon Systems Conference (June 2015, Dallas, TX)*, 2015, pp. 1–8. https://doi.org/10.2514/6.2015-2907.

[7] Fairbrother, D. A., “2017 NASA Balloon Program Update,” *Proceedings of the 2017 AIAA Balloon Systems Conference (June 2017, Denver, CO)*, 2017, pp. 1–8. https://doi.org/10.2514/6.2017-3090.

[8] Jones-Wilson, L., Susca, S., Diaz, C., Chang, H., Duffy, E., Effinger, R., Lewis, D., Liewer, K., Lo, K., Ochoa, H., Perez, J., Rizvi, A., Seubert, C., Umsted, C., Borden, M., Clark, P., Massey, R., and Porter, M., “A Sub-Arcsecond Pointing Stability Fine Stage for a High Altitude Balloon Platform,” *Proceedings of the 2017 IEEE Aerospace Conference (March 2017, Big Sky, MT)*, 2017, pp. 1–15. https://doi.org/10.1109/AERO.2017.7943590.

[9] Stuchlik, D. W., “The Wallops Arc Second Pointer - A Balloon Borne Fine Pointing System,” *Proceedings of the 2015 AIAA Balloon Systems Conference (June 2015, Dallas, TX)*, 2015, pp. 1–15. https://doi.org/10.2514/6.2015-3039.

[10] Stuchlik, D. W., and Lanzi, R., “The NASA Wallops Arc-Second Pointer (WASP) System for Precision Pointing of Scientific Balloon Instruments and Telescopes,” *Proceedings of the 2017 AIAA Balloon Systems Conference (June 2017, Denver, CO)*, 2017, pp. 1–8. https://doi.org/10.2514/6.2017-3609.

[11] Kopp, G., Smith, P., Belting, C., Castleman, Z., Drake, G., Espejo, J., Heuerman, K., Lanzi, J., and Stuchlik, D., “Radiometric flight results from the HyperSpectral Imager for Climate Science (HySICS),” *Geoscientific Instrumentation, Methods and Data Systems*, Vol. 6, 2017, pp. 169–191. https://doi.org/10.5194/gi-6-169-2017.

[12] Hurford, T. A., Mandell, A., Reddy, V., and Young, E., "Observatory for Planetary Investigations from the Stratosphere," *Proceedings of the 46th Lunar and Planetary Science Conference (March 2015, The Woodlands, TX)*, 2015, pp. 1–2.

[13] Mendillo, C. B., Hewawasam, K., Howe, G. A., Martel, J., Cook, T. A., and Chakrabarti, S., "The PICTURE-C exoplanetary direct imaging balloon mission: first flight preparation," *Techniques and Instrumentation for Detection of Exoplanets IX, Proceedings of the 2019 International Society for Optics and Photonics Conference (August 2019, San Diego, CA)*, Vol. 11117, SPIE, 2019, pp. 101 – 111. https://doi.org/10.1117/12.2529710.

[14] Mendillo, C. B., Hewawasam, K., Martel, J., Potter, T., Cook, T. A., and Chakrabarti, S., "The PICTURE-C Exoplanetary Imaging Balloon Mission: First Flight Preparation," *Techniques and Instrumentation for Detection of Exoplanets IX, Proceedings of the 2021 International Society for Optics and Photonics Conference (August 2021, San Diego, CA)*, Vol. 11823, SPIE, 2021, pp. 125 – 133. https://doi.org/10.1117/12.2594749.

[15] Gopalswamy, N., Newmark, J., Yashiro, S., Mäkelä, P., Reginald, N., Thakur, N., Gong, Q., Kim, Y. H., Cho, K. S., Choi, S. H., Baek, J. H., Bong, S. C., Yang, H. S., Park, J. Y., Kim, J. H., Park, Y. D., Lee, J. O., Kim, R. S., and Lim, E. K., "The Balloon-Borne Investigation of Temperature and Speed of Electrons in the Corona (BITSE): Mission Description and Preliminary Results," *Solar Physics*, Vol. 296, 2021. https://doi.org/10.1007/s11207-020-01751-8.

[16] Abarr, Q., Awaki, H., Baring, M., Bose, R., De Geronimo, G., Dowkontt, P., Errando, M., Guarino, V., Hattori, K., Hayashida, K., Imazato, F., Ishida, M., Iyer, N., Kislat, F., Kiss, M., Kitaguchi, T., Krawczynski, H., Lisalda, L., Matake, H., Maeda, Y., Matsumoto, H., Mineta, T., Miyazawa, T., Mizuno, T., Okajima, T., Pearce, M., Rauch, B., Ryde, F., Shreves, C., Spooner, S., Stana, T.-A., Takahashi, H., Takeo, M., Tamagawa, T., Tamura, K., Tsunemi, H., Uchida, N., Uchida, Y., West, A., Wulf, E., and Yamamoto, R., "XL-Calibur – a Second-Generation Balloon-Borne Hard X-Ray Polarimetry Mission," *Astroparticle Physics*, Vol. 126, 2021, p. 102529. https://doi.org/10.1016/j.astropartphys.2020.102529.

[17] McCarthy, D. J., "Operating Characteristics of the Stratoscope II Balloon-Borne Telescope," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 5, No. 2, 1969, pp. 323–329. https://doi.org/10.1109/TAES.1969.309922.

[18] Pascale, E., Ade, P. A. R., Bock, J. J., Chapin, E. L., Chung, J., Devlin, M. J., Dicker, S., Griffin, M., Gundersen, J. O., Halpern, M., Hargrave, P. C., Hughes, D. H., Klein, J., MacTavish, C. J., Marsden, G., Martin, P. G., Martin, T. G., Mauskopf, P., Netterfield, C. B., Olmi, L., Patanchon, G., Rex, M., Scott, D., Semisch, C., Thomas, N., Truch, M. D. P., Tucker, C., Tucker, G. S., Viero, M. P., and Wiebe, D. V., "The Balloon-Borne Large Aperture Submillimeter Telescope: BLAST," *The Astrophysical Journal*, Vol. 681, No. 1, 2008, pp. 400–414. https://doi.org/10.1086/588541.

[19] Barthol, P., Gandorfer, A., Solanki, S. K., Schüssler, M., Chares, B., Curdt, W., Deutsch, W., Feller, A., Germerott, D., Grauf, B., Heerlein, K., Hirzberger, J., Kolleck, M., Meller, R., Müller, R., Riethmüller, T. L., Tomasch, G., Knölker, M., Lites, B. W., Card, G., Elmore, D., Fox, J., Lecinski, A., Nelson, P., Summers, R., Watt, A., Martínez Pillet, V., Bonet, J. A., Schmidt, W., Berkefeld, T., Title, A. M., Domingo, V., Gasent Blesa, J. L., del Toro Iniesta, J. C., López Jiménez, A., Álvarez-Herrero, A., Sabau-Graziati, L., Widani, C., Haberler, P., Härtel, K., Kampf, D., Levin, T., Pérez Grande, I., Sanz-Andrés, A., and Schmidt, E., "The Sunrise Mission," *Solar Physics*, Vol. 268, 2011, pp. 1–34. https://doi.org/10.1007/s11207-010-9662-9.

[20] Romualdez, L. J., Damaren, C. J., Li, L., Galloway, M. N., Hartley, J. W., Netterfield, C. B., Clark, P., and Massey, R. J., "Precise Pointing and Stabilization Performance for the Balloon-Borne Imaging Testbed: 2015 Test Flight," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, Vol. 231, No. 4, 2017, pp. 713–727. https://doi.org/10.1177/0954410016641451.

[21] Borden, M., Lewis, D., Ochoa, H., Jones-Wilson, L., Susca, S., Porter, M., Massey, R., Clark, P., and Netterfield, B., "Thermal, Structural, and Optical Analysis of a Balloon-Based Imaging System," *Publications of the Astronomical Society of the Pacific*, Vol. 129, 2017. https://doi.org/10.1088/1538-3873/129/973/035001.

[22] Markley, F. L., and Crassidis, J. L., *Fundamentals of Spacecraft Attitude Determination and Control*, Springer, First Edition, 2014.

[23] Slotine, J. E., and Li, W., *Applied Nonlinear Control*, Prentice Hall, 1995.

[24] Abdo, M., Vali, A. R., Toloei, A., and Arvan, M. R., "Research on the Cross-Coupling of a Two Axes Gimbal System with Dynamic Unbalance," *International Journal of Advanced Robotic Systems*, Vol. 10, 2013, p. 357. https://doi.org/10.5772/56963.

[25] Ekstrand, B., "Equations of Motion for a Two-Axes Gimbal System," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 37, No. 3, 2001, pp. 1083–1091. https://doi.org/10.1109/7.953259.

[26] Liu, S., Lu, T., Shang, T., and Xia, Q., "Dynamic Modeling and Coupling Characteristic Analysis of Two-Axis Rate Gyro Seeker," *International Journal of Aerospace Engineering*, Vol. 2018, 2018. https://doi.org/10.1155/2018/8513684.

[27] Kahlil, H. K., *Nonlinear Control*, Pearson, First Edition, 2015.

[28] Darling, J. E., Legrand, K. A., Galchenko, P., Pernicka, H. J., DeMars, K. J., Shirley, A. T., McCabe, J. S., Schmid, C. L., Haberberger, S. J., and Mundahl, A. J., "Development and Flight of a Stereoscopic Imager for Use in Spacecraft Close Proximity Operations," *Proceedings of the 39th Annual AAS Rocky Mountain Section Guidance and Control Conference (February 2016, Breckenridge, CO)*, Vol. 157, 2016, pp. 489–500.

[29] Davis, J., Galchenko, P., Jennings, D., and Pernicka, H. J., "Development and Validation of a GNC Algorithm Using a Stereoscopic Imaging Sensors in Close Proximity Operations," *Proceedings of the 2017 AAS/AIAA Astrodynamics Specialist Conference (August 2017, Stevenson, WA)*, Vol. 162, 2018, pp. 3149–3161.

[30] Galchenko, P., and Pernicka, H. J., "Precision Control of Microsatellite Swarms Using Plasmonic Force Propulsion," *Proceedings of the 2018 AAS/AIAA Astrodynamics Specialist Conference (August 2018, Snowbird, UT)*, Vol. 167, 2018, pp. 935–954.

[31] Jennings, D., Davis, J., Galchenko, P., and Pernicka, H. J., "Validation of a GNC Algorithm Using a Stereoscopic Imaging Sensor to Conduct Close Proximity Operations," *Proceedings of the 41st Annual AAS Rocky Mountain Section Guidance and Control Conference (2018, Breckenridge, CO)*, Vol. 164, 2018, pp. 47–58.

[32] Galchenko, P., and Pernicka, H. J., "Precision Control of Microsatellite Swarms Using Plasmonic Force Propulsion," *Proceedings of the 2018 AAS/AIAA Astrodynamics Specialist Conference (August 2018, Snowbird, UT)*, Vol. 167, 2019, pp. 935–954.

[33] MacKunis, W., Leve, F., Patre, P., Fitz-Coy, N., and Dixon, W., "Adaptive Neural Network-Based Satellite Attitude Control in the Presence of CMG Uncertainty," *Aerospace Science and Technology*, Vol. 54, 2016, pp. 218–228. https://doi.org/10.1016/j.ast.2016.04.022.

[34] Ghafoor, A., Galchenko, P., Balakrishnan, S. N., Pernicka, H., and Yucelen, T., "ETNAC Design Enabling Formation Flight at Liberation Points," *Proceedings of the 2019 American Control Conference (July 2019, Philadelphia, PA)*, 2019, pp. 3689–3694. https://doi.org/10.23919/ACC.2019.8814922.

[35] Galchenko, P., Pernicka, H. J., and Balakrishnan, S. N., "Pointing System Design for the COronal Diagnostic Experiment (CODEX) using a Modified State Observer and a Neural Network Controller," *Proceedings of the 2020 AAS/AIAA Astrodynamics Specialist Conference (August 2020, South Lake Tahoe, CA)*, 2021, pp. 645–660.

[36] Rajagopal, K., Mannava, A., Balakrishnan, S., Nguyen, N., and Krishnakumar, K., "Neuroadaptive Model Following Controller Design for Non-Affine and Non-Square Aircraft Systems," *Proceedings of the 2009 AIAA Guidance, Navigation, and Control Conference (August 2009, Chicago, IL)*, 2009, pp. 1–21. https://doi.org/10.2514/6.2009-5737.

[37] Mannava, A., Balakrishnan, S., Tang, L., and Landers, R., "Optimal Tracking Control of Motion Systems," *IEEE Transactions on Control Systems Technology*, Vol. 20, 2012, pp. 1548–1558. https://doi.org/10.1109/TCST.2011.2168608.

[38] Harl, N., Rajagopal, K., and Balakrishnan, S. N., "Neural Network Based Modified State Observer for Orbit Uncertainty Estimation," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 4, 2013, pp. 1194–1209. https://doi.org/10.2514/1.55711.

[39] Selva, D., and Krejci, D., “A survey and assessment of the capabilities of Cubesats for Earth observation,” *Acta Astronautica*, Vol. 74, 2012, pp. 50–68. https://doi.org/https://doi.org/10.1016/j.actaastro.2011.12.014.

[40] Le Moigne, J., “Distributed Spacecraft Missions (DSM) Technology Development at NASA Goddard Space Flight Center,” *2018 IEEE International Geoscience and Remote Sensing Symposium (July 2018, Valencia, Spain)*, 2018, pp. 293–296. https://doi.org/10.1109/IGARSS.2018.8519065.

[41] Yost, B., and Weston, S., “State-of-the-Art of Small Spacecraft Technology,” Tech. rep., Small Spacecraft Systems Virtual Institute, Ames Research Center, Moffett Field, California, 2017.

[42] Noecker, C., and Kilston, S., “Terrestrial Planet Finder: the search for life elsewhere,” *1999 IEEE Aerospace Conference. Proceedings (Cat. No.99TH8403)*, Vol. 4, 1999, pp. 49–57. https://doi.org/10.1109/AERO.1999.792078.

[43] Cockell, C. S., Herbst, T., Léger, A., and et. all, “Darwin—an experimental astronomy mission to search for extrasolar planets,” *Experimental Astronomy*, Vol. 23, No. 1, 2009, pp. 435–461. https://doi.org/10.1007/s10686-008-9121-x.

[44] Gendreau, K. C., Cash, W. C., Shipley, A. F., and White, N., “MAXIM Pathfinder x-ray interferometry mission,” *X-Ray and Gamma-Ray Telescopes and Instruments for Astronomy, Proceedings of the 2019 International Society for Optics and Photonics Conference (March 2000, Munich, Germany)*, Vol. 4851, 2003, pp. 353–364. https://doi.org/10.1117/12.461316.

[45] Shaddock, D. A., “An Overview of the Laser Interferometer Space Antenna,” *Publications of the Astronomical Society of Australia*, Vol. 26, No. 2, 2009, p. 128–132. https://doi.org/10.1071/AS08059.

[46] Carpenter, K. G., Schrijver, C. J., Karovska, M., and SI Mission Concept Development Team, “The Stellar Imager (SI) project: a deep space UV/Optical Interferometer (UVOI) to observe the Universe at 0.1 milli-arcsec angular resolution,” *Astrophysics and Space Science*, Vol. 320, 2009, pp. 217–223. https://doi.org/10.1007/s10509-008-9815-6.

[47] Gendreau, K. C., Cash, W. C., Gorenstein, P., Windt, D. L., Kaaret, P., and Reynolds, C., “MAXIM: the black hole imager,” *UV and Gamma-Ray Space Telescope Systems, Proceedings of the 2019 International Society for Optics and Photonics Conference (June 2004, Glasgow, United Kingdom)*, Vol. 5488, 2004, pp. 394–402. https://doi.org/10.1117/12.551250.

[48] Carpenter, K. G., Schrijver, C. J., and Karovska, M., “The Stellar Imager (SI) vision mission,” *Advances in Stellar Interferometry, Proceedings of the 2019 International Society for Optics and Photonics Conference (May 2006, Orlando, Florida)*, Vol. 6268, 2006, pp. 610–621. https://doi.org/10.1117/12.669713.

[49] Marchand, B. G., and Howell, K. C., “Control Strategies for Formation Flight In the Vicinity of the Libration Points,” *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 6, 2005, pp. 1210–1219. https://doi.org/10.2514/1.11016.

[50] Howell, K. C., and Marchand, B. G., “Natural and non-natural spacecraft formations near the L1 and L2 libration points in the Sun–Earth/Moon ephemeris system,” *Dynamical Systems*, Vol. 20, No. 1, 2005, pp. 149–173. https://doi.org/10.1080/1468936042000298224.

[51] Xin, M., Balakrishnan, S. N., and Pernicka, H. J., “Multiple spacecraft formation control with Θ-D method,” *IET Control Theory & Applications*, Vol. 1, 2007, pp. 485–493. https://doi.org/10.1049/iet-cta_20050410.

[52] Xin, M., Balakrishnan, S. N., and Pernicka, H. J., “Position and Attitude Control of Deep-Space Spacecraft Formation Flying Via Virtual Structure and Θ-D Technique,” *Journal of Dynamic Systems, Measurement, and Control*, Vol. 129, No. 5, 2007, pp. 689–698. https://doi.org/10.1115/1.2764509.

[53] Li, P., Cui, P., and Cui, H., “An improved nonlinear control strategy for deep space formation flying spacecraft,” *Acta Mechanica Sinica*, Vol. 25, No. 6, 2009, pp. 847–856. https://doi.org/10.1007/s10409-009-0274-2.

[54] Xu, M., Zhou, N., and Wang, J., “Robust adaptive strategy for stationkeeping of Halo orbit,” *2012 24th Chinese Control and Decision Conference (May 2012, Taiyuan, China)*, 2012, pp. 3086–3091. https://doi.org/10.1109/CCDC.2012.6244486.

[55] Queiroz, M. S. d., Kapila, V., and Yan, Q., “Adaptive Nonlinear Control of Multiple Spacecraft Formation Flying,” *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 3, 2000, pp. 385–390. https://doi.org/10.2514/2.4549.

[56] Gurfil, P., Idan, M., and Kasdin, N. J., “Neurocontrol of Spacecraft Formation Flying in the Elliptic Restricted Three-Body Problem,” *AIAA Guidance, Navigation, and Control Conference and Exhibit (August 2002, Monterey, California)*, 2002, pp. 1–11. https://doi.org/10.2514/6.2002-4962.

[57] Wang, W., Mengali, G., Quarta, A. A., and Yuan, J., “Distributed adaptive synchronization for multiple spacecraft formation flying around Lagrange point orbits,” *Aerospace Science and Technology*, Vol. 74, 2018, pp. 93–103. https://doi.org/10.1016/j.ast.2018.01.007.

[58] Xu, M., Liang, Y., and Fu, X., “Formation flying on quasi-halo orbits in restricted Sun–Earth/Moon system,” *Aerospace Science and Technology*, Vol. 67, 2017, pp. 118–125. https://doi.org/https://doi.org/10.1016/j.ast.2017.03.038.

[59] Shahid, K., and Kumar, K. D., “Formation Control at the Sun-Earth L2 Libration Point Using Solar Radiation Pressure,” *Journal of Spacecraft and Rockets*, Vol. 47, No. 4, 2010, pp. 614–626. https://doi.org/10.2514/1.47342.

[60] Wang, W., Baoyin, H., Mengali, G., and Quarta, A. A., "Solar sail cooperative formation flying around L2-type artificial equilibrium points," *Acta Astronautica*, Vol. 169, 2020, pp. 224–239. https://doi.org/10.1016/j.actaastro.2019.10.028.

[61] Qi, R., Xu, S., and Xu, M., "Impulsive Control for Formation Flight About Libration Points," *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 2, 2012, pp. 484–496. https://doi.org/10.2514/1.54383.

[62] Kane, T. R., and Marsh, E. L., "Attitude stability of a symmetric satellite at the equilibrium points in the restricted three-body problem," *Celestial mechanics*, Vol. 4, No. 1, 1971, pp. 78–90. https://doi.org/10.1007/BF01230323.

[63] Guzzetti, D., and Howell, K. C., "Natural periodic orbit-attitude behaviors for rigid bodies in three-body periodic orbits," *Acta Astronautica*, Vol. 130, 2017, pp. 97–113. https://doi.org/10.1016/j.actaastro.2016.06.025.

[64] Guzzetti, D., and Howell, K. C., "Attitude dynamics in the circular restricted three-body problem," *Astrodynamics*, Vol. 2, No. 2, 2018, pp. 87–119. https://doi.org/10.1007/s42064-017-0012-7.

[65] Vallado, D. A., *Fundamentals of Astrodynamics and Applications*, Microcosm Press, Fourth Edition, 2013.

[66] Galchenko, P., and Pernicka, H. J., "Precision Formation Flying and Spacecraft Pointing Using Plasmonic Force Propulsion," *Proceedings of the 2017 AAS/AIAA Astrodynamics Specialist Conference (August 2017, Stevenson, WA)*, Vol. 162, 2017, pp. 2985–2997.

[67] Carpenter, K. G., Schrijver, C. J., and Karovska, M., "SI – The Stellar Imager," Tech. rep., Vision Mission Study Report, NASA Goddard Space Flight Facility, 2017.

[68] Wertz, J. R., Everett, D. F., and Puschell, J. J., *Space Mission Engineering: The New SMAD*, Microcosm Press, First Edition, 2011.

[69] Rovey, J. L., Yang, X., Friz, P. D., Hu, C., and Glascock, M. S., "Plasmonic force propulsion revolutionizes nano/picosatellite capability," Tech. rep., NASA Innovative Advanced Concepts Phase I Final Report, 2014.

[70] Rovey, J. L., Friz, P. D., Hu, C., Glascock, M. S., and Yang, X., "Plasmonic Force Space Propulsion," *Journal of Spacecraft and Rockets*, Vol. 52, No. 4, 2015, pp. 1163–1168. https://doi.org/10.2514/1.A33155.

VITA

Pavel Galchenko was born in Oleksandriya, Ukraine and was raised by his parents, Fiodor and Valentina Galchenko. After graduating from Green Ridge High School in Missouri in 2003, he enrolled in State Fair Community College in Sedalia, Missouri, where he received his Associate of Arts in General Studies and Associate of Applied Science in Machine Tool Technologies in May 2006. He continued his undergraduate education at the University of Central Missouri where he received his Bachelor of Science in Nursing in December 2009. In January 2011, Pavel changed careers to pursue an engineering degree from the Missouri University of Science and Technology, where he received his Bachelor of Science in Aerospace Engineering in May 2015. After completing a summer internship with NASA Jet Propulsion Laboratories, he began his Ph.D. in Aerospace Engineering at the Missouri University of Science and Technology in August 2015. During his graduate career, he worked as a Graduate Research Assistant on a NASA Innovative Advanced Concepts project as well as teaching a lab course as a Graduate Teaching Assistant. In February 2019, he received a NASA Pathways position, where he worked for NASA Wallops Flight Facility and NASA Goddard Space Flight Center as he continued working towards completing his degree. Pavel received his Ph.D. in Aerospace Engineering from Missouri University of Science and Technology in May 2022. He began his full time employment with NASA Wallops Flight Facility, a part of NASA Goddard Space Flight Center, as an Aerospace Engineer in the Guidance, Navigation, and Control and Mission Systems Engineering Branch.