A SIX DEGREE-OF-FREEDOM SPACECRAFT DYNAMICS SIMULATOR FOR FORMATION CONTROL RESEARCH

Yashwanth Kumar Nakka[*]**, Rebecca C. Foust**[†]**,
Elena Sorina Lupu**[‡]**, David B. Elliott**[§]**, Irene S. Crowell**[¶]**,
Soon-Jo Chung**[‖]**, and Fred Y. Hadaegh**[**]

This paper presents a new six-degree-of-freedom robotic spacecraft simulator, the Multi-Spacecraft Testbed for Autonomy Research (M-STAR), for testing formation guidance, relative navigation, and control algorithms. The simulator dynamics are governed by five degrees of frictionless translational and rotational air-bearing motion and one degree of kinematic motion in the gravity direction with flight-like actuators, in a 1-g environment. M-STAR is designed to be modular and accommodates 3-DOF, 4-DOF, 5-DOF, and 6-DOF operation with minimal mechanical modifications. The simulator is modelled as a 3-D pendulum on a floating platform with sixteen thrusters and four reaction wheels as on-board actuators. Based on this plant model, a nonlinear hierarchical control law is proposed for position and attitude trajectory tracking. A weighted generalized pseudo-inverse strategy for control allocation to map control inputs to actuator inputs is discussed. The thruster actuation model for mapping smooth allocated input to non-smooth actuator input that achieves equivalent performance is derived. The control law, allocation scheme, and thruster model are tested on the simulator for real-time position tracking control using a Robot Operating System (ROS) based software framework.

INTRODUCTION

Spacecraft formation flying technologies using smallsats, such as microsatellites and CubeSats, as individual agents offer a robust, adaptable, and cost-effective way to establish space telescopes,[1] communication systems[2] for observation, and various other applications.[3–5] The ability of these systems to perform equivalent to a monolithic system depends on achieving high-precision relative navigation, guidance, control (GNC), and synchronization of the individual agents in the formation. A ground-based robotic dynamics simulator that can mimic the frictionless motion in a disturbance torque free environment with flight-like subsystems provides a platform to test and validate the GNC algorithms required to design and build such a multi-agent spacecraft system.

[*]Graduate Student, GALCIT, California Institute of Technology, Pasadena, CA, 91125, USA

[†]Graduate Student, Aerospace Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, 61801, USA

[‡]Research Engineer, GALCIT, California Institute of Technology, Pasadena, CA, 91125, USA

[§]Undergraduate, California Institute of Technology, Pasadena, CA, 91125, USA

[¶]Undergraduate, California Institute of Technology, Pasadena, CA, 91125, USA

[‖]Associate Professor of Aerospace and Bren Scholar; Jet Propulsion Laboratory Research Scientist, California Institute of Technology, Pasadena, CA, 91125, USA

[**]Senior Research Scientist and Technical Fellow, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, 91109, USA.

Organization	Name	DOF
Naval Postgraduate School	POSEIDYN	3
Georgia Institute of Technology	ASTROS	5
Florida Institute of Technology	ORION	6
University of Florida	ADAMUS	6
Yonsei University	ASTERIX	5
NASA Jet Propulsion Laboratory (JPL)	FCT	5
	SSDT	3
German Aerospace Center (DLR)	TEAMS	3 and 5
Massachusetts Institute of Technology	SPHERES	3
	ARGOS	3 (attitude)

Table 1: A sample of spacecraft simulators from other institutions. [7–23]

Historically, air bearing[6] platforms have been a popular choice to build spacecraft dynamics simulators. Air bearing spacecraft simulation platforms were developed by several research laboratories;[7–23] a selection of these simulation platforms is shown in Table 1. Existing air bearing platforms can be classified into four types based on the mode of operation: 3 degrees-of-freedom (DOF) planar,[9, 16, 20] 3-DOF attitude,[8, 10, 20] 5-DOF planar and attitude,[7, 14, 17, 22] and 6-DOF planar and attitude with gravity-axis motion.[13, 19, 21] The air bearing system acts as a ground-based simulator platform for flight-like actuators and sensors, which provides an opportunity to test flight algorithms and emulate space dynamics.[24]

In this paper, we describe the development of a new 6-DOF spacecraft simulator, the Multi-Spacecraft Testbed for Autonomy Research (M-STAR), that is designed to be modular and accommodates 3-DOF, 4-DOF, 5-DOF, and 6-DOF operation with minimal mechanical modifications. The spacecraft simulator hardware was designed to have decentralized control and information sharing capabilities with neighboring agents in view of the future goal of testing multi-agent GNC algorithms using up to five of these simulators. Each spacecraft has 16 thrusters and 4 reaction wheels to study fault-tolerant control.

In view of the model-based GNC algorithms a detailed nonlinear dynamic model for the 5-DOF system was derived by modelling it as a 3D pendulum on a gliding planar platform with a center of gravity offset in the 3D pendulum. The nonlinear dynamic model is decoupled by assuming a small center of gravity offset. A nonlinear hierarchical control law is proposed for fast attitude dynamics and slower position dynamics due to the time-scale separation. The control law computes forces and torques collocated to the dynamics. Control allocation[25] is done to map the collocated control signal to the actuator signal. Optimization formulations[26] can be used to solve the control allocation problem, typically formulated as a linear program. For the M-STAR control allocation, we implement a generalized pseudo-inverse method for control allocation with a weighted influence matrix to account for actuator saturation limits, as the optimization formulations are computationally expensive for real-time implementation.

The position control of the M-STAR is performed using on-off solenoids, which are inherently nonlinear due to mechanical delays and varying pressure in the manifold that supplies compressed air to the solenoids. The solenoids are characterized[27] by measuring the force produced for varying on-off time, using a calibrated load cell. A linear model to compute the on time of a thruster is developed using the measured data for a given force requirement at each time step. The control
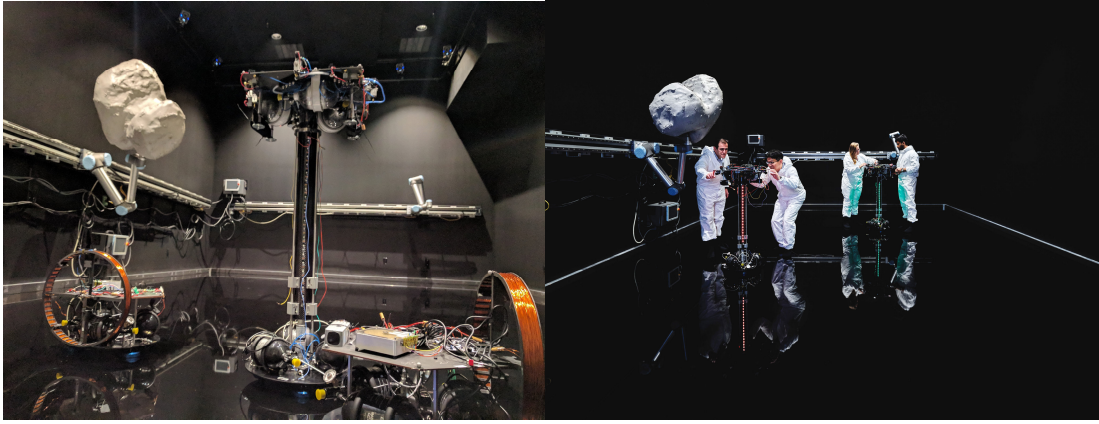
Figure 1: Multiple 6-DOF M-STAR spacecraft at Caltech's Aerospace Robotics and Control Lab.

law, control allocation scheme, and thruster model are tested for position tracking using a Robot Operating System (ROS) based software framework.

The paper is divided into four sections: 1) hardware description of the spacecraft simulator; 2) nonlinear dynamics and control law design of the M-STAR testbed; 3) discussion on actuator models and control allocation; and 4) preliminary position control experimental results.

THE SPACECRAFT DYNAMICS SIMULATOR FACILITY AT CALTECH

Overview of the Facility

The spacecraft simulator facility requires the following three components to be operational: the epoxy flat floor, the compressed air filling station, and the M-STAR. The epoxy flat floor is a high precision floor with flatness within 0.001 inches for frictionless translation of the spacecraft dynamics simulator using three flat air-bearing pads. Figure 1 shows the facility with multiple M-STAR spacecraft simulators and protection for collisions on the outer edge of the floor. The full 6-DOF spacecraft simulator can be seen in the middle with two 3-DOF simulators on the sides. The second component, the filling station, is comprised of an industrial air compressor and two 6,000 psi storage tanks. The filling station is used to fill the on-board air cylinders that supply air to the flat air bearings, spherical air bearing, and 16 on-off non-latching solenoid valves that act as thrusters on the simulator. The M-STAR shown in Fig. 2 acts as the dynamic simulation platform for a smallsat and includes all the necessary on-board sensors, actuator systems, and computing to achieve full 6-DOF control. The pose of the spacecraft simulator is estimated using 14 motion capture cameras mounted on the ceiling of the facility. In the following section, we elaborate on the subsystem hardware of the simulator.

M-STAR Spacecraft Simulator Hardware

The Caltech Aerospace Robotics and Controls Lab's 6-DOF spacecraft dynamics simulator for spacecraft formation control research was designed to accommodate up to a 12U CubeSat as a payload. The floating test bed simulates 5-DOF dynamic motion and 1-DOF kinematic motion along the gravity direction, with translation and attitude stages decoupled via a spherical air bearing. The translation stage floats frictionlessly on the precision floor using three flat round air bearings. The attitude stage has a hemispherical air bearing ball that floats frictionlessly on the cup mounted

at the top of the linear actuator on the translation stage. Tables 2 and 3 list the hardware components on both the translation stage and attitude stages respectively. The hardware on each stage is divided into three subsystems: 1) mechanical, including structural and pneumatic components; 2) electrical, including power, computing, and low level controller boards; and 3) actuation, to impart torque or impulse in the required degree of freedom. Each of these components plays an essential role in achieving torque-free controlled motion.
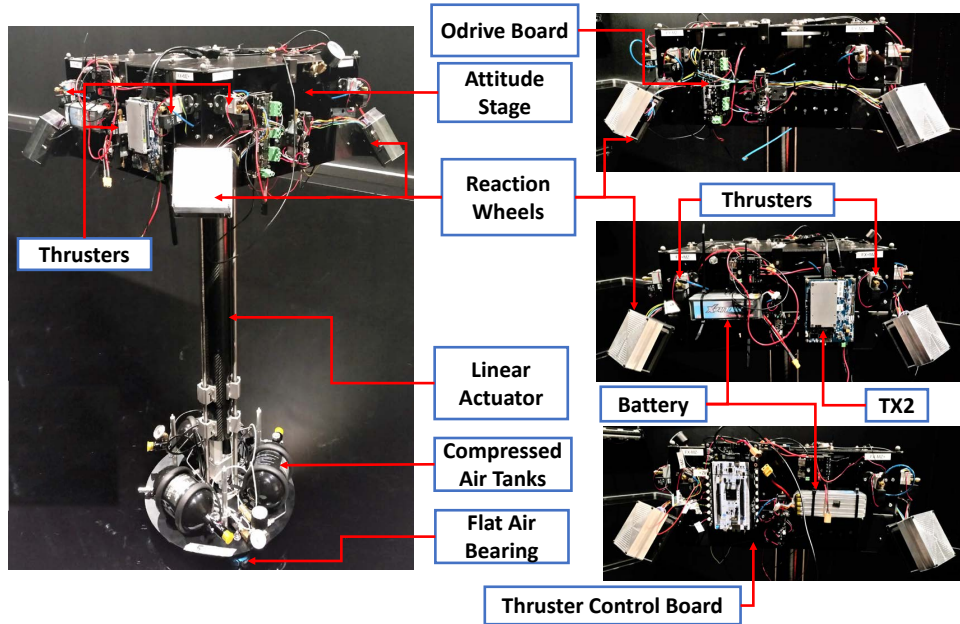
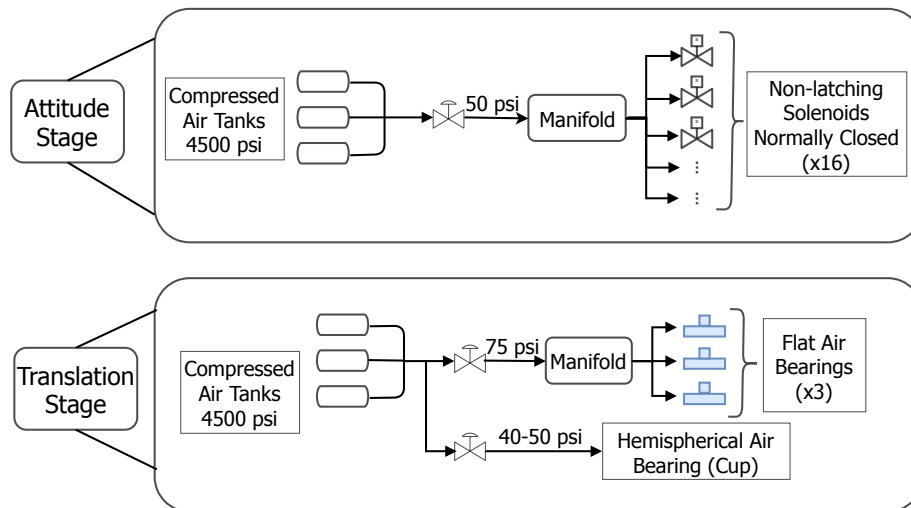**Figure 2**: M-STAR spacecraft dynamics simulator.

Figure 3: Flowchart of pneumatic system on translation and attitude stage.

Translation Stage. The translation stage provides frictionless in-plane motion for the whole simulator using three linear flat round air bearings. It consists of three compressed air cylinders running

at 4500 psi, a spherical air bearing cup, pneumatic components for pressure regulation, and tubing required to supply air for the bearings. The pneumatic system on the translation stage is shown in Fig. 3. In addition, it is equipped with a linear actuator, a brushless DC linear motor for achieving motion in the gravity direction with supporting control electronics.

The different operation modes of operation (3-DOF, 4-DOF, 5-DOF, and 6-DOF) can be achieved as follows:

- 3-DOF: spherical air bearing turned off and linear actuator replaced with a passive tube

- 4-DOF: spherical air bearing turned off

- 5-DOF: linear actuator replaced with a passive tube

- 6-DOF: all actuators active

This provides flexibility in operation and allows the construction of algorithms with increased complexity. The compressed air storage tanks' capacity was designed to achieve at least 25-30 minutes of flotation time at the operating pressure in 6-DOF mode. Several custom-designed add-ons can be incorporated on the translation stage such as docking ports and reaction wheels for yaw control.

Subsystem	Component
Mechanical	NewWay Air Bearing
	Compressed Air Cylinders
	Structure Design
	Spherical Air bearing
	Regulator
Actuator	Progressive Automation Linear Actuator
Electronics and Power	Battery
	Linear Actuator Controller
	Raspberry Pi

Table 2: List of components on the translation stage.

Attitude Stage. The attitude stage structure was designed using carbon fiber composites and honeycomb materials, optimized to provide a flotation time of up to 30 minutes with a payload of 12 kilograms. It has a box structure and acts as a platform for a potential payload, such as a 12U CubeSat. The attitude stage structure has the hemispherical ball of the air bearing pair and floats on the translation stage to provide 3-DOF frictionless attitude motion. This stage has 16 on-off non-latching solenoids with custom made nozzles and four in-house reaction wheels as actuators. The power distribution board for the attitude stage and the low-level controller for the thrusters are designed at Caltech. The schematic of the pneumatic subsystem for supplying regulated compressed air to the thrusters is shown in Fig. 3. It includes three compressed air cylinders, a regulator, and a manifold for air distribution. The regulated pressure is supplied to all the thrusters through the manifold to maintain the pressure across them. The operating pressure of the thrusters is decided based on experimental characterization of the solenoids. The electrical subsystem of the attitude stage is shown in Fig. 4. We chose an NVIDIA Jetson TX2 as the main computer to run the GNC and perception algorithms. The computer communicates the control signal to the low level boards as shown in Fig. 4. The subsystem components of the stages are listed in the Table 3.
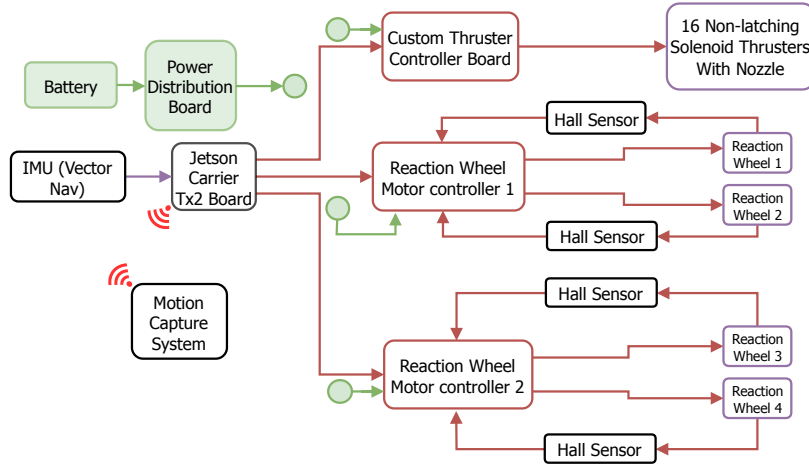
Figure 4: Attitude stage architecture.

Subsystem	Component
	Structure
Mechanical	Nozzles
	Pneumatics
Actuator	Thrusters
	Reaction Wheel Assembly
	Battery
	Power Distribution Board
Electronics and Power	Thruster Control Board
	ODRIVE Reaction Wheel Driver
	Maxon Motor Reaction Wheel Motor
	NVIDIA Jetson TX2 Computer

Table 3: List of components on the attitude stage.

Reaction Wheel Sizing and Manufacturing. Custom reaction wheels were designed for the spacecraft simulator to achieve certain nominal performance specifications. The principle axis of inertia of the whole simulator including a 12U CubeSat payload, shown in Eq. (1), was estimated using CAD software. The nominal torque and angular momentum requirements for attitude control corresponding to the estimated principle inertia of the spacecraft is shown in Table 4.

$$J = \begin{bmatrix} 1.19 & 0 & 0 \\ 0 & 1.24 & 0 \\ 0 & 0 & 1.43 \end{bmatrix} [\mathrm{kgm}^2] \tag{1}$$

The flywheel was made out of brass, fabricated using a CNC lathe for better tolerances. The motor selected was an EC frameless motor from Maxon Motor, which has a decoupled stator and the rotor, leading to an increased lifetime. The entire structure (flywheel, shaft, and motor) was constrained between two ball bearings to reduce vibration as shown in Fig. 5.

Nominal torque X-axis [Nm]	0.069
Nominal torque Y-axis [Nm]	0.072
Nominal torque Z-axis [Nm]	0.044
Angular momentum X-axis [Nms]	0.2077
Angular momentum Y-axis [Nms]	0.2164
Angular momentum Z-axis [Nms]	0.4492

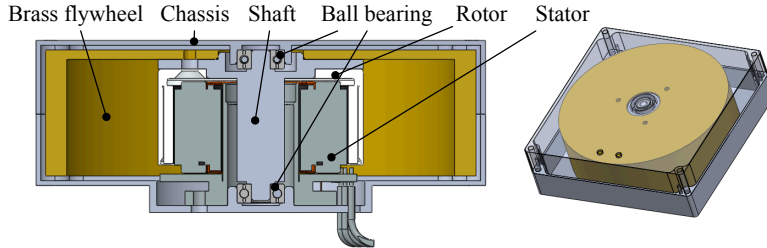**Table 4**: Nominal torque and angular momentum of the spacecraft.

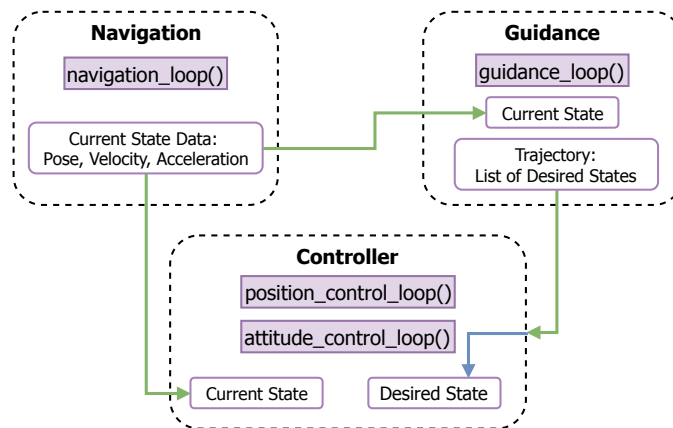**Figure 5**: Section view of Caltech's custom-made reaction wheels.

Figure 6: Software architecture design.

M-STAR Software Architecture

The software for the simulator was designed to allow for interchangeable guidance, navigation, and control modules. The architecture is implemented in C++ using abstract base classes for the three modules, with virtual loop functions for subclasses to implement. As illustrated in Fig. 6, navigation subclasses are responsible for generating updated state data for the guidance system and controller. The guidance system maintains a trajectory of desired states, from which the controller selects a target state for the current time step and implements the required dynamics. The current experimental setup features waypoint guidance, motion capture camera based navigation, and the 5-DOF controller outlined in the next section. However, these could respectively be swapped for an arbitrary motion-planning algorithm, pose feedback from integrated on board sensor data, and controllers for the four configurations of the simulator.

The architecture is built on Robotic Operating System (ROS) framework, which allows for each loop to be scheduled at a unique rate that can be changed at run time. Data from other modules is automatically fetched before each loop runs. ROS also provides a messaging architecture for communicating with peripheral boards, the ability to create unique launch configurations for different module setups, and test logging.

DYNAMICS AND CONTROL

Each M-STAR has two links coupled using a spherical air bearing as a joint. This system can be modelled as a three dimensional pendulum on a floating platform with a ball joint to provide 3-DOF rotation of the pendulum (modelling the attitude of the spacecraft) and 2-DOF planar motion of the floating platform. Constraints on the 3D pendulum motion due to mechanical interference between the attitude stage and the translation stage are shown in Table 5. The coordinate systems used for deriving the kinematics and dynamics of the system are shown in Fig. 7. The inertial reference frame on the test floor is defined by the coordinate system (X_i, Y_i, Z_i) with origin O_i . A non-rotating reference frame (X_{ib}, Y_{ib}, Z_{ib}) that is parallel to the inertial frame, is attached to the attitude stage with origin O_b at the center of the hemispherical bearing to define the orientation of the attitude stage. The attitude stage dynamics are derived in terms of the angular rates in the body frame (X_b, Y_b, Z_b) at origin O_b. Before proceeding to the discussion on the dynamics and control implementation, the attitude representation used for describing the motion of the 3D pendulum in SO(3) space is discussed.

Pitch (rotation about X_{ib})	$\pm 45°$
Roll (rotation about Y_{ib})	$\pm 45°$
Yaw (rotation about Z_{ib})	$\pm 180°$

Table 5: Constraints on the angular motion of the attitude stage.

Attitude Kinematics

The attitude of the 3D pendulum can be represented by any attitude representations including quaternions,[28] Modified Rodrigues Parameters (MRPs),[28] and SO(3) rotation matrix. For example, the MRPs $p \in \mathcal{R}^3$ are stereographic projections of the unit quaternions,[28] $q \in \mathcal{H}$, where $\mathcal{H}$ is the Hamiltonian space and have a bijective mapping to the quaternion sphere are used here. The attitude representation in MRPs takes into account the unit norm of the quaternions. The attitude kinematics equation is given using the body angular rates $\omega \in \mathcal{R}^3$. The kinematics of MRPs are given as follows.

$$\dot{p} = Z(p)\omega; \text{ where } Z(p) = \frac{1}{2}\left(I_3\left(\frac{1-p^T p}{2}\right) + pp^T + S(p)\right), S(p) = \begin{bmatrix} 0 & -p_3 & p_2 \\ p_3 & 0 & -p_1 \\ -p_2 & p_1 & 0 \end{bmatrix} \quad (2)$$

The rotation matrix $R(p)^\top$ to transform from the frame (X_{ib}, Y_{ib}, Z_{ib}) to the body frame (X_b, Y_b, Z_b) in terms of the MRPs is given as:

$$R(p)^\top = I_{3\times 3} - \frac{4(1-p^T p)}{(1+p^T p)^2}S(p) + \frac{8}{(1+p^T p)^2}S(p)^2 \quad (3)$$

The transformation $R(p)$ is used in mapping the external force due to thrusters in the body frame to the inertial frame for controlling the translation dynamics.
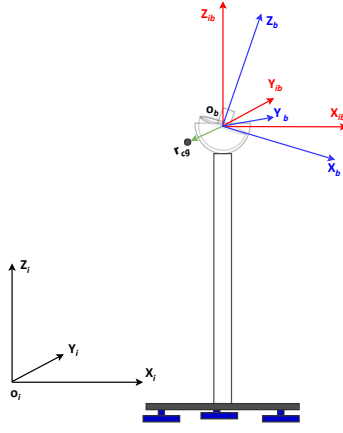
8

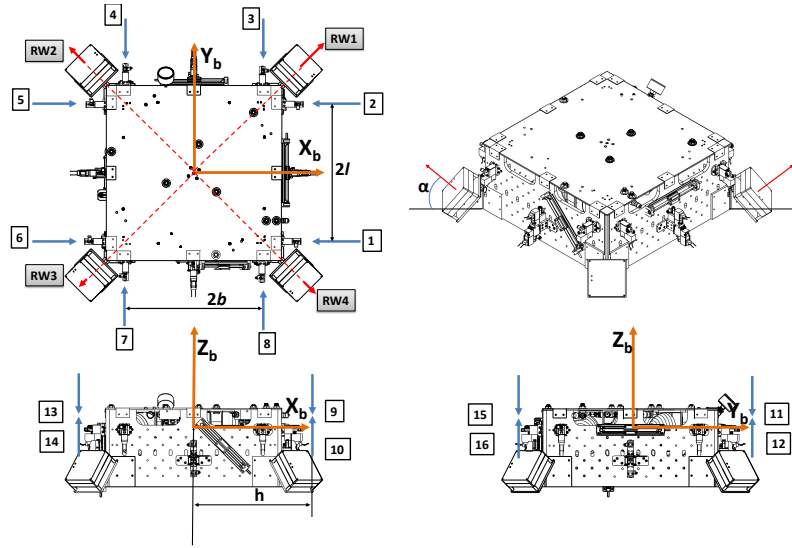**Figure 7**: Coordinate Systems used for the derivation of the dynamic model.

Figure 8: Attitude Stage with actuator configuration and nomenclature in the body frame.

Nonlinear Dynamic Model

The dynamics of the 5-DOF system with the velocity v_b at the centre of rotation of the attitude stage and angular rates of the attitude stage ω in body frame (X_b, Y_b, Z_b) is given in the Eq. (5), where, r_{cg} is the center of gravity offset from the center of rotation of the attitude stage in the body frame coordinates, J is the mass moment of inertia of the attitude stage about the center of rotation in the body frame, $R(p)$ is defined in Eq. (3), (x, y) is the planar location of the center of rotation from the inertial frame origin, m_a is the mass of the attitude stage, and m_t is the mass of the translation stage. In the following equations, $a_1 = [1; 0; 0]$, $a_2 = [0; 1; 0]$ and $a_3 = [0; 0; 1]$ are unit vectors in the reference frame (X_{ib}, Y_{ib}, Z_{ib}).

$$\dot{P} = R(p)v_b \quad \text{where} \quad \bar{P} = (x, y, 0)^\top, \quad D = (a_1^\top; a_2^\top; 0) \tag{4}$$

$$M_b(p) \begin{bmatrix} \dot{\omega} \\ \dot{v}_b \end{bmatrix} + C_b \begin{bmatrix} \omega \\ v_b \end{bmatrix} + H_b = \tau_b \tag{5}$$

9

$$M_b(p) = \begin{bmatrix} J & m_a S(r_{cg})R(p)^\top DR(p) \\ m_a \left(S(r_{cg})R(p)^\top DR(p) \right)^\top & (m_a + m_t) \end{bmatrix} \tag{6}$$

$$C_b = \begin{bmatrix} -S(J\omega) & m_a S(r_{cg})R(p)^\top DR(p)S(\omega) \\ -m_a R(p)^\top DR(p)S(\omega)S(r_{cg}) & (m_t + m_a)S(\omega) \end{bmatrix} \tag{7}$$

$$H_b = \begin{bmatrix} -m_a g S(r_{cg})R(p)^\top a_3 \\ 0 \end{bmatrix} \tag{8}$$

The control inputs to the system are represented by $\tau_b = \left[\begin{smallmatrix} \tau_f \\ \tau_t \end{smallmatrix}\right]$, which include forces due to thrusters τ_f and torques τ_t due to thrusters and reaction wheels in body frame. The control design is done in body frame. The forces computed in body frame τ_f are transformed to forces in inertial frame $\tau_p = R(p)\tau_f$ for implementation of the position control law. The implementation of the transformation and the influence of thrusters in the body frame on the position dynamics in inertial frame is discussed in the following sections. In the body frame, for the 5-DOF dynamics in Eq. (5) it can be proved that $\dot{M}_b - (C_b + C_b^T) = 0$ and that $\dot{M}_b - 2C_b$ is a skew-symmetric matrix. The matrix form in Eq. (5) will be used in the following section to derive a controller that globally exponentially tracks a given position and almost globally exponentially tracks an attitude trajectory.

Control Design for Full Nonlinear Dynamics

The objective of the control design is to ensure that the 5-DOF of M-STAR, $[\bar{P}(t), p(t)]$ given in Eq. (4), exponentially tracks a given trajectory $[\bar{P}_d(t), p_d(t)] \in C^2([0, \infty])$. The following theorem states the nonlinear control law and proves the global exponential stability of the closed-loop system in Eq. (11). Here the variables $s_\omega = \omega - \omega_r$ and $s_v = v_b - v_{b_r}$ define the states for virtual dynamics. The variables w_r and v_{b_r} define the reference signal computed from filtered desired states dynamics given in the following Eq. (9).

$$\begin{aligned} \omega_r &= Z^{-1}(p)\dot{p}_d(t) + Z^{-1}(p)\Lambda_\omega(p_d(t) - p) \\ v_{b_r} &= R^\top \dot{\bar{P}}_d(t) + R^\top \Lambda_v(\bar{P}_d(t) - \bar{P}) \end{aligned} \tag{9}$$

Theorem 1. *The closed-loop system in terms of virtual states s_ω, s_v, given in Eq. (11), with the control law Eqs. (9–10), is globally exponentially stable in the sense of the Euclidean norm, assuming the feedback gains $K_\omega, K_v, \Lambda_\omega, \Lambda_v > 0$ and the inertia matrix M_b is positive definite and uniformly bounded with lower bound $\lambda_{\min}$ and upper bound $\lambda_{\max}$.*

$$\tau_b = M_b \begin{bmatrix} \dot{\omega}_r \\ \dot{v}_{b_r} \end{bmatrix} + C_b \begin{bmatrix} \omega_r \\ v_{b_r} \end{bmatrix} + H_b - \begin{bmatrix} K_\omega & 0 \\ 0 & K_v \end{bmatrix} \begin{bmatrix} s_\omega \\ s_v \end{bmatrix} \tag{10}$$

$$M_b \begin{bmatrix} \dot{s}_\omega \\ \dot{s}_v \end{bmatrix} + C_b \begin{bmatrix} s_\omega \\ s_v \end{bmatrix} + \begin{bmatrix} K_\omega & 0 \\ 0 & K_v \end{bmatrix} \begin{bmatrix} s_\omega \\ s_v \end{bmatrix} = 0 \tag{11}$$

Proof. The inertia matrix M_b, due to the properties of positive definiteness and uniform boundedness, is used to compute the norm $V = \frac{1}{2} \begin{bmatrix} \delta s_\omega \\ \delta s_v \end{bmatrix}^\top M_b \begin{bmatrix} \delta s_\omega \\ \delta s_v \end{bmatrix}$, for Lyapunov-like stability analysis.[29, 30] The closed-loop system in Eq. (11) has two particular solutions $[s_\omega, s_v] = 0$ and $[s_\omega, s_v] = s_e = [\omega - \omega_r, v_b - v_{b_r}]$. We perform a squared length analysis using the norm, after obtaining the

infinitesimal distance δs_e at fixed time. The derivative of the squared length is given in the following equation.

$$\dot{V} = \frac{1}{2} \begin{bmatrix} \delta s_\omega \\ \delta s_v \end{bmatrix}^\top \dot{M}_b \begin{bmatrix} \delta s_\omega \\ \delta s_v \end{bmatrix} + \begin{bmatrix} \delta s_\omega \\ \delta s_v \end{bmatrix}^\top M_b \begin{bmatrix} \dot{\delta s_\omega} \\ \dot{\delta s_v} \end{bmatrix} \tag{12}$$

Using the closed-loop dynamics and $s^\top \left(\dot{M}_b - 2C_b \right) s = 0$, on the right hand side of the above equation we get the following.

$$\begin{aligned} \dot{V} &= \frac{1}{2} \begin{bmatrix} \delta s_\omega \\ \delta s_v \end{bmatrix}^\top \left(\dot{M}_b - 2C_b - 2 \begin{bmatrix} K_\omega & 0 \\ 0 & K_v \end{bmatrix} \right) \begin{bmatrix} \delta s_\omega \\ \delta s_v \end{bmatrix} \\ &= - \begin{bmatrix} \delta s_\omega \\ \delta s_v \end{bmatrix}^\top \begin{bmatrix} K_\omega & 0 \\ 0 & K_v \end{bmatrix} \begin{bmatrix} \delta s_\omega \\ \delta s_v \end{bmatrix} \end{aligned} \tag{13}$$

With the inertia matrix bounds and $K = \min\{K_\omega, K_v\}$, we get the following inequality.

$$\dot{V} \leq -\frac{2K}{\lambda_{\max}\{M_b\}} V \tag{14}$$

Using Comparison Lemma[29] and the bounds on the inertia matrix M_b, we obtain:

$$V(t) \leq \exp\left(-\frac{2K}{\lambda_{\max}\{M_b\}} t \right) V(0)$$

$$\left\| \begin{bmatrix} \delta s_\omega \\ \delta s_v \end{bmatrix} \right\|_2 \leq \sqrt{\frac{\lambda_{\max}\{M_b\}}{\lambda_{\min}\{M_b\}}} \exp\left(-\frac{K}{\lambda_{\max}\{M_b\}} t \right) \left\| \begin{bmatrix} \delta s_\omega(0) \\ \delta s_v(0) \end{bmatrix} \right\|_2 \tag{15}$$

It follows from contraction-based incremental stability analysis[30, 31] that all the system trajectories of the closed-loop system in Eq. (11) converge exponentially fast with rate $\frac{2K}{\lambda_{\max}\{M_b\}}$, i.e. $\delta s_e \to 0$, which implies $[\omega - \omega_r, v_b - v_{b_r}] \to 0$. With converged virtual dynamics, from the definition of reference trajectories in Eq. (9), we get the following equations.

$$\begin{aligned} \omega - \omega_r &= Z^{-1}(p)(\dot{p} - \dot{p}_d(t)) + Z^{-1}(p)\Lambda_\omega(p(t) - p_d(t)) \\ v_b - v_{b_r} &= R^\top(\dot{\bar{P}} - \dot{\bar{P}}_d(t)) + R^\top\Lambda_v(\bar{P}(t) - \bar{P}_d) \end{aligned} \tag{16}$$

From Eq. (16), it is clear that the attitude trajectory and the position trajectory converge globally exponentially fast to the desired trajectory as $[\omega - \omega_r, v_b - v_{b_r}] \to 0$. The closed-loop virtual dynamics with a bounded disturbance at input $d(t) = \begin{bmatrix} d_\omega(t) \\ d_v(t) \end{bmatrix}$ is as following.

$$M_b \begin{bmatrix} \dot{s}_\omega \\ \dot{s}_v \end{bmatrix} + C_b \begin{bmatrix} s_\omega \\ s_v \end{bmatrix} + \begin{bmatrix} K_\omega & 0 \\ 0 & K_v \end{bmatrix} \begin{bmatrix} s_\omega \\ s_v \end{bmatrix} = \begin{bmatrix} d_\omega(t) \\ d_v(t) \end{bmatrix} \tag{17}$$

The Lyapunov derivative with the closed-loop system as in Eq. (17) can be simplified to following equation using Cauchy-Schwarz inequality and the bounds on the inertia matrix M_b.

$$\dot{V} = -\frac{2K}{\lambda_{\max}\{M_b\}} V + \sqrt{\frac{2V}{\lambda_{\min}\{M_b\}}} \|d(t)\|_2 \tag{18}$$

Using the transformation $W = \sqrt{V}$, followed by application of Comparison Lemma[29] and the uniform bounds on the inertia matrix M_b, we get the bounds on the norm of the virtual coordinates, where $\sup_{t \geq 0} |d(t)| \leq \gamma_\infty$.

$$
\left\| \begin{bmatrix} \delta s_\omega \\ \delta s_v \end{bmatrix} \right\|_2 \leq \sqrt{\frac{\lambda_{\max}\{M_b\}}{\lambda_{\min}\{M_b\}}} \exp\left(-\frac{K}{\lambda_{\max}\{M_b\}} t\right) \left\| \begin{bmatrix} \delta s_\omega(0) \\ \delta s_v(0) \end{bmatrix} \right\|_2
$$
$$
+ \frac{\lambda_{\max}\{M_b\}\gamma_\infty}{\lambda_{\min}\{M_b\}K} \left(1 - \exp\left(-\frac{Kt}{\lambda_{\max}\{M_b\}}\right)\right)
\tag{19}
$$

By taking a limit $t \to \infty$, we get the bounds on the virtual states $\left\| \begin{bmatrix} \delta s_\omega \\ \delta s_v \end{bmatrix} \right\|_2 \to \frac{\lambda_{\max}\{M_b\}\gamma_\infty}{\lambda_{\min}\{M_b\}K}$. Thus the exponentially stable closed-loop virtual dynamics, Eq. (11), is finite-gain $\mathcal{L}_p$ stable and Input-to-State Stable (ISS) for a bounded disturbance $d \in \mathcal{L}_p$ at the input.[30] $\qquad\square$

Control Implementation

For the control implementation, it is assumed that the attitude stage is coarsely balanced with small r_{cg}. Equation (20) shows the decoupled translation dynamics in inertial frame and rotational dynamics in body frame with small center of gravity offset. The terms in the dynamics corresponding to the r_{cg} act as a bounded disturbance at the input $d(t) = \begin{bmatrix} d_w(t) \\ d_p(t) \end{bmatrix}$ for small accelerations.

$$
\begin{bmatrix} J & 0 & 0 \\ 0 & m_a + m_t & 0 \\ 0 & 0 & m_a + m_t \end{bmatrix} \begin{bmatrix} \dot\omega \\ \ddot x \\ \ddot y \end{bmatrix} + \begin{bmatrix} \omega \times J\omega \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} -m_a g S(r_{cg})R(p)^\top a_3 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \tau_r \\ \tau_p \end{bmatrix} + \begin{bmatrix} d_w(t) \\ d_p(t) \end{bmatrix}
\tag{20}
$$

$$
d_w(t) = -m_a S(r_{cg})R(p)^\top a_1 \ddot x - m_a S(r_{cg})R(p)^\top a_2 \ddot y
$$
$$
d_p(t) = \begin{bmatrix} -m_a(S(r_{cg})R(p)^\top a_1)^\top \dot\omega - m_a a_1^\top R(p)S(\omega)^2 r_{cg} \\ -m_a(S(r_{cg})R(p)^\top a_2)^\top \dot\omega - m_a a_2^\top R(p)S(\omega)^2 r_{cg} \end{bmatrix}
\tag{21}
$$

A hierarchical control law was implemented with an inner attitude control loop and an outer position control loop because of the timescale separation between the two dynamics, Eq. (20). Given a desired position trajectory, $[x_d(t), y_d(t)] \in \mathcal{R}^2$, and attitude trajectory represented in MRPs, $p_d(t) \in \mathcal{R}^3$, the control law presented below exponentially tracks both position and attitude trajectories using smooth control inputs for the decoupled dynamics for no disturbance. In the case with a bounded disturbance at the input, the closed-loop system is finite-gain $\mathcal{L}_p$ stable. The control input to the position dynamics is simplified from Eq. (10) and is given by Eq. (22).

$$
\tau_p = (m_t + m_a) \begin{bmatrix} \ddot x_d \\ \ddot y_d \end{bmatrix} - K_d \begin{bmatrix} \dot x - \dot x_d \\ \dot y - \dot y_d \end{bmatrix} - K_p \begin{bmatrix} x - x_d \\ y - y_d \end{bmatrix}
\tag{22}
$$

$$
(m_t + m_a) \begin{bmatrix} \ddot x - \ddot x_d \\ \ddot y - \ddot y_d \end{bmatrix} - K_d \begin{bmatrix} \dot x - \dot x_d \\ \dot y - \dot y_d \end{bmatrix} - K_p \begin{bmatrix} x - x_d \\ y - y_d \end{bmatrix} = d_p(t)
\tag{23}
$$

The closed-loop position dynamics with the control law in Eq. (22) are given in Eq. (23). The gain values K_d and K_p are chosen to achieve the required position tracking performance. The attitude controller in Eq. (24) is exponentially stable[32] with no disturbance and tracks a given desired attitude trajectory that is $\mathcal{C}^2$ continuous. It can be shown that this control law is simplified form of

the controller proposed in Eq. (10). The nonlinear controller is finite-gain $\mathcal{L}_p$ stable with bounded disturbance at the input.

$$\tau_r = J\dot{\omega}_r - S(J\omega)\omega_r - K_r(\omega - \omega_r) - m_a g S(r_{cg})R(p)^\top a_3$$
$$\omega_r = Z^{-1}(p)\dot{p}_d(t) + Z^{-1}(p)\Lambda_r(p_d(t) - p) \tag{24}$$

$$J(\dot{\omega} - \dot{\omega}_r) - S(J\omega)(\omega - \omega_r) - K_r(\omega - \omega_r) = d_w(t) \tag{25}$$

The closed-loop attitude dynamics are given in the Eq. (25). The matrices Λ_r and K_r are positive definite and are chosen to achieve required tracking performance. The control laws presented above compute control signals which are at least $\mathcal{C}^2$ continuous and the number of control inputs are collocated with the states. Considering the overactuated design of the simulator and the impulse actuation of the thrusters, a transformation from the continuous control signal to the thruster on-off times is required to achieve equivalent performance with non-smooth control inputs. In the following two sections, we discuss the actuator models for thrusters and reaction wheels to make this transformation, along with the influence matrices due to the location of the actuators.

Thruster Model and Influence Matrix

Influence Matrix. Equations (22) and (24) give force and torque inputs that need to be applied collocated with the five degrees of freedom of the system. The spacecraft has 16 thrusters mounted in the configuration shown in Fig. 8, with thrusters 1-8 used for position and yaw angle control, and 9-16 used for roll and pitch angle control. The collocated force and torque inputs from the control law are transformed to the force input requirements on each of the 16 actuators through control allocation using an influence matrix. For the position controller, the following is the actuator input to control input mapping called the influence matrix.

$$\tau_p = R(p)B_p F_1 \tag{26}$$

In the equation 26, $R(p)$ transforms the actuator input in the body frame to the inertial frame. B_p corresponds to the influence matrix given by Eq. (27) for position control. The force vector, $F_1 = [f_1\ f_2\ f_3\ f_4\ f_5\ f_6\ f_7\ f_8]^\top$, acts as the input to the spacecraft dynamics simulator thrusters mounted for position and yaw control. The actuator numbering is shown in Fig. 8.

$$B_p = \begin{bmatrix} -1 & -1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & -1 & -1 & 0 & 0 & 1 & 1 \end{bmatrix} \tag{27}$$

For attitude control, the thruster force to control input mapping is given as follows.

$$\tau_r = B_r F \quad \text{where } B_r = [B_1\ B_2] \text{ and } F = \begin{bmatrix} F_1^\top & F_2^\top \end{bmatrix}^\top \tag{28}$$

where $F_2 = [f_9\ f_{10}\ f_{11}\ f_{12}\ f_{13}\ f_{14}\ f_{15}\ f_{16}]^\top$. Also, see Fig. 8 for the thruster numbering and nomenclature of ℓ, b, and h.

$$B_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\ell & \ell & -b & b & -\ell & \ell & -b & b \end{bmatrix},$$

$$B_2 = \begin{bmatrix} 0 & 0 & h & -h & 0 & 0 & -h & h \\ h & -h & 0 & 0 & -h & h & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{29}$$

13

Control Allocation. The control allocation scheme for the position controller computes the 8 dimensional thruster forces F_1 given the transformation matrix $R(p)$ and the influence matrix B_p. A generalized right psuedo-inverse solution to the control allocation problem that minimizes the $\mathcal{L}^2$-norm of the control effort is given by $F_1 = B_p^\top (B_p B_p^\top)^{-1} R(p)^{-1} \tau_p$ and weighted pseudo-inverse is given in Eq. (30), with a diagonal weighing matrix W. We use this algorithm for real-time implementation.

$$F_1 = W(B_p W)^\top (B_p W (B_p W)^\top)^{-1} R(p)^{-1} \tau_p \tag{30}$$

The elements of the diagonal matrix W can be chosen to take into account actuator saturation limits. For example, given the maximum u_{max} and minimum u_{min} thrust that can be produced by the thruster f_1, we choose the corresponding diagonal element in W as $\frac{1}{|u_{\text{max}} - u_{\text{min}}|}$. For attitude control using thrusters and reaction wheels, the same approach can be used for computing the actuator force.

Thruster Firing Time. The continuous actuator force computed using the control allocation scheme needs to be transformed to the thruster firing times because the thrusters on the spacecraft simulator are on-off non-latching solenoids. The on time of the thrusters is controlled using a PWM signal with the duty cycle mapped to the on time requirements. Consider a PWM signal with frequency f_{pwm} with duty cycle corresponding to firing time Δt, and continuous force F_r that needs to be applied by a thruster at time step t. Let F_a be the force applied by the thruster when open/on and the control loop frequency be f_{cl}. It is assumed that control frequency is same as the PWM signal frequency. The firing time is given in the following equation.

$$\Delta t = \frac{F_r}{f_{\text{cl}} F_a} \tag{31}$$

The equation above assumes that the thruster produces the same force for all firing times. To verify this claim and validate the model, an experimental setup was built as shown in Fig. 9. In the following section, we discuss the details of the experimental setup and the thruster model obtained from experiments.

Experimental Characterization of the Solenoidal Thruster. The experimental setup built to characterize the performance of the solenoidal thrusters includes a thruster mounted on a load cell with a regulated power supply. An instrumental amplifier is used to amplify the load cell output voltage, and the amplified voltage is sampled by a dSPACE MicroLabBox at $1\,\text{kHz}$. The thruster is supplied with pressure-regulated compressed air at 40 psi, 50 psi, or 60 psi, the three operating pressures tested. The thrusters are fired for a multiple of $10\,\text{ms}$ between $10\,\text{ms}$ and $80\,\text{ms}$, and the load cell force is recorded as a function of time for the duration of firing, as shown in Fig. 9b.

Figure 10a shows the experimental data and the linear fit of the average force produced by the thruster for varying firing time. The force produced is not constant due to the nonlinear behaviour of the thruster. The impulse produced with varying firing time is linear as seen in Fig. 10b. For the control implementation, we use the linear fit equations in Table 6 to compute the firing time of a thruster for a given force F_r that needs to be applied at time t with control loop frequency f_{cl}.
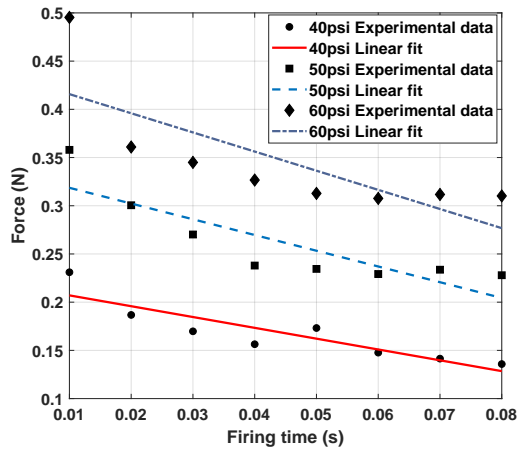
14
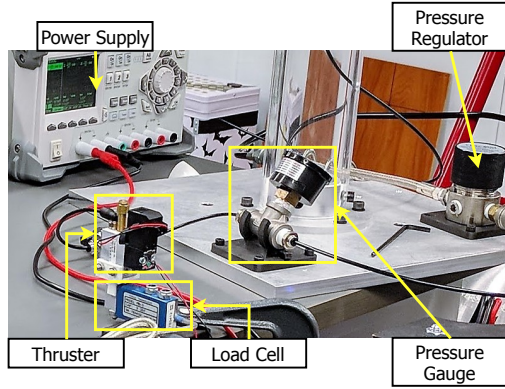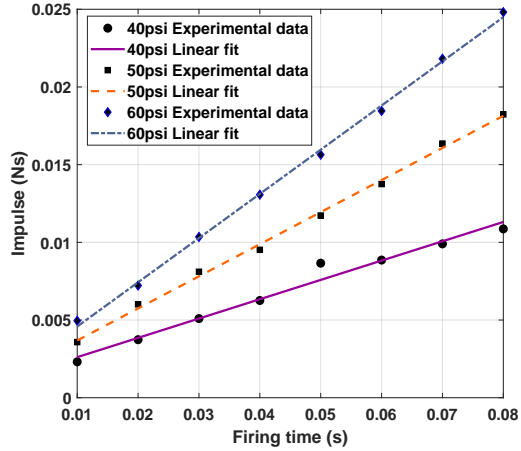
(a) Thruster characterization experimental setup.
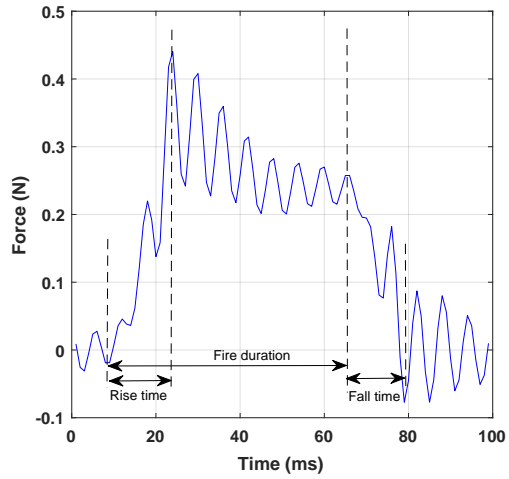
(b) Example force output for $50\,\mathrm{ms}$ fire duration.

Figure 9: Thruster characterization setup and sample results recorded by the load cell, showing rise time and fall time.

(a) Thruster firing time vs. average force.

(b) Thruster firing time vs. average impulse.

Figure 10: Experimental data and linear fit of average force and impulse vs. thruster firing time at 40 psi, 50 psi, and 60 psi operating pressure.

Operating Pressure (psi)	Fit equation
40	$\Delta t = 7.863\,\frac{F_r}{f_{\mathrm{cl}}}\ \text{-0.009727}$
50	$\Delta t = 4.829\frac{F_r}{f_{\mathrm{cl}}}\ \text{-0.007686}$
60	$\Delta t = 3.51\frac{F_r}{f_{\mathrm{cl}}}\ \text{-0.006035}$

Table 6: Linear fit equations for firing time computation for a given control signal F_r at time t and control frequency f_{cl}.

Reaction Wheel Configuration and Model

The simulator is equipped with four reaction wheels for attitude control arranged in a pyramid configuration (see Fig. 8). The angle α made by the axis of the wheel and the (X_b, Y_b) plane is chosen to have maximum momentum storage,[33] $\alpha = 35.26°$. The overactuated configuration will be used to study the fault detection, isolation, and recovery of reaction wheels, which is a major source of failure[34] in flight missions. The attitude dynamics with four reaction wheels in the pyramid configuration and no gravity torques is given in Eq. (32). The influence matrix is given by G in Eq. (33).

$$J\dot{\omega} + \omega \times J\omega = -GJ_w\dot{\Omega} - \omega \times GJ_w\Omega \tag{32}$$

$$G = \begin{bmatrix} c(\alpha)c(45°) & -c(\alpha)c(45°) & -c(\alpha)c(45°) & c(\alpha)c(45°) \\ c(\alpha)s(45°) & c(\alpha)s(45°) & -c(\alpha)s(45°) & -c(\alpha)s(45°) \\ s(\alpha) & s(\alpha) & s(\alpha) & s(\alpha) \end{bmatrix} \tag{33}$$

$$J_w = \begin{bmatrix} J_{w1} & 0 & 0 & 0 \\ 0 & J_{w2} & 0 & 0 \\ 0 & 0 & J_{w3} & 0 \\ 0 & 0 & 0 & J_{w4} \end{bmatrix} \tag{34}$$

In the above equation, J is the mass moment of inertia including the four wheels, J_w is a diagonal matrix with the mass moment of inertia of the wheels about the rotation axis, $\Omega = [\Omega_1 \ \Omega_2 \ \Omega_3 \ \Omega_4]^\top$ is the rotation speed of the each of the four wheels, and $s(\cdot), c(\cdot)$ denote the sine and cosine of a given angle, respectively. For the numbering and location of the wheels with respect to body frame see Fig. 8. The term $-GJ_w\dot{\Omega}$ is the control input to the attitude dynamics. The attitude controller in Eq. (24), is modified to cancel the cross-coupling term $-\omega \times GJ_w\Omega$ by feeding the wheel speed to the control law. The final control law is given in the Eq. 12d.

$$\tau_b = M_b \begin{bmatrix} \dot{\omega}_r \\ \dot{v}_{b_r} \end{bmatrix} + C_b \begin{bmatrix} \omega_r \\ v_{b_r} \end{bmatrix} + H_b - \begin{bmatrix} S(GJ_w\Omega) & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \omega_r \\ v_{b_r} \end{bmatrix} - \begin{bmatrix} K_\omega & 0 \\ 0 & K_v \end{bmatrix} \begin{bmatrix} s_\omega \\ s_v \end{bmatrix} \tag{35}$$

The wheel torques can be computed using the generalized pseudo-inverse from the control inputs. The reaction wheels are designed to run at the nominal speeds $[-2500, 2500, -2500, 2500]$ rpm, which is the null space of the influence matrix G, to avoid excitation of the attitude dynamics. The speed control of the wheel is done using a Hall sensor integrated into the selected Maxon motor.

Hardware Implementation of the Hierarchical Control Law

In this section, we elaborate on the implementation of the hierarchical control law discussed earlier. The schematic of the control law is shown in Fig. 11. The attitude control is done in the inner-loop with control frequency between $80 - 100$ Hz using reaction wheels. The thrusters can be used to do coarse attitude control, or desaturate the reaction wheels. The X, Y position controller is done using thrusters, it is coupled with the attitude dynamics by a rotation matrix to map the actuator force in the body frame to the inertial frame. The position dynamics are slow compared to the attitude dynamics, so it is run as an outer-loop with feedback on position data for control computations and attitude data for control allocation at control frequency between $1 - 10$ Hz.
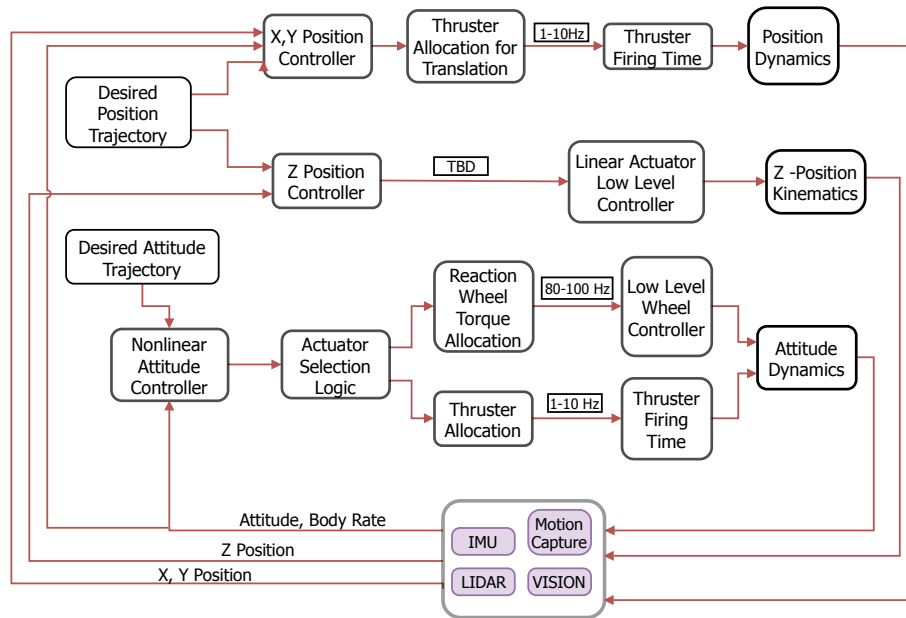
Figure 11: Closed-loop control implementation for the 6DOF simulator.
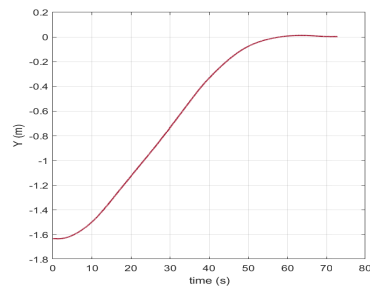
EXPERIMENTS

In this section, we present the preliminary experimental results for the position tracking controller discussed earlier. Here, we try to track a step input and demanding harmonic trajectories using the control law, control allocation and firing time schemes developed in the paper. The position and orientation data of the simulator is measured using the motion capture camera system running at 100 Hz. The thrusters are operated at 50 psi. The tracking results are discussed in the following section.

Results

Figures 12 and 13 show preliminary results of waypoint reaching experiments. The task for the controller was to reach origin of the inertial frame and stay there until a further command was communicated. The controller performs well for the two presented cases. The current position controller can be easily extended for tracking a trajectory with coarse way points. The steady-state error in both of the cases was less than the assigned value of 5cm. In this particular test the yaw angle attitude was coarsely maintained around 0, except when the system faced perturbations from uneven flow and varying pressure in the pressure manifold that supplies air to the thrusters, which caused a couple on the simulator due to firing forces that do not balance. Further investigation into characterizing the viscous friction due to air gap between the simulator and the epoxy floor, and the dead zone of the thrusters needs to be done to improve the performance of the controller.
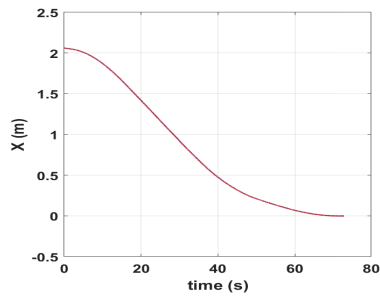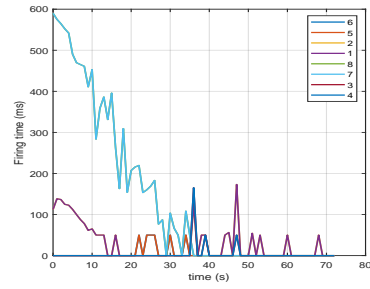
(a) x position (m) vs. time (s).
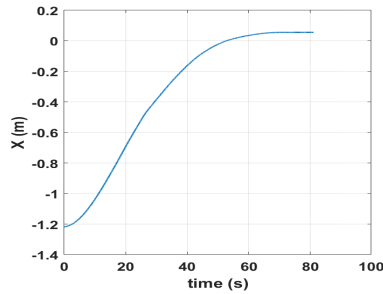
(b) y position (m) vs. time (s).
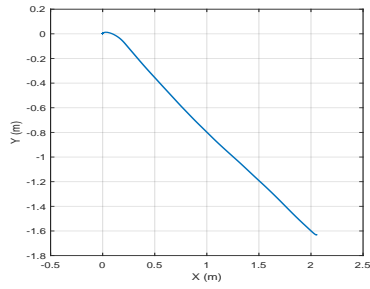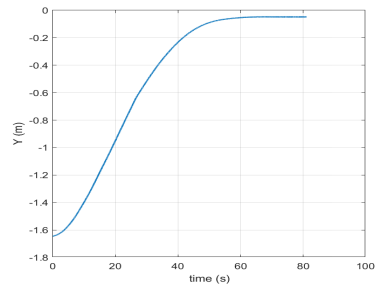
(c) (x,y) trajectory (m).
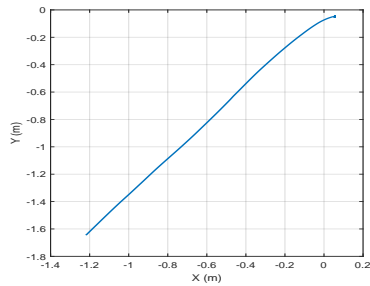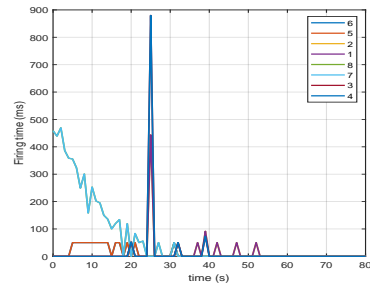
(d) Thruster firing time vs. time.

Figure 12: Closed-loop waypoint reaching experimental result- test case 1.

(a) x position (m) vs. time (s).

(b) y position (m) vs. time (s).

(c) (x,y) trajectory in m.

(d) Thruster firing time vs. time.

Figure 13: Closed-loop waypoint reaching experimental result- test case 2.

CONCLUSION

In this paper, we discussed the hardware development of a 6-DOF robotic spacecraft simulator M-STAR for testing formation guidance, navigation and control algorithms. The simulator has 6-DOF with translation and attitude stages decoupled using a spherical air bearing. The translation stage floats on the epoxy flat floor using three flat round air bearings. The hardware architecture of M-STAR and its subsystems including mechanical structure, pneumatic system for flat air bearings, spherical air bearing required to achieve frictionless and disturbance torque free motion of the simulator were discussed in detail. The low level control architecture for thrusters and reaction wheels was mentioned for controlling the dynamics.

A nonlinear dynamic model of M-STAR was presented by modelling the system as a 3D pendulum on a floating platform. A hierarchical model-based control law for the nonlinear system was discussed for tracking a given position and attitude trajectory. A generalized pseudo-inverse control allocation scheme, with a thruster actuator model developed using experiments, was used to implement the control law in a ROS based software framework for testing position control. Future work will focus on multi-agent guidance and control experiments exercising all five M-STARs.

ACKNOWLEDGEMENT

REFERENCES

[1] H. Levine, S. Shaklan, and J. Kasting, "Terrestrial Planet Finder Coronagraph Science and Technology Definition Team (STDT) Report," *JPL document D-34923 (Pasadena: JPL)*, 2006.

[2] R. J. Barton, "MIMO Architectures for Efficient Communication in Space," *arXiv preprint arXiv:1601.03664*, 2016.

[3] S. Bandyopadhyay, R. Foust, G. P. Subramanian, S.-J. Chung, and F. Y. Hadaegh, "Review of formation flying and constellation missions using nanosatellites," *Journal of Spacecraft and Rockets*, Vol. 53, No. 3, 2016, pp. 567–578.

[4] S. Mandutianu, F. Hadaegh, and P. Elliot, "Multi-agent system for formation flying missions," *Aerospace Conference, 2001, IEEE Proceedings.*, Vol. 6, IEEE, 2001, pp. 2793–2802.

[5] S.-J. Chung, A. A. Paranjape, P. Dames, S. Shen, and V. Kumar, "A Survey on Aerial Swarm Robotics," *IEEE Transactions on Robotics*, Vol. 34, No. 4, 2018, pp. 837–855.

[6] J. L. Schwartz, M. A. Peck, and C. D. Hall, "Historical Review of Air-Bearing Spacecraft Simulators," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 4, 2003, pp. 513–522, 10.2514/2.5085.

[7] Y. Eun, C. Park, and S.-Y. Park, "Design and Development of Ground-Based 5-DOF Spacecraft Formation Flying Testbed," *AIAA Modeling and Simulation Technologies Conference*, January 2016, pp. 1–7, 10.2514/6.2016-1668.

[8] J. L. Schwartz, *The distributed spacecraft attitude control system simulator: from design concept to decentralized control*. PhD thesis, Virginia Tech, 2004.

[9] D. W. Miller, A. Saenz-Otero, J. Wertz, A. Chen, G. Berkowski, C. Brodel, S. Carlson, D. Carpenter, S. Chen, S. Cheng, *et al.*, "SPHERES: a testbed for long duration satellite formation flying in microgravity conditions," *Proceedings of the AAS/AIAA space flight mechanics meeting*, Clearwater, Florida, January, 2000, pp. 167–179.

[10] S.-J. Chung, D. W. Miller, and O. L. d. Weck, "ARGOS testbed: study of multidisciplinary challenges of future spaceborne interferometric arrays," *Optical Engineering*, Vol. 43, No. 9, 2004, pp. 2156–2168.

[11] B. N. Agrawal and R. E. Rasmussen, "Air-bearing-based satellite attitude dynamics simulator for control software research and development," Vol. 4366, No. 831, 2001, pp. 204–214, 10.1117/12.438072.

[12] D. Gallardo and R. Bevilacqua, "Six Degrees of Freedom Experimental Platform for Testing Autonomous Satellites Operations," *Proceedings of the 8th International Conference on Guidance, Navigation and Control*, 2011.

[13] K. Saulnier, D. Pérez, R. C. Huang, D. Gallardo, G. Tilton, and R. Bevilacqua, "A six-degree-of-freedom hardware-in-the-loop simulator for small spacecraft," *Acta Astronautica*, Vol. 105, No. 2, 2014, pp. 444–462, 10.1016/j.actaastro.2014.10.027.

[14] D. P. Scharf, J. A. Keim, and F. Y. Hadaegh, "Flight-like ground demonstrations of precision maneuvers for spacecraft formations - Part II," *IEEE Systems Journal*, Vol. 4, No. 1, 2010, pp. 96–106, 10.1109/JSYST.2010.2044281.

[15] M. W. Regehr, A. B. Acikmese, A. Ahmed, M. Aung, K. Clark, P. MacNeal, J. Shields, G. Singh, R. Bailey, C. Bushnell, *et al.*, "The formation control testbed," *Aerospace Conference, 2004. Proceedings. 2004 IEEE*, Vol. 1, IEEE, 2004, pp. 557–564.

[16] R. Zappulla, J. Virgili-Llop, C. Zagaris, H. Park, and M. Romano, "Dynamic air-bearing hardware-in-the-loop testbed to experimentally evaluate autonomous spacecraft proximity maneuvers," *Journal of Spacecraft and Rockets*, Vol. 54, No. 4, 2017, pp. 825–839.

[17] P. Tsiotras, "ASTROS: A 5DOF experimental facility for research in space proximity operations," *Advances in the Astronautical Sciences*, Vol. 151, 01 2014, pp. 717–730.

[18] D. Gallardo, R. Bevilacqua, and R. Rasmussen, "Advances on a 6 degrees of freedom testbed for autonomous satellites operations," *AIAA Guidance, Navigation, and Control Conference*, 2011, p. 6591.

[19] M. Wilde, B. Kaplinger, T. Go, H. Gutierrez, and D. Kirk, "ORION: A simulation environment for spacecraft formation flight, capture, and orbital robotics," *Aerospace Conference, 2016 IEEE*, IEEE, 2016, pp. 1–14.

[20] D. C. Sternberg, C. Pong, N. Filipe, S. Mohan, S. Johnson, and L. Jones-Wilson, "Jet Propulsion Laboratory Small Satellite Dynamics Testbed Simulation: On-Orbit Performance Model Validation," *Journal of Spacecraft and Rockets*, Vol. 55, No. 2, 2017, pp. 322–334.

[21] S. P. Viswanathan, A. Sanyal, and L. Holguin, "Dynamics and control of a six degrees of freedom ground simulator for autonomous rendezvous and proximity operation of spacecraft," *AIAA Guidance, Navigation, and Control Conference*, 2012, p. 4926.

[22] M. Schlotterer, E. Edlerman, F. Fumenti, P. Gurfil, S. Theil, and H. Zhang, "On-Ground Testing of Autonomous Guidance for Multiple Satellites in a Cluster," *Proceedings of the 8th International Workshop on Satellite Constellations and Formation Flying*, 06 2015.

[23] S.-J. Chung and D. W. Miller, "Propellant-free control of tethered formation flight, part 1: Linear control and experimentation," *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 3, 2008, pp. 571–584.

[24] M. Ciarcia, R. Cristi, and M. Romano, "Experimental Emulation of the Scaled Clohessy-Wiltshire Dynamics on a Flat Air-Bearing Testbed," *AIAA Guidance, Navigation, and Control Conference*, 2017, p. 1047.

[25] T. A. Johansen and T. I. Fossen, "Control allocation—a survey," *Automatica*, Vol. 49, No. 5, 2013, pp. 1087–1103.

[26] M. Bodson, "Evaluation of optimization methods for control allocation," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 4, 2002, pp. 703–711.

[27] A. Chen, "Propulsion system characterization for the spheres formation flight and docking testbed. Master's thesis," 2002.

[28] F. L. Markley and J. L. Crassidis, *Fundamentals of spacecraft attitude determination and control*, Vol. 33. Springer, 2014.

[29] H. K. Khalil, "Noninear Systems," *Prentice-Hall, New Jersey*, Vol. 2, No. 5, 1996, pp. 5–1.

[30] S.-J. Chung, S. Bandyopadhyay, I. Chang, and F. Y. Hadaegh, "Phase synchronization control of complex networks of Lagrangian systems on adaptive digraphs," *Automatica*, Vol. 49, No. 5, 2013, pp. 1148–1161.

[31] W. Lohmiller and J.-J. E. Slotine, "On contraction analysis for non-linear systems," *Automatica*, Vol. 34, No. 6, 1998, pp. 683–696.

[32] S. Bandyopadhyay, S.-J. Chung, and F. Y. Hadaegh, "Nonlinear attitude control of spacecraft with a large captured object," *Journal of Guidance, Control, and Dynamics*, Vol. 39, No. 4, 2016, pp. 754–769.

[33] F. L. Markley, R. G. Reynolds, F. X. Liu, and K. L. Lebsock, "Maximum torque and momentum envelopes for reaction wheel arrays," *Journal of Guidance, Control, and Dynamics*, Vol. 33, No. 5, 2010, pp. 1606–1614.

[34] R. Cowen, "The Wheels Come Off Kepler," *Nature*, Vol. 497, No. 7450, 2013, pp. 417–418.