

# Cooperative Object Transport in 3D with Multiple Quadrotors using No Peer Communication

Zijian Wang<sup>1</sup>, Sumeet Singh<sup>1</sup>, Marco Pavone<sup>1</sup> and Mac Schwager<sup>1</sup>

**Abstract**—We present a framework to enable a fleet of rigidly attached quadrotor aerial robots to transport heavy objects along a known reference trajectory without inter-robot communication or centralized coordination. Leveraging a distributed wrench controller, we provide exponential stability guarantees for the entire assembly, under a mild geometric condition. This is achieved by each quadrotor independently solving a local optimization problem to counteract the biased torque effects from each robot in the assembly. We rigorously analyze the controllability of the object, design a distributed compensation scheme to address these challenges, and show that the resulting strategy collectively guarantees full group control authority. To ensure feasibility for online implementation, we derive bounds on the net desired control wrench, characterize the output wrench space of each quadrotor, and perform subsequent trajectory optimization under these input constraints. We thoroughly validate our method in simulation with eight quadrotors transporting a heavy object in a cluttered environment subject to various sources of uncertainty, and demonstrate the algorithm’s resilience.

## I. INTRODUCTION

In this paper, we present a distributed controller that allows a group of rigidly-attached quadrotor aerial robots to cooperatively transport heavy objects in 3D. Distinct from existing cooperative aerial manipulation literature, our approach addresses the challenging problem where no peer communication is allowed among the robots. The only available information to each individual robot are the inertial properties of the object, its own attachment point on the object, and a reference trajectory that is broadcast to all robots. Notably, the robots do not know the locations, nor the actions taken by other robots. Instead, each quadrotor locally solves an independent optimization problem at each time-step, the collective result of which guarantees the desired group behavior. By eliminating the communication bottleneck, which has been shown to be noisy, vulnerable, complicated and non-scalable in large swarm systems [1], our method is suitable for a broad range of applications that require fast response, quick setup, and frequent reconfiguration. For example, in a disaster relief scenario, our approach can be used as a modular system to deliver equipment of various sizes, by utilizing up to tens or hundreds of drones at a time. In the civil sector, packages can be delivered in

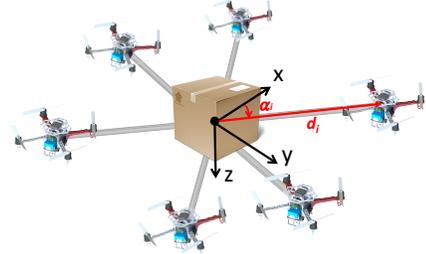


Fig. 1. An example of object transport in 3D with six rigidly attached quadrotors. The quadrotors do not communicate with each other, thus allowing for fast reconfiguration for objects of different sizes. This is achieved by each quadrotor independently computing their control action onboard based on a reference trajectory that is broadcast to them.

the most efficient and economical way by matching the size of the package with the required number of robots.

Our controller is based upon the SE(3) geometric controller and differential flatness theory [2], [3], [4], which are powerful tools for controlling a single quadrotor. In our method, each quadrotor takes equal responsibility for the desired nominal wrench for the object with respect to its center of mass, computed independently by each quadrotor. This nominal wrench is usually not feasible for a single quadrotor due to its inherent biased torque controllability. Through a decomposition into *unbiased axes* and *biased axis* (see Figure 3 for an illustration), we show that three components of the 4D nominal wrench are feasible for a single quadrotor. A local optimization is then solved by each quadrotor to best realize the desired moment along the *biased axis* while still adhering to the three feasible components of the nominal wrench along the *unbiased axes*.

Under a mild centro-symmetric condition (Assumption 1), we show that the proposed control strategy is exponentially stable and is tolerant of non-centro-symmetric robot configurations as well. We perform thorough analysis of the feasibility of the controller, where we derive explicit bounds on the required thrust and moments and characterize each quadrotor’s wrench output space. Finally, we leverage bi-level constrained trajectory optimization to compute snap- and time-optimal paths that satisfy the computed control bounds and solve the problem using an exterior point method and iterative coordinate descent.

Our work is related to a number of cooperative object transport methods for 2D planar motion that also do not require explicit inter-robot communication [5], [6], [7], [8], [9], and [10] where a decentralized adaptive control scheme is developed to allow multiple robots to estimate unknown parameters online. Our solution to the 3D case greatly

\*This work was supported in part by NASA under the Early State Innovations Program (Grant NNX16AD19G), the King Abdulaziz City for Science and Technology (KACST), the Toyota-SAIL Center for AI Research, ONR (Grant N00014-16-1-2787), and NSF (Grant IIS-1646921). We are grateful for the support.

<sup>1</sup>Department of Aeronautics and Astronautics, Stanford University, Stanford, CA 94305, USA, {zjwang, ssingh19, pavone, schwager}@stanford.edu

broadens the allowable workspace. In terms of cooperative aerial manipulation, a centralized control allocation approach is presented in [11] for rigidly attached quadrotors. A tele-manipulation framework is proposed in [12] by translating hand motion into quadrotor formation and interaction force control. Other researchers have considered using cables to suspend the payload by multiple aerial robots [13], [14], [15]. However, in many applications it is impractical to connect a large number of cables to a payload. In package delivery or autonomous construction applications where significant aerial traffic is expected, entangled cables and collisions between swinging payloads becomes a concern. In addition, the unilateral nature of cable tensions introduces hybrid dynamics [13] that renders stability analysis challenging, especially for the multi-robot case. Alternatively, one may use multidirectional thrusters [16] for full 6D pose control. However, for lifting heavy objects where the primary hurdle is gravity, lateral thrusters are an inefficient design choice. In [17] and [18], the quadrotors are augmented with a 2-DOF robotic arm and the problems are addressed from the perspective of path planning and decentralized flatness-based control. A formation-based cooperative manipulation approach is presented in [19]. Finally, our work is also inspired by trajectory generation methods for quadrotors in [20], [21]. However, we additionally incorporate closed-loop control constraints and tracking stability into the design.

The contributions of this paper are threefold. First, we propose a decentralized wrench controller for cooperative aerial manipulation without peer communication (Section III). Under a mild centro-symmetric condition (Assumption 1), we show that the net assembly is exponentially stable in position and attitude tracking, based on pairwise controllability analysis. Second, in Section IV, we conduct a feasibility analysis for the online execution of the control algorithm by computing bounds on the tracking error and control effort, and characterize each quadrotor's control space. Third, we present a differential flatness-inspired trajectory optimization algorithm (Section V) that additionally incorporates the bounds from feasibility analysis as constraints, yielding the open-loop reference inputs. Simulation results are presented in Section VI that successfully validate the proposed approach.

## II. PROBLEM FORMULATION

We use a group of quadrotors to collectively manipulate a heavy object, which has mass  $m$  and inertia tensor  $J$ . The motion of the object in 3D space is governed by the Newton-Euler equations. Denote the 12-dimensional state variable as  $\xi = (x, y, z, v_x, v_y, v_z, \phi, \theta, \psi, p, q, r)$ , corresponding to the 3D inertial position  $\mathbf{h} := (x, y, z)$ , linear velocities  $\mathbf{v} := (v_x, v_y, v_z)$ , Euler angles  $(\phi, \theta, \psi)$  and (body-frame) angular velocities  $\boldsymbol{\omega} := (p, q, r)$ . We adopt the “z down” body frame convention as shown in Figure 2, and the  $ZYX$  Euler angle rotation sequence.

Consider a fleet of  $N$  quadrotors, each rigidly attached to the object with their body z-axis aligned with that of the object. We assume that each individual quadrotor does not have

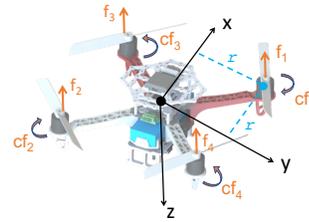


Fig. 2. The configuration and axes definition of a single quadrotor. Without loss of generality, we follow the “x” quadrotor convention in this paper, though all the results are also applicable to other quadrotor configurations.

sufficient power to lift the object. Let  $\mathbf{f}^i = [f_1^i, f_2^i, f_3^i, f_4^i]^T$  denote the thrust forces corresponding to quadrotor  $i$ 's four propellers, subject to the limits

$$0 \leq f_j^i \leq f_{\max}, \quad i \in \{1, \dots, N\}, \quad j \in \{1, 2, 3, 4\}.$$

Each quadrotor can generate a net thrust and three independent moments and will contribute a fraction of the total required wrench. The net resultant wrench due to quadrotor  $i$  expressed in its own body aligned frame is given by

$$\mathbf{w}_b^i := \begin{pmatrix} 1 & 1 & 1 & 1 \\ -r & r & r & -r \\ r & -r & r & -r \\ c & c & -c & -c \end{pmatrix} \begin{pmatrix} f_1^i \\ f_2^i \\ f_3^i \\ f_4^i \end{pmatrix}, \quad (1)$$

where  $r$  is the moment arm length of each motor with respect to the quadrotor center of mass (see Figure 2), and  $c$  is a constant coefficient for the induced torque of the motor. The quadrotors are assumed to be attached to the x-y plane of the object with distance  $d_i$  and angle  $\alpha_i \in [-\pi, \pi]$  measured with respect to the object x-axis, as shown in Figure 1. The wrench imparted by quadrotor  $i$  to the object is given by

$$\mathbf{w}_{\text{obj}}^i := \begin{pmatrix} f_z^i \\ \boldsymbol{\tau}^i \end{pmatrix} = \begin{pmatrix} f_z^i \\ \tau_x^i \\ \tau_y^i \\ \tau_z^i \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -d_i \sin \alpha_i & 1 & 0 & 0 \\ d_i \cos \alpha_i & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \mathbf{w}_b^i, \quad (2)$$

where without loss of generality, we assume that all quadrotor frames are aligned with the object's frame. Under the combined inputs from all quadrotors,  $\mathbf{w}_{\text{obj}} := \sum_{i=1}^N \mathbf{w}_{\text{obj}}^i := (f_z, \boldsymbol{\tau})$ , the equations of motion of the object are

$$\dot{\mathbf{v}} = g\mathbf{e}_3 - \frac{1}{m}Rf_z\mathbf{e}_3, \quad (3)$$

$$\dot{\mathbf{h}} = \mathbf{v}, \quad (4)$$

$$\dot{\boldsymbol{\omega}} = J^{-1}\boldsymbol{\tau} - J^{-1}\hat{\boldsymbol{\omega}}J\boldsymbol{\omega}, \quad (5)$$

$$\dot{R} = R\hat{\boldsymbol{\omega}}, \quad (6)$$

where  $R$  is the body-to-inertial rotation matrix,  $g$  is the gravitational acceleration,  $\mathbf{e}_3 := [0, 0, 1]^T$ , and  $(\cdot) : \mathbb{R}^3 \rightarrow \mathfrak{so}_3$  is the hat map transporting vectors in  $\mathbb{R}^3$  to the  $\text{SO}(3)$  Lie algebra,  $\mathfrak{so}_3$ .

In order to transport the object to the destination, we assume that a smooth reference trajectory (continuously differentiable in time up to fourth order) is broadcast to all quadrotors. However, no peer communication is available between any two quadrotors. We also assume that each quadrotor knows the net mass  $m$ , inertia  $J$ , and number of

quadrotors  $N$ , as well as its *own* attachment point on the object, i.e., the value of  $d_i$  and  $\alpha_i$ . It does not, however, know the locations of other quadrotors. Finally, we assume each quadrotor can measure the position, orientation, linear, and angular velocity of the object using onboard sensors.

### III. DISTRIBUTED WRENCH CONTROL

Since all quadrotors have access to the reference trajectory and real-time state of the object, they can independently compute the total wrench required to track the trajectory. The combined payload and quadrotors assembly is a rigid body whose dynamics resemble those of a single quadrotor; hence we will leverage the SE(3) controller first proposed in [2], [3] to compute the net object wrench. However, input constraints prohibit one individual quadrotor from exerting the required total wrench. In this section, we propose a distributed controller that allows each quadrotor to independently compute its control inputs, without peer communication. Collectively, this local strategy results in a provably stable group behavior that guarantees successful tracking. We first briefly review the SE(3) controller.

Let  $\sigma : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^3$  denote the reference position trajectory, continuously differentiable up to 4<sup>th</sup> order and  $\sigma_\psi : \mathbb{R}_{\geq 0} \rightarrow S^1$  the reference yaw trajectory, continuously differentiable up to second order. Given the current state of the object, the net desired thrust  $f_z$  and torque  $\tau$  are given by:

$$f_z = - \overbrace{(-k_p e_p - k_v e_v - m g \mathbf{e}_3 + m \ddot{\sigma})}^{:= \mathbf{F}_{\text{des}}} \cdot \mathbf{R} \mathbf{e}_3, \quad (7)$$

$$\tau = -k_R e_R - k_\omega e_\omega + \dot{\omega} J \omega + J \left( -\dot{\omega} R^T R_{\text{des}} \omega_{\text{des}} + R^T R_{\text{des}} \dot{\omega}_{\text{des}} \right), \quad (8)$$

where  $e_p := \mathbf{h} - \sigma$ ,  $e_v := \mathbf{v} - \dot{\sigma}$ ,

$$e_R := \frac{1}{2} (R_{\text{des}}^T R - R^T R_{\text{des}})^v, \quad (9)$$

$$e_\omega := \omega - R^T R_{\text{des}} \omega_{\text{des}}, \quad (10)$$

$(\cdot)^v : \mathfrak{so}_3 \rightarrow \mathbb{R}^3$  is the inverse hat map, and  $k_p, k_v, k_R, k_\omega$  are positive constant gains. The desired rotation matrix  $R_{\text{des}}$  is defined by the desired z-axis  $\mathbf{z}_b := -\mathbf{F}_{\text{des}} / \|\mathbf{F}_{\text{des}}\|$ , and yaw angle  $\sigma_\psi$ . The desired angular velocity  $\omega_{\text{des}}$  and acceleration  $\dot{\omega}_{\text{des}}$  are defined by the time-derivatives of  $\mathbf{z}_b$  (thereby incorporating acceleration and jerk feedback) and  $\sigma_\psi$ ; refer to [3] for a derivation of these quantities. For simplicity, similar to [22], we compute  $\dot{\sigma}_\psi$  (and by integration,  $\sigma_\psi$ ) online by constraining  $\omega_{\text{des}_z} = 0$ .

#### A. Wrench Allocation

To achieve the desired net wrench in (7) and (8), one needs to assign motor thrusts to each quadrotor – a challenging problem for two reasons: (1) a quadrotor does not know the positions of other quadrotors, and cannot communicate with them, and (2) each quadrotor's applied wrench is significantly biased about one axis due to its off-center attachment point (see Figure 3 for an illustration of this observation). To address the first challenge, we assume that each quadrotor takes on equal responsibility for the net thrust  $f_z$  and torque  $\tau$ ; that is, the wrench command to the  $i^{\text{th}}$  quadrotor expressed in the object's frame is given by:  $(f_z/N, \tau/N)$ , with each

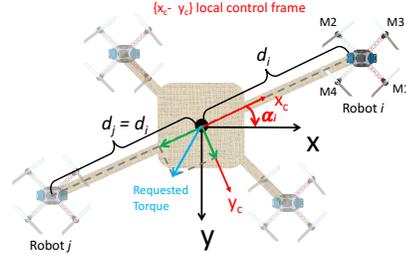


Fig. 3. Illustration of the *control* frame  $\{x_c, y_c\}$  for quadrotor  $i$ . For a given requested torque generated by the SE(3) controller in the object's frame, we can express it in the *control* frame and decompose it into  $x_c$  and  $y_c$  axis. Along the *unbiased*  $x_c$  axis, robot  $i$  can exert both positive and negative torque. However, it usually cannot apply negative torque along the *biased*  $y_c$  axis due to the large moment arm created by  $d_i$ .

robot computing eqs. (7) and (8) independently. Second, we introduce the following mild assumption regarding the arrangement of the quadrotors on the object:

**Assumption 1** (Centro-symmetry). *The robots attachment points are centro-symmetric around the center of mass of the object, meaning that for any robot  $i$ , there exists another robot  $j \neq i$ , such that  $\alpha_j = \alpha_i - \pi$  and  $d_i = d_j$ .*

In practice, although it might be hard to strictly satisfy this assumption, the robots are likely to evenly spread out as the number of the robots increases [1] such that the assumption is approximately true. In addition, the symmetric configuration is an intuitive way for a user to attach the robots to a payload. Centro-symmetry is required for our analysis, but in practice our controller still works well if the assumption is violated, as explored in simulation in Section VI.

While the equal wrench assignment is generally non-optimal for a given attachment configuration, we stress that such a design choice stems from the constraint that no peer communication is allowed and the limited knowledge each quadrotor has regarding the attachment geometry. In future work, we plan to investigate distributed *adaptive* strategies in which each quadrotor estimates the configuration geometry and appropriately adjusts its own wrench assignment.

Given the pairwise centro-symmetry assumption, it will be useful to introduce a local reference frame for each quadrotor, hereby referred to as the *control* frame, defined by simply rotating the object reference frame around the z-axis by angle  $\alpha_i$ ; see Figure 3 for an illustration. Then, the *commanded* wrench for quadrotor  $i$  in its control frame may be expressed using the following rotation:

$${}^c R := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha_i & \sin \alpha_i & 0 \\ 0 & -\sin \alpha_i & \cos \alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (11)$$

$$\mathbf{w}^i := \begin{bmatrix} f_{c_z}^i \\ \tau_c^i \end{bmatrix} = {}^c R \begin{bmatrix} f_z/N \\ \tau/N \end{bmatrix}. \quad (12)$$

By centro-symmetry,  $\tau_{c_x}^j = -\tau_{c_x}^i$  and  $\tau_{c_y}^j = -\tau_{c_y}^i$ , while  $\tau_{c_z}^j = \tau_{c_z}^i = \tau_z/N$  and  $f_{c_z}^i = f_{c_z}^j = f_z/N$ . We also denote the actual wrench achieved by quadrotor  $i$  in the control frame by  $\mathbf{w}^i = {}^c R \mathbf{w}_{\text{obj}}^i := W_c^i [f_1^i, f_2^i, f_3^i, f_4^i]^T$ .

From (1), (2), and (11),  $W_c^i$  is given by

$$\begin{bmatrix} 1 & 1 & & \\ -rC\alpha_i + rS\alpha_i & rC\alpha_i - rS\alpha_i & & \\ d_i + rC\alpha_i + rS\alpha_i & d_i - rC\alpha_i - rS\alpha_i & \cdots & \\ c & c & & \\ & 1 & & \\ rC\alpha_i + rS\alpha_i & -rC\alpha_i - rS\alpha_i & & \\ d_i + rC\alpha_i - rS\alpha_i & d_i - rC\alpha_i + rS\alpha_i & & \\ -c & -c & & \end{bmatrix}, \quad (13)$$

where  $S$  and  $C$  denote sin and cos respectively. By expressing the desired and actual wrench in the control frame, one can isolate each quadrotor's biased torque controllability to the control frame  $y$ -axis. In particular, observe that the third row of  $W_c^i$  is biased by a constant amount  $d_i$ , which prohibits exerting negative torque along the control frame  $y$ -axis since  $d_i$  is usually much larger than  $r$ . Therefore, naively solving for motor thrusts  $\mathbf{f}^i$  by equating (12) and  $\mathbf{w}^i$  could lead to infeasibility. In order to ensure that the fleet collectively achieves the desired SE(3) wrench, we further analyze this controllability in the next section.

### B. Pairwise Controllability

As each quadrotor possesses the ability to generate the desired thrust and both positive and negative torques along its control  $x$ - and  $z$ -axes, consider the local optimization problem:

$$\begin{aligned} \min_{0 \leq \mathbf{f}^i \leq \mathbf{f}_{\max}} & \quad |\mathbf{w}^i(3) - \tau_{c_y}^i| \\ \text{subject to} & \quad \mathbf{w}^i(1) = f_{c_z}^i, \\ & \quad \mathbf{w}^i(2) = \tau_{c_x}^i, \\ & \quad \mathbf{w}^i(4) = \tau_{c_z}^i. \end{aligned} \quad (14)$$

The objective function tries to find motor thrusts  $\mathbf{f}^i$  that minimize the difference between the desired and actual  $y$ -axis wrench, subject to the desired wrench constraints along the other axes (thrust, and  $x$ - and  $z$ -axes torques).

Consider now, problem (14) for quadrotor  $j$  in the centrosymmetric pair  $(i, j)$ . Thus  $d_j = d_i$ , and  $\alpha_j = \alpha_i - \pi$ . For quadrotor  $j$ ,  $W_c^j$  has identical first and fourth rows as  $W_c^i$  as well as identical thrust and  $z$ -axes torque commands (i.e.,  $\tau_{c_x}^j = \tau_{c_x}^i$ , and  $f_{c_z}^i = f_{c_z}^j$ ). The second constraint in (14) for quadrotor  $i$  reads as

$$\tau_{c_x}^i = r(C\alpha_i - S\alpha_i)(f_2^i - f_1^i) + r(C\alpha_i + S\alpha_i)(f_3^i - f_4^i),$$

and for quadrotor  $j$ :

$$\tau_{c_x}^j = r(C\alpha_i - S\alpha_i)(f_1^j - f_2^j) + r(C\alpha_i + S\alpha_i)(f_4^j - f_3^j),$$

since  $\tau_{c_x}^j = -\tau_{c_x}^i$  and  $\alpha_j = \alpha_i - \pi$ . The two equations above are equivalent, indicating that robot  $i$  and  $j$  have the same set of constraints when solving (14) in their respective control frames. For the objective, notice that

$$\begin{aligned} \mathbf{w}^i(3) &= d_i(f_1^i + f_2^i + f_3^i + f_4^i) + r(C\alpha_i + S\alpha_i)(f_1^i - f_2^i) \\ &\quad + r(C\alpha_i - S\alpha_i)(f_3^i - f_4^i), \end{aligned} \quad (15)$$

$$\begin{aligned} \mathbf{w}^j(3) &= d_i(f_1^j + f_2^j + f_3^j + f_4^j) + r(C\alpha_i + S\alpha_i)(f_2^j - f_1^j) \\ &\quad + r(C\alpha_i - S\alpha_i)(f_4^j - f_3^j). \end{aligned} \quad (16)$$

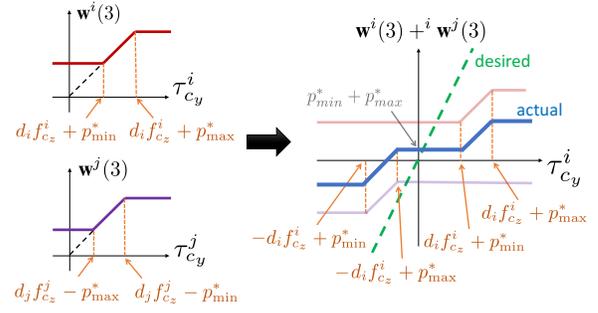


Fig. 4. A visualization of (19) (upper-left), (20) (lower-left), and the combined output torque profile (21) (right). The combined output torque from the  $(i, j)$  pair, shown in blue, has a non-zero deadband around the  $y$  axis. Note how  $\mathbf{w}^j(3)$  gets negated and reflected after being transformed into  $i$ 's frame. Additionally, the desired torque, as shown in green, has a  $+2$  slope since the desired output from the pair is  $2\tau_{c_y}^i$  given the command  $\tau_{c_y}^i$  for quad  $i$  in the pair  $(i, j)$ . The slope on the actual output curve, shown in blue, has  $+1$  slope on the two non-flat sections according to (21).

Define the last two terms in (15) and (16) as

$$g(\mathbf{f}^i) = r(C\alpha_i + S\alpha_i)(f_1^i - f_2^i) + r(C\alpha_i - S\alpha_i)(f_3^i - f_4^i),$$

$$g(\mathbf{f}^j) = r(C\alpha_i + S\alpha_i)(f_1^j - f_2^j) + r(C\alpha_i - S\alpha_i)(f_3^j - f_4^j).$$

Due to the identical constraints,  $g(\mathbf{f}^i)$  and  $g(\mathbf{f}^j)$  must have the same minimal and maximal value, denoted as

$$p_{\min}^* = \min g(\mathbf{f}^i) = \min g(\mathbf{f}^j), \quad (17)$$

$$p_{\max}^* = \max g(\mathbf{f}^i) = \max g(\mathbf{f}^j), \quad (18)$$

subject to the constraints in (14). Then according to (15), (16), (17), (18) and provided the feasibility set of (14) is non-empty, the optimal  $\mathbf{w}^i(3)$  and  $\mathbf{w}^j(3)$  for problem (14) are

$$\mathbf{w}^i(3)^* = \begin{cases} d_i f_{c_z}^i + p_{\min}^* & \text{if } \tau_{c_y}^i \leq d_i f_{c_z}^i + p_{\min}^* \\ \tau_{c_y}^i & \text{if } p_{\min}^* < \tau_{c_y}^i - d_i f_{c_z}^i \leq p_{\max}^* \\ d_i f_{c_z}^i + p_{\max}^* & \text{else.} \end{cases} \quad (19)$$

$$\mathbf{w}^j(3)^* = \begin{cases} d_j f_{c_z}^j - p_{\max}^* & \text{if } \tau_{c_y}^j \leq d_j f_{c_z}^j - p_{\max}^* \\ \tau_{c_y}^j & \text{if } -p_{\max}^* < \tau_{c_y}^j - d_j f_{c_z}^j \leq -p_{\min}^* \\ d_j f_{c_z}^j - p_{\min}^* & \text{else.} \end{cases} \quad (20)$$

These essentially describe two biased saturated curves, as shown in Figure 4.

To characterize the combined  $y$ -axis torque output of the pair  $(i, j)$  under the strategy (14), we transform  $\mathbf{w}^j(3)$ , which is in  $j$ 's local frame, into  $i$ 's frame by reflecting and negating the curve for  $\mathbf{w}^j(3)$ . Then the total  $y$ -axis torque of pair  $(i, j)$ , expressed in  $i$ 's control frame is

$$\begin{aligned} \mathbf{w}^i(3)^* + {}^i\mathbf{w}^j(3)^* &= \\ &\begin{cases} 2p_{\min}^* & \text{if } \tau_{c_y}^i \leq -d_i f_{c_z}^i + p_{\min}^*, \\ \tau_{c_y}^i + d_i f_{c_z}^i + p_{\min}^* & \text{if } +p_{\min}^* < \tau_{c_y}^i + d_i f_{c_z}^i < +p_{\max}^*, \\ p_{\min}^* + p_{\max}^* & \text{if } -d_i f_{c_z}^i + p_{\max}^* \leq \tau_{c_y}^i \leq d_i f_{c_z}^i + p_{\min}^*, \\ \tau_{c_y}^i - d_i f_{c_z}^i + p_{\max}^* & \text{if } d_i f_{c_z}^i + p_{\min}^* < \tau_{c_y}^i < d_i f_{c_z}^i + p_{\max}^*, \\ 2p_{\max}^* & \text{if } \tau_{c_y}^i \geq d_i f_{c_z}^i + p_{\max}^*. \end{cases} \end{aligned} \quad (21)$$

which is a piecewise linear function with respect to the requested wrench from the SE(3) controller, as visualized in

Figure 4. Given these response curves, we present a pairwise compensation technique to address the bias and deadband characteristics of (21).

### C. Pairwise Torque Compensation

The output torque profile of (14) plotted in Figure 4 makes control challenging and stability analysis difficult. In this section, however, we show that under Assumption 1 the compensation can be done without communication such that the actual y-axis combined torque output of the  $(i, j)$  pair exactly replicates the desired SE(3) torque, as shown in the green dashed line in Figure 4. Observe from (21) and Figure 4 that when

$$p_{\min}^* + p_{\max}^* = 0, \quad (22)$$

the torque output profile becomes a symmetric deadband curve centered at the origin. Consequently, the *capable* quadrotor (defined as the quadrotor with positive requested y-axis torque in a given symmetric pair) can exert additional torque (beyond its original local command) to compensate for the offset from its complement in the symmetric pair. Mathematically, this process requires each quadrotor to solve two optimization problems. First, find  $p_{\min}^*$  and  $p_{\max}^*$  by solving (17) and (18) under the constraints in (14). Denote

$$p^* = \min\{|p_{\min}^*|, |p_{\max}^*|\}, \quad (23)$$

and choose  $p_{\min}^{\prime} = -p^*$  and  $p_{\max}^{\prime} = p^*$ , thereby allowing the pair to satisfy condition (22). This means that both quadrotors  $i$  and  $j$  will choose their y-axis wrench within  $[d_i f_{c_z}^i - p^*, d_i f_{c_z}^i + p^*]$ , which we know is feasible since it is a subset of original y-axis torque range as a result of (23). Second, compute thruster forces using:

**Problem 1.** (*Distributed Wrench Controller*) *During the cooperative aerial manipulation task, each quadrotor's motor thrusts are given by the solution of*

$$\min_{\mathbf{f}^i} |\mathbf{w}^i(3) - \tau_{c_y}^{i'}| \quad (24)$$

subject to Constraints in (14),

$$d_i f_{c_z}^i - p^* \leq \mathbf{w}^i(3) \leq d_i f_{c_z}^i + p^*,$$

where

$$\tau_{c_y}^{i'} = \begin{cases} 2\tau_{c_y}^i + d_i f_{c_z}^i - p^* & \text{if } \tau_{c_y}^i \geq 0, \\ \tau_{c_y}^i & \text{if } \tau_{c_y}^i < 0. \end{cases} \quad (25)$$

In (25),  $\tau_{c_y}^{i'}$  is the adjusted torque along the local y-axis. Notice that the capable quadrotor compensates for the deadband *and* the offset torque created by the “incapable” quadrotor (i.e., quadrotor  $j$  in this notation); see Figure 4 for the  $(i, j)$  pair. Finally, note that all the computation here requires only local information so that the compensation can be done without communication.

### D. Closed-Loop Stability

Given the pairwise compensation strategy presented in the preceding discussion, closed-loop stability is now a straightforward conclusion of the following proposition. For

simplicity, assume all diagonal gain matrices are equal and given by  $k_p, k_v, k_R, k_\omega$ .

**Proposition 1** (Closed-Loop Stability). (i) *Define the (positive-definite) attitude error function*

$$\Psi(R, R_{\text{des}}) := \frac{1}{2} \text{tr}[I - R_{\text{des}}^T R],$$

*and, consistent with the assumptions for Proposition 3 in [2], suppose that (1) the initial errors satisfy the bounds:*

$$\begin{aligned} \Psi(R(0), R_{\text{des}}(0)) &< \psi_1 < 1, \\ \|e_\omega(0)\|^2 &\leq \frac{2}{\bar{\lambda}(J)} k_R (\psi_1 - \Psi(R(0), R_{\text{des}}(0))), \\ \|e_p(0)\| &< e_{p_{\max}}, \end{aligned}$$

where  $e_{p_{\max}} > 0$  is a design parameter and  $\bar{\lambda}(\cdot)$  and  $\underline{\lambda}(\cdot)$  refer to the largest, respectively, smallest eigenvalues, and (2) define  $\gamma := \sqrt{\psi_1(2 - \psi_1)} < 1$  and choose positive constants  $A_1, A_2$  and gains  $k_R, k_\omega$  such that:

$$\begin{aligned} A_1 &< \min \left\{ k_v(1 - \gamma), \sqrt{k_p m}, \frac{4mk_p k_v(1 - \gamma)^2}{k_v^2(1 + \gamma)^2 + 4mk_p(1 - \gamma)} \right\} \\ A_2 &< \min \left\{ k_\omega, \sqrt{k_R \underline{\lambda}(J)}, \frac{4k_R k_\omega \underline{\lambda}^2(J)}{k_\omega^2 \bar{\lambda}(J) + 4k_R \underline{\lambda}^2(J)} \right\} \end{aligned}$$

$$\bar{\lambda}(D_2) > \frac{4\|D_{12}\|^2}{\underline{\lambda}(D_1)},$$

where the constant matrices  $D_1, D_{12}, D_2$  (a function of the constants introduced above) are provided in Appendix I.

(ii) *Suppose problem (24) is feasible at every timestep with optimal value zero for the “capable” quadrotor and  $d_j f_z^j - p^*$  for the “incapable” quadrotor.*

*Then, the closed-loop equilibrium  $(e_p, e_v, e_R, e_\omega)$  for the object trajectory errors is exponentially stable.*

*Proof:* The results follow from the stability of the SE(3) controller [2] and the fact that the compensation scheme given in (24) and (25) results in a total applied wrench equal to the wrench commanded by the SE(3) controller.  $\square$

## IV. ONLINE FEASIBILITY

As Proposition 1 states, closed-loop exponential stability is contingent upon both feasibility and optimality of (14) and (24). By symmetry of the desired thrust, and x- and z-axes torques for a given centro-symmetric pair  $(i, j)$ , this is equivalent to the feasibility of the following problem for every capable quadrotor  $i$ :

$$0 \leq \mathbf{f}^i \leq f_{\max}, \quad W_c^i \mathbf{f}^i = \mathbf{w}_c^i, \quad \mathbf{w}^i(3) - d_i f_{c_z}^i \in [-p^*, p^*], \quad (26)$$

where  $\mathbf{w}_c^i = (f_{c_z}^i, \tau_{c_x}^i, \tau_{c_y}^{i'}, \tau_{c_z}^i)^T$ . While the control law given in eqs. (7) and (8) does not give an a priori bound on the control input, in this section we derive conservative bounds for the initial trajectory errors and reference trajectory signals so that the problem above is always feasible. We will do this in two steps. We first derive a bound on the SE(3) controller given in (7) and (8) as a function of the nominal trajectory and its derivatives, and the tracking errors. Next, we characterize the wrench output space of each quadrotor.

### A. Bounding the SE(3) Controller

We begin by deducing bounds on all tracking errors, provided the stability conditions given in Proposition 1 are satisfied. The proof of the following proposition is provided in Appendix I.

**Proposition 2** (Trajectory Tracking Bounds). *Provided that the assumptions of Proposition 1 hold, then*

$$\|e_R(t)\| \leq \sqrt{2\psi_1}, \quad \|e_\omega(t)\| \leq \sqrt{\frac{2k_R\psi_1}{\lambda(J)}}, \quad \forall t \geq 0, \quad (27)$$

$$\|e_p(t)\|^2 + \|e_v(t)\|^2 \leq \frac{k_p e_{p_{\max}}^2}{2\lambda(M_1)}, \quad \forall t \geq 0, \quad (28)$$

where  $M_1$  is the positive definite matrix given as

$$M_1 := \frac{1}{2} \begin{pmatrix} k_p & -A_1 \\ -A_1 & m \end{pmatrix}.$$

Having obtained bounds on all errors, we now bound the net SE(3) control wrench. Let the *nominal* thrust of the trajectory, i.e.,  $m\|\ddot{\sigma} - g_{e_3}\|$  be bounded between  $[b, B]$ . Then, By Cauchy-Schwarz and triangle inequalities,

$$b - k_p\|e_p\| - k_v\|e_v\| \leq f_z \leq k_p\|e_p\| + k_v\|e_v\| + B. \quad (29)$$

The SE(3) control torque is bounded as

$$\begin{aligned} \|\tau\| &\leq k_R\|e_R\| + k_\omega\|e_\omega\| + \sqrt{\lambda(J)} (\|\omega_{\text{des}}\| + \|e_\omega\|)^2 \\ &\quad + \sqrt{\lambda(J)} (\|e_\omega\| \|\omega_{\text{des}}\| + \|\dot{\omega}_{\text{des}}\|), \end{aligned} \quad (30)$$

where

$$\|\omega_{\text{des}}\| \leq \frac{X(\|e_p\|, \|e_v\|, m\|\ddot{\sigma}\|, B)}{b - k_p\|e_p\| - k_v\|e_v\|}. \quad (31)$$

The expression for  $X$  and the derivation itself are detailed in Appendix I. We now make the following simplifying assumption: while the desired angular acceleration  $\dot{\omega}_{\text{des}}$  depends upon the second derivative of the unit vector  $-\mathbf{F}_{\text{des}}/\|\mathbf{F}_{\text{des}}\|$  which in itself involves terms related to jerk feedback, we approximate this term via its nominal value as derived from the differential flatness mapping (see, e.g., [4]) and assume that the relevant errors within  $\mathbf{F}_{\text{des}}$  are negligible.

The control bounds in eqs. (29), (30), (31) are a function of tracking error bounds (Proposition 2), and the trajectory design parameters governing nominal thrust range  $[b, B]$ , jerk  $\ddot{\sigma}$ , and angular acceleration  $\dot{\omega}_{\text{des}}$ . This allows us to conservatively bound the SE(3) wrench in the object reference frame. In the next subsection, we show how to isolate the most constrained quadrotor wrench output space.

### B. Quadrotor Wrench Output Space

Consider problem (26) for any capable quadrotor  $i$  (i.e.,  $\tau_{c_y}^i > 0$ ). In order for the quadrotor to achieve a y-axis torque equal to the adjusted value  $\tau_{c_y}^{i'}$ , one requires  $\mathbf{w}_c^{i'}$  to lie in the set:

$$\begin{aligned} \mathcal{W}_c^{i'} := \left\{ \mathbf{w}_c^{i'} \in \mathbb{R}^4 : W_c^{i'} \mathbf{f}^i = \mathbf{w}_c^{i'}, 0 \leq \mathbf{f}^i \leq f_{\max}, \right. \\ \left. \mathbf{w}_c^{i'}(3) - d_i f_{c_z}^i \in [-p^*, p^*] \right\}. \end{aligned}$$

From (25), the constraint on  $\tau_{c_y}^{i'}$  is equivalent to  $\tau_{c_y}^i \in [0, 2p^*]$ . Thus, we deduce that the uncompensated, i.e., rotated  $1/N$  wrench output from the SE(3) controller for each quadrotor must lie in the set:

$$\begin{aligned} \mathcal{W}_c^i &:= \left\{ \mathbf{w}_c^i : \mathbf{w}_c^{i'} \in \mathcal{W}_c^{i'} \right\} \\ &= \left\{ \mathbf{w}_c^i : W_{c-3}^i \mathbf{f}^i = \mathbf{w}_{c-3}^i, 0 \leq \mathbf{f}^i \leq f_{\max}, \right. \\ &\quad \left. |\mathbf{w}_c^i(3)| \leq 2p^*(\mathbf{w}_{c-3}^i) \right\}, \end{aligned} \quad (32)$$

where  $W_{c-3}^i$  is the sub-matrix of  $W_c^i$  excluding the third row and  $\mathbf{w}_{c-3}^i$  is a similarly defined sub-vector of  $\mathbf{w}_c^i$ , and we leverage an absolute value constraint on  $\tau_{c_y}^i$  since a negative  $\tau_{c_y}^i$  simply implies that this is the incapable quadrotor and thus only the remaining wrench commands are relevant. The set  $\mathcal{W}_c^i$  is referred to as the quadrotor wrench output space.

It directly follows from (32) that the projection of the set  $\mathcal{W}_c^i$  onto the  $(f_{c_z}^i, \tau_{c_x}, \tau_{c_z})$  dimensions is a convex polytope. For any given point in this polytope (i.e., given  $\mathbf{w}_{c-3}^i$ ), the remaining wrench (y-axis torque) may be visualized as a vector stemming from this point with direction orthogonal to the polytope and magnitude constrained by  $p^*(\mathbf{w}_{c-3}^i)$ . In Appendix II, we establish that the set  $\mathcal{W}_c^i$  is convex and thus the inequality constraint on  $\mathbf{w}_c^i(3)$  is also convex in  $\mathbf{w}_{c-3}^i$ .

Given the result above along with Euclidean bounds on net SE(3) control thrust and torque, checking closed-loop control feasibility reduces to verifying whether or not a set of the form  $f_z/N \in [\underline{\varepsilon}_{f_z}, \bar{\varepsilon}_{f_z}]$ ,  $\|\tau\|/N \leq \varepsilon_\tau$  lies within  $\mathcal{W}_c^i$ , for all  $i = 1, \dots, N$ .

## V. TRAJECTORY PLANNING UNDER INPUT CONSTRAINTS

In this section we design the reference trajectory  $\sigma(t)$  by optimizing suitable objective functions, constrained by the requirement that the expected closed loop SE(3) wrench commands, conservatively bounded in Section IV-A, lie within the smallest quadrotor output wrench space, as characterized in Section IV-B.

We begin with a series of  $n + 1$  specified waypoints,  $\{P_i \in \mathbb{R}^3\}$ ,  $i \in \{1, 2, \dots, n + 1\}$ , for 3D position, obtained using, for instance, a sampling-based planner [23]. The traversal times between waypoints are unspecified, and will be automatically determined by the trajectory optimization. Each trajectory segment is represented by a polynomial,

$$\sigma_i(t_i) = \sum_{j=0}^L a_{ij} t_i^j, \quad 0 \leq t_i \leq T_i, \quad i \in \{1, \dots, n\} \quad (33)$$

where  $L$  is the order of the polynomial,  $a_{ij} \in \mathbb{R}^{4 \times 1}$  are the coefficients,  $t_i$  is the time within each section, and  $T_i$  is the duration of the  $i$ -th section.

Our goal is to minimize both the total time and the integration of snap squared under the input constraints. This is a challenging multi-objective optimization, whose variants are also considered in [21], [20], [22] for quadrotor planning. Notably however, the formulation in [21] only accounts for thrust constraints while [20], [22] bound thrust and angular rates. Neither of these works also consider expected closed-loop tracking errors and their subsequent effect on control

effort. In the following, we detail a penalty-based bi-level optimization method.

### A. Snap Minimization with Fixed Duration

We first consider the subproblem where we only minimize the integral of snap, assuming given section duration times  $\{T_i\}$  and neglecting input constraints. The formulation here corresponds to the one presented in [21], however, it is included here for self-containment. Formally, we solve:

$$\begin{aligned} \min_{\{a_{ij}\}} \quad & \sum_{i=1}^n \int_0^{T_i} (\ddot{\sigma}_i)^2 dt_i & (34) \\ \text{subject to} \quad & \sigma_i(0) = P_i, \quad i \in \{1, \dots, n\} \\ & \sigma_i(T_i) = P_{i+1}, \quad i \in \{1, \dots, n\} \\ & \left. \frac{d^r \sigma_i}{dt_i^r} \right|_{t_i=T_i} = \left. \frac{d^r \sigma_{i+1}}{dt_{i+1}^r} \right|_{t_{i+1}=0}, \quad r \in \{1, \dots, 5\}, \\ & \quad \quad \quad i \in \{1, \dots, n-1\}, \\ & \left. \frac{d^q \sigma_1}{dt_1^q} \right|_{t_1=0} = \left. \frac{d^q \sigma_n}{dt_n^q} \right|_{t_n=T_n} = 0, \quad q = \{1, \dots, 4\}. \end{aligned}$$

The first two constraints in (34) ensure that the trajectory passes through the given waypoints. The third constraint enforces continuity on derivatives up to fifth order. The last constraint specifies initial and final velocity, acceleration, jerk, and snap, which are all zero in our case. Using (33), the objective has an analytical quadratic form in the polynomial coefficients  $a_{ij}$ . Furthermore, all constraints are affine in  $\{a_{ij}\}$ . Thus, (34) is a QP and can be solved efficiently. Moreover, the four dimensions of  $\sigma$  are decoupled so that (34) can be solved independently for each dimension.

### B. Coordinate Descent on Section Duration

We now consider the section duration times and input constraints by solving the following higher-level optimization:

$$\begin{aligned} \min_{\{T_i\}} \quad & \sum_{i=1}^n T_i & (35) \\ \text{subject to} \quad & T_i \geq 0, \\ & \mathbf{w}_{\text{obj}}(\ddot{\sigma}, \ddot{\ddot{\sigma}}, \ddot{\ddot{\ddot{\sigma}}})/N \in \mathcal{W}_c^i, \quad \forall i = 1, \dots, N, \end{aligned}$$

where  $\sigma$  is the solution of the snap minimization subproblem. Note that the wrench expression considered in (35) corresponds to the closed-loop bounds derived in Section IV-A and are therefore, implicitly *parametric* in the tracking error bounds (Proposition 2), as well as explicitly dependent upon the derivatives of  $\sigma$ . Due to the complexity of these bounds, we leverage an exterior point method using penalty functions, and evaluate the input constraints numerically, as described in Section IV-B, by traversing the entire trajectory. Note that by ignoring closed-loop effects and only leveraging the open-loop control signal, the constraint verification reduces to checking if the open-loop thrust and torque lie within the sets  $\{\mathcal{W}_c^i\}$ , instead of the set  $\{f_z/N \in [\underline{\varepsilon}_{f_z}, \overline{\varepsilon}_{f_z}], \|\tau\|/N \leq \varepsilon_\tau\}$ .

To solve the optimization, we use a gradient-free coordinate descent algorithm and perform line search along each dimension. Coordinate descent has the advantage of avoiding ill-conditioned gradients which may arise, for example due to the input constraints, or by adding penalty functions to

the cost itself. The overall solution to (35) is detailed in Algorithm 1. In line 12, we evaluate the penalty due to the worst case violation of control bounds along the trajectory using  $\delta$ , a measure of the amount of violation. In line 15, we use golden section search to perform the one-dimension line search on the interval  $[0, T_{\max}]$ , where  $T_{\max}$  is given and large enough. Also, when doing line search on the  $i$ -th dimension, we fix  $T_j$  for  $\forall j \neq i$  and only vary  $T_i$ . The entire algorithm thus finds a locally optimal duration set in an iterative fashion.

---

### Algorithm 1 Trajectory Optimization with Input Constraints

---

```

1:  $T_i \leftarrow T_{\max}, \forall i \in \{1, \dots, n\}, r = 1$ 
2: while not converged do
3:   for  $i = 1$  to  $n$  do
4:      $T_i \leftarrow \text{LINESEARCH}(i, T, r)$ 
5:   end for
6:    $r \leftarrow 10 \times r$ 
7: end while
8:
9: function LINESEARCH( $i, T, r$ )
10:  // Construct objective function
11:   $\sigma(T) \leftarrow \text{Solve}(\text{Snap Minimization})$ 
12:   $P(T) = \max(\max(\delta(\sigma)), 0)$ 
13:   $f(T) = \sum_{i=1}^n T_i + rP(T)$ 
14:  // Line search on  $i$ -th dimension
15:   $T_i \leftarrow \text{GoldenSectionSearch}(f, T, i, [0, T_{\max}])$ 
16:  return  $T_i$ 
17: end function

```

---

An important observation here is that the problem in (35) is always feasible by selecting  $T_i$  to be sufficiently large and initial tracking errors sufficiently small. Then, since the QP subproblem can be solved extremely quickly (indeed, analytically using the method given in [21]), the full bi-level optimization algorithm may be run in anytime fashion with solution quality only being a function of online computational time limits.

An example result of Algorithm 1 is shown in Figures 5, 6 and 7, where we are given six waypoints, and control inputs were verified using a single fixed size inner approximation to the quadrotor wrench output spaces, namely:  $\underline{\varepsilon}_{f_z} = (mg - 0.5)/N$ ,  $\overline{\varepsilon}_{f_z} = (mg + 0.5)/N$ ,  $\varepsilon_\tau = 0.01/N$ . As shown in the figures, the reference thrust and moments using the full optimization stay within the selected bounds at all times.

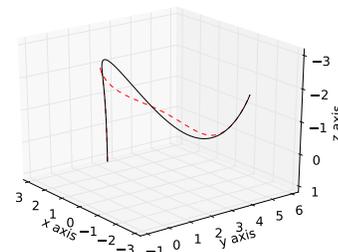


Fig. 5. Trajectory comparison between using only snap minimization with handpick section duration (dashed line) and the full optimization with input constraints (35) (solid line).

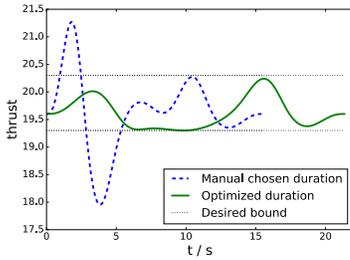


Fig. 6. Reference thrust comparison, where  $[(mg - 0.5), (mg + 0.5)]$  is the desired bound on the thrust for entire assembly and  $mg = 19.8\text{N}$ . Algorithm 1 effectively bounds the thrust without significantly elongating the total time.

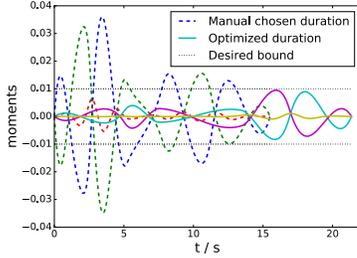


Fig. 7. Reference moments comparison where  $\|\tau\| \leq 0.01\text{Nm}$  is the torque bound for the entire assembly. All three moments stay within the specified bounds for all time.

## VI. SIMULATION

In this section we present simulation studies<sup>1</sup> validating the proposed distributed control algorithms. The snapshots of our case study is shown in Figure 10. Eight quadrotors are used to lift an object and traverse a complex 3D environment, where the straight line path to the destination is blocked and numerous 3D maneuvers are necessary to avoid collision. This emulates a disaster relief scenario where the highly unstructured space is difficult to navigate for humans and ground robots. The quadrotor-object assembly weighs 2kg, with moment of inertia  $0.17\text{kg}\cdot\text{m}^2$  along  $x, y$  axis and  $0.34\text{kg}\cdot\text{m}^2$  along the  $z$  axis. Each quadrotor has a small footprint with maximum thrust capability  $f_{\max} = 0.9\text{N}$  per rotor; therefore at least 6 quadrotors are needed to balance gravity. We used Algorithm 1 to generate the reference trajectory, with waypoints generated using FMT\* [23]. During the simulation, each quadrotor independently computes its control inputs using (24) given the trajectory broadcast. As shown in Figure 10, the quadrotors successfully follow the reference trajectory and transport the object to the destination. The position tracking performance and Euler angles of the object are plotted in Figure 8.

To demonstrate the robustness of our approach, we performed additional simulations by adding the following challenges: (i) independent zero-mean Gaussian noise is applied to the sensors on all quadrotors, (ii) the attachment points of the quadrotors are randomly perturbed within a  $0.05\text{m}$  radius around their nominal location on the object (which is  $0.5\text{m}$  away from the center of mass of the object), thereby violating the centro-symmetry condition, and (iii) initial

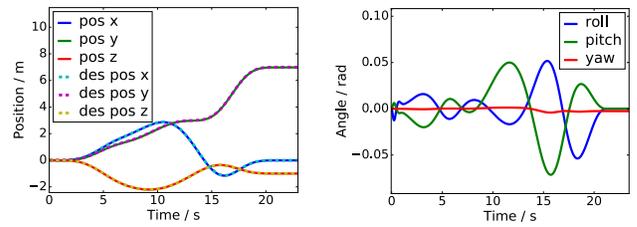


Fig. 8. The desired and actual position (left), actual Euler angles (right) of the object during the simulation. The two position curves overlap with each other, indicating an excellent tracking performance.

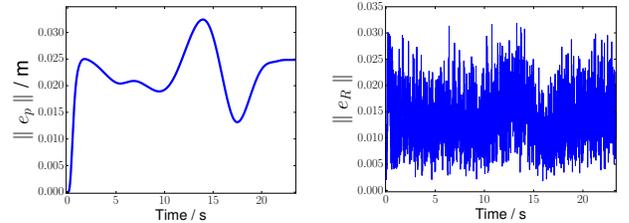


Fig. 9. Position ( $\|e_p\|$ ) and attitude ( $\|e_R\|$ ) errors for the non-centro-symmetric configuration. Each quadrotor's location is randomly perturbed within a 5cm radius around its nominal location (0.5m away from object center of mass). The assembly still demonstrates tight tracking performance, an encouraging result despite the violation of centro-symmetry.

tracking errors up to 0.1m are introduced. In Figure 9, we plot the magnitude of position  $\|e_p\|$  and attitude  $\|e_R\|$  errors for the non-centro-symmetric configuration (i.e., Challenge (i)). Figure 11 illustrates these errors when subject to all three challenges listed above. One observes that the assembly still demonstrates tight tracking performance despite the violation of centro-symmetry and effects of noise, verifying the practicality of our approach.

## VII. CONCLUSION AND FUTURE WORK

In this work we presented a distributed algorithm to transport heavy objects using a fleet of rigidly attached aerial robots with no peer communication. Under a mild geometric assumption, we rigorously analyzed pairwise controllability and derived a compensation scheme to guarantee collective group control authority and ensure stable tracking behavior. The feasibility of the algorithm is ensured by bounding the expected closed-loop control, characterizing the wrench capabilities of each quadrotor in the assembly, and explicitly enforcing these constraints along the time- and snap-optimized trajectory. The algorithms were thoroughly tested in simulation and shown to be resilient to sensor noise and violation of the symmetry assumption.

We provide two key avenues for future investigation. First, we wish to investigate online adaptation techniques to eliminate the centrosymmetry condition and improve control allocation efficiency. Second, we plan to validate our algorithms on a hardware testbed for a variety of lifting configurations.

## REFERENCES

- [1] Z. Wang and M. Schwager, "Force-amplifying n-robot transport system (force-ants) for cooperative planar manipulation without commu-

<sup>1</sup>The source code is available at: <https://github.com/StanfordMSL/QuadsManip>

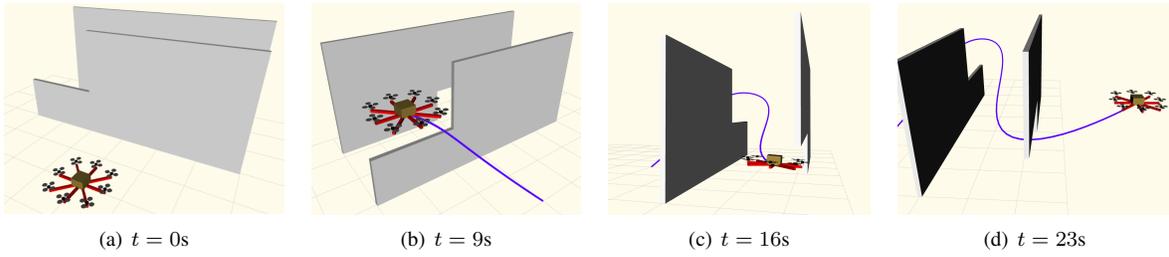


Fig. 10. Snapshots of the simulation in rviz as the assembly moves from rest, through a narrow corridor, and under a wall, to successfully transport the object to the destination. The blue solid line is the actual trajectory of the object. Simulation video available at: <https://youtu.be/MY0gharJuoY>.

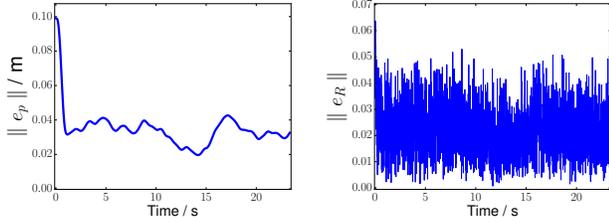


Fig. 11. Position ( $\|e_p\|$ ) and attitude ( $\|e_R\|$ ) errors for a non-centro-symmetric configuration with noisy sensors and non-zero initial error.

- nication,” *The International Journal of Robotics Research*, vol. 35, no. 13, pp. 1564–1586, 2016.
- [2] T. Lee, M. Leok, and N. H. McClamroch, “Control of complex maneuvers for a quadrotor UAV using geometric methods on  $SE(3)$ ,” *arXiv:1003.2005v4*, 2011, [Online]. Available: <http://arxiv.org/abs/1003.2005v4>.
  - [3] D. Mellinger, *Trajectory generation and control for quadrotors*. Ph.D. thesis, University of Pennsylvania, 2012.
  - [4] D. Zhou and M. Schwager, “Vector field following for quadrotors using differential flatness,” in *Robotics and Automation (ICRA), IEEE International Conference on*, 2014, pp. 6567–6572.
  - [5] Z. Wang, G. Yang, X. Su, and M. Schwager, “Oujibabots: Omnidirectional robots for cooperative object transport with rotation control using no communication,” in *International Symposium on Distributed Autonomous Robotics Systems (DARS)*, 2016.
  - [6] J. Chen, M. Gauci, W. Li, A. Kolling, and R. Groß, “Occlusion-based cooperative transport with a swarm of miniature mobile robots,” *IEEE Transactions on Robotics*, vol. 31, no. 2, pp. 307–321, 2015.
  - [7] H. Farivarnejad, S. Wilson, and S. Berman, “Decentralized sliding mode control for autonomous collective transport by multi-robot systems,” in *Decision and Control (CDC), IEEE 55th Conference on*, 2016, pp. 1826–1833.
  - [8] M. H. M. Alkilabi, A. Narayan, and E. Tuci, “Cooperative object transport with a swarm of e-puck robots: robustness and scalability of evolved collective strategies,” *Swarm Intelligence*, vol. 11, no. 3, pp. 185–209, 2017.
  - [9] S. T. Kalat, S. G. Faal, and C. D. Onal, “Scalable collective impedance control of an object via a decentralized force control method,” in *American Control Conference (ACC)*, 2017, pp. 2680–2686.
  - [10] P. Culbertson and M. Schwager, “Decentralized adaptive control for collaborative manipulation,” in *Robotics and Automation (ICRA), IEEE International Conference on*, 2018.
  - [11] D. Mellinger, M. Shomin, N. Michael, and V. Kumar, “Cooperative grasping and transport using multiple quadrotors,” in *Distributed autonomous robotic systems*. Springer, 2013, pp. 545–558.
  - [12] G. Gioioso, A. Franchi, G. Salvietti, S. Scheggi, and D. Prattichizzo, “The flying hand: A formation of uavs for cooperative aerial tele-manipulation,” in *Robotics and Automation (ICRA), IEEE International Conference on*, 2014, pp. 4335–4341.
  - [13] N. Michael, J. Fink, and V. Kumar, “Cooperative manipulation and transportation with aerial robots,” *Autonomous Robots*, vol. 30, no. 1, pp. 73–86, 2011.
  - [14] K. Sreenath, N. Michael, and V. Kumar, “Trajectory generation and control of a quadrotor with a cable-suspended load—a differentially-flat

hybrid system,” in *Robotics and Automation (ICRA), IEEE International Conference on*, 2013, pp. 4888–4895.

- [15] A. Tagliabue, M. Kamel, S. Verling, R. Siegwart, and J. Nieto, “Collaborative transportation using mavs via passive force control,” in *Robotics and Automation (ICRA), IEEE International Conference on*, 2017, pp. 5766–5773.
- [16] D. Brescianini and R. D’Andrea, “Design, modeling and control of an omni-directional aerial vehicle,” in *Robotics and Automation (ICRA), IEEE International Conference on*, 2016, pp. 3261–3266.
- [17] H. Lee, H. Kim, and H. J. Kim, “Planning and control for collision-free cooperative aerial transportation,” *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 1, pp. 189–201, 2016.
- [18] M. Tognon, B. Yüksel, G. Buondonno, and A. Franchi, “Dynamic decentralized control for protocentric aerial manipulators,” in *Robotics and Automation (ICRA), IEEE International Conference on*, 2017, pp. 6375–6380.
- [19] J. Alonso-Mora, S. Baker, and D. Rus, “Multi-robot formation control and object transport in dynamic environments via constrained optimization,” *The International Journal of Robotics Research*, vol. 36, no. 9, pp. 1000–1021, 2017.
- [20] M. W. Mueller, M. Hehn, and R. D’Andrea, “A computationally efficient motion primitive for quadcopter trajectory generation,” *IEEE Transactions on Robotics*, vol. 31, no. 6, pp. 1294–1310, 2015.
- [21] C. Richter, A. Bry, and N. Roy, “Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments,” in *Robotics Research*. Springer, 2016, pp. 649–666.
- [22] M. Hehn and R. D’Andrea, “Real-time trajectory generation for quadrotors,” *IEEE Transactions on Robotics*, vol. 31, no. 4, pp. 877–892, 2015.
- [23] L. Janson, E. Schmerling, A. Clark, and M. Pavone, “Fast marching tree: A fast marching sampling-based method for optimal motion planning in many dimensions,” *The International Journal of Robotics Research*, vol. 34, no. 7, pp. 883–921, 2015.

## APPENDIX I

### SE(3) CONTROL WRENCH BOUNDS

We first define the matrices  $D_1, D_{12}, D_2$  stated in Proposition 1.

$$D_1 = \begin{bmatrix} \frac{A_1 k_p}{m} (1 - \gamma) & -\frac{A_1 k_v}{2m} (1 + \gamma) \\ -\frac{A_1 k_v}{2m} (1 + \gamma) & k_v (1 - \gamma) - A_1 \end{bmatrix} \quad (36)$$

$$D_{12} = \begin{bmatrix} \frac{A_1}{m} B & 0 \\ B + k_p e_{p_{\max}} & 0 \end{bmatrix} \quad (37)$$

$$D_2 = \begin{bmatrix} \frac{A_2 k_R}{\lambda(J)} & -\frac{A_2 k_\omega}{2\lambda(J)} \\ -\frac{A_2 k_\omega}{2\lambda(J)} & k_\omega - A_2 \end{bmatrix}, \quad (38)$$

where recall that  $B$  is the upper bound on the open-loop thrust  $m\|\ddot{\sigma} - g\mathbf{e}_3\|$ .

*Proof:* [Proof of Proposition 2] Provided the conditions stated in the proposition above hold, Prop. 3 in [2] establishes the following conclusions: First,

$$\Psi(R(t), R_{\text{des}}(t)) \leq \psi_1 \quad \forall t \geq 0, \quad (39)$$

i.e., the attitude error, represented by the rotation matrix  $R_{\text{des}}^T R$  is less than  $90^\circ$  for all time. Second, the function

$$V_R := \frac{1}{2} \|e_\omega\|_J^2 + k_R \Psi(R, R_{\text{des}}), \quad (40)$$

is non-increasing, and third, the function

$$V := \|z_1\|_{M_1}^2 + \|z_2\|_{M_2}^2, \quad (41)$$

where  $z_1 := (\|e_p\|, \|e_v\|)^T$ ,  $z_2 := (\|e_R\|, \|e_\omega\|)^T$ , and  $M_2$  is a strictly positive definite matrix, is bounded above by  $(1/2)k_p e_{p_{\max}}^2$  for all  $t \geq 0$ . The matrix  $M_2$  is defined as:

$$M_2 := \frac{1}{2} \begin{pmatrix} k_R & -A_2 \\ -A_2 & \lambda(J) \end{pmatrix}.$$

As a consequence of (39), write  $R_{\text{des}}^T R = \exp(\beta \hat{\mathbf{v}})$ , where  $\beta \in [0, \pi/2)$  and  $\mathbf{v} \in S^2$ , the 2-sphere. By Rodrigues' formula,  $\|e_R\| = |\sin \beta| = \sin \beta$  and  $\Psi(R, R_{\text{des}}) = 1 - \cos \beta \leq \psi_1 < 1$ . The bounds then follow straightforwardly from (39), (40), and (41).  $\square$

We now bound the net SE(3) control torque. Re-writing  $\omega$  as  $e_\omega + R^T R_{\text{des}} \omega_{\text{des}}$ , we obtain  $\hat{\omega} R^T R_{\text{des}} \omega_{\text{des}} = \hat{e}_\omega R^T R_{\text{des}} \omega_{\text{des}}$  which is simply the cross product of  $e_\omega$  (defined in the current object body frame) and the projection of  $\omega_{\text{des}}$  into the current body frame. Thus, we obtain

$$\|\hat{\omega} R^T R_{\text{des}} \omega_{\text{des}}\| \leq \|e_\omega\| \|\omega_{\text{des}}\|.$$

Finally,  $\|\hat{\omega} J \omega\|$  is trivially bounded above by  $\sqrt{\lambda(J)} (\|\omega_{\text{des}}\| + \|e_\omega\|)^2$ . Thus, the net desired torque is bounded by

$$\begin{aligned} \|\tau\| &\leq k_R \|e_R\| + k_\omega \|e_\omega\| + \\ &\quad \sqrt{\lambda(J)} (\|\omega_{\text{des}}\| + \|e_\omega\|)^2 + \\ &\quad \sqrt{\lambda(J)} (\|e_\omega\| \|\omega_{\text{des}}\| + \|\dot{\omega}_{\text{des}}\|). \end{aligned}$$

To obtain a bound on  $\omega_{\text{des}}$ , note that

$$\begin{pmatrix} \omega_{des_y} \\ -\omega_{des_x} \\ 0 \end{pmatrix} = R_{\text{des}}^T \left( \left( \frac{\mathbf{F}_{\text{des}} \mathbf{F}_{\text{des}}^T}{\|\mathbf{F}_{\text{des}}\|^2} - I \right) \frac{\dot{\mathbf{F}}_{\text{des}}}{\|\mathbf{F}_{\text{des}}\|} \right)$$

which is simply the orthogonal projection of  $\dot{\mathbf{F}}_{\text{des}}/\|\mathbf{F}_{\text{des}}\|$  onto the plane with normal  $\mathbf{F}_{\text{des}}/\|\mathbf{F}_{\text{des}}\|$  [22]. Furthermore, by appropriately choosing  $\dot{\sigma}_\psi$  (and via integration,  $\sigma_\psi$ ) online, we constrain  $\omega_{des_z}$  at 0. Then,

$$\begin{aligned} \|\omega_{\text{des}}\| &\leq \left\| \frac{\dot{\mathbf{F}}_{\text{des}}}{\|\mathbf{F}_{\text{des}}\|} \right\| \\ &\leq \frac{\| -k_p e_v - k_v \dot{e}_v + m \ddot{\sigma} \|}{b - k_p \|e_p\| - k_v \|e_v\|} \\ &\leq \frac{X}{b - k_p \|e_p\| - k_v \|e_v\|}, \end{aligned}$$

where the last inequality follows from bounding  $\dot{e}_v$  whose expression is derived in [2], and

$$\begin{aligned} X &= \frac{k_p k_v}{m} (\gamma + 1) \|e_p\| + \\ &\quad \left( \left| \frac{k_v^2}{m} - k_p \right| + \gamma \frac{k_v^2}{m} \right) \|e_v\| \\ &\quad + m \|\ddot{\sigma}\| + \gamma \frac{k_v}{m} B. \end{aligned}$$

## APPENDIX II

### CONVEXITY OF WRENCH OUTPUT SET

**Lemma 1** (Convexity of  $\mathcal{W}_c^i$ ). *The set  $\mathcal{W}_c^i$  is convex.*

*Proof:* Consider any two elements  $\tilde{\mathbf{w}}_c^i, \hat{\mathbf{w}}_c^i \in \mathcal{W}_c^i$  and let  $\mathbf{w}_c^i := \beta \tilde{\mathbf{w}}_c^i + (1 - \beta) \hat{\mathbf{w}}_c^i$ , where  $\beta \in [0, 1]$ . Define

$$\tilde{\mathcal{F}} := \{0 \preceq \tilde{\mathbf{f}} \preceq f_{\max} : W_{c-3}^i \tilde{\mathbf{f}} = \tilde{\mathbf{w}}_{c-3}^i\}$$

$$\hat{\mathcal{F}} := \{0 \preceq \hat{\mathbf{f}} \preceq f_{\max} : W_{c-3}^i \hat{\mathbf{f}} = \hat{\mathbf{w}}_{c-3}^i\}$$

$$\mathcal{F} := \{0 \preceq \mathbf{f} \preceq f_{\max} : W_{c-3}^i \mathbf{f} = \mathbf{w}_{c-3}^i\}.$$

Then, it follows that

$$\beta \tilde{\mathcal{F}} \oplus (1 - \beta) \hat{\mathcal{F}} \subseteq \mathcal{F}, \quad (42)$$

where  $\oplus$  denotes the Minkowski sum. Now, by definition:

$$|\tilde{\mathbf{w}}_c^i(3)| \leq -2 \min_{\tilde{\mathbf{f}} \in \tilde{\mathcal{F}}} g(\tilde{\mathbf{f}}), \quad \text{and} \quad |\hat{\mathbf{w}}_c^i(3)| \leq 2 \max_{\hat{\mathbf{f}} \in \hat{\mathcal{F}}} g(\hat{\mathbf{f}}).$$

Similarly,

$$|\hat{\mathbf{w}}_c^i(3)| \leq -2 \min_{\hat{\mathbf{f}} \in \hat{\mathcal{F}}} g(\hat{\mathbf{f}}), \quad \text{and} \quad |\mathbf{w}_c^i(3)| \leq 2 \max_{\mathbf{f} \in \mathcal{F}} g(\mathbf{f}).$$

Thus, by triangle inequality,

$$\begin{aligned} |\mathbf{w}_c^i(3)| &\leq \beta |\tilde{\mathbf{w}}_c^i(3)| + (1 - \beta) |\hat{\mathbf{w}}_c^i(3)| \\ &\leq -2\beta \min_{\tilde{\mathbf{f}} \in \tilde{\mathcal{F}}} g(\tilde{\mathbf{f}}) - 2(1 - \beta) \min_{\hat{\mathbf{f}} \in \hat{\mathcal{F}}} g(\hat{\mathbf{f}}) \\ &= -2 \min_{\tilde{\mathbf{f}} \in \tilde{\mathcal{F}}, \hat{\mathbf{f}} \in \hat{\mathcal{F}}} \left( \beta g(\tilde{\mathbf{f}}) + (1 - \beta) g(\hat{\mathbf{f}}) \right) \\ &= -2 \min_{\tilde{\mathbf{f}} \in \tilde{\mathcal{F}}, \hat{\mathbf{f}} \in \hat{\mathcal{F}}} g(\beta \tilde{\mathbf{f}} + (1 - \beta) \hat{\mathbf{f}}) \\ &= -2 \min_{\mathbf{f} \in \beta \tilde{\mathcal{F}} \oplus (1 - \beta) \hat{\mathcal{F}}} g(\mathbf{f}) \\ &\leq -2 \min_{\mathbf{f} \in \mathcal{F}} g(\mathbf{f}) = 2 |p_{\min}^*(\mathbf{w}_{c-3}^i)|, \end{aligned}$$

where the second equality follows by linearity of  $g(\mathbf{f})$ , and the last inequality follows from eq. (42). Similarly, it follows that

$$|\mathbf{w}_c^i(3)| \leq 2 \max_{\mathbf{f} \in \mathcal{F}} g(\mathbf{f}) = 2 p_{\max}^*(\mathbf{w}_{c-3}^i).$$

Thus,  $\mathbf{w}_c^i$  lies in  $\mathcal{W}_c^i$ , proving convexity.