

# Decentralized Adaptive Control for Collaborative Manipulation of Rigid Bodies

Preston Culbertson , *Student Member, IEEE*, Jean-Jacques Slotine , and Mac Schwager , *Member, IEEE*

**Abstract**—In this work, we consider a group of robots working together to manipulate a rigid object to track a desired trajectory in SE(3). The robots do not know the mass or friction properties of the object, or where they are attached to the object. They can, however, access a common state measurement, either from one robot broadcasting its measurements to the team, or by all robots communicating and averaging their state measurements to estimate the state of their centroid. To solve this problem, we propose a decentralized adaptive control scheme wherein each agent maintains and adapts its own estimate of the object parameters in order to track a reference trajectory. We present an analysis of the controller's behavior, and show that all closed-loop signals remain bounded, and that the system trajectory will almost always (except for initial conditions on a set of measure zero) converge to the desired trajectory. We study the proposed controller's performance using numerical simulations of a manipulation task in 3-D, as well as hardware experiments which demonstrate our algorithm on a planar manipulation task. These studies, taken together, demonstrate the effectiveness of the proposed controller even in the presence of numerous unmodeled effects, such as discretization errors and complex frictional interactions.

**Index Terms**—Cooperative manipulators, distributed robot systems, multirobot systems, robust/adaptive control of robotic systems.

## I. INTRODUCTION

AS ROBOTS are fielded in applications ranging from disaster relief to autonomous construction, it is likely that such applications will, at times, require manipulation of objects, which exceed a single robot's actuation capacities (e.g., debris removal, material transport) [1]. Notably, when humans undertake such tasks, they are able to work together to flexibly and reliably transport objects, which are too massive for one person alone. We are interested in enabling similar capabilities in robot teams.

Manuscript received November 5, 2020; revised March 19, 2021; accepted March 25, 2021. Date of publication May 10, 2021; date of current version December 6, 2021. This work was supported in part by the NASA Space Technology Research Fellowship under Grant 80NSSC18K1180. This article was recommended for publication by Associate Editor N. Gans and Editor P. Robuffo Giordano upon evaluation of the reviewers' comments. (*Corresponding author: Preston Culbertson.*)

Preston Culbertson is with the Department of Mechanical Engineering, Stanford University, Stanford, CA 94305 USA (e-mail: pculbertson@stanford.edu).

Jean-Jacques Slotine is with the Department of Mechanical Engineering and the Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: jjs@mit.edu).

Mac Schwager is with the Department of Aeronautics and Astronautics, Stanford University, Stanford, CA 94305 USA (e-mail: schwager@stanford.edu).

This article has supplementary material provided by the authors and color versions of one or more figures available at <https://doi.org/10.1109/TRO.2021.3072021>.

Digital Object Identifier 10.1109/TRO.2021.3072021

In this work, we consider the problem of cooperative manipulation, where a group of robots works together to manipulate a common payload. This problem has long been of significant interest to both the multirobot systems community, and the broader robotics community [2], since it requires close collaboration between robots in addition to the numerous sensing and actuation challenges involved in any manipulation task.

While collaborative manipulation has been studied somewhat extensively in the literature [3]–[5], manipulation teams are still quite limited in their functionality, often due to restrictive assumptions. Many methods require exact payload knowledge (i.e., the payload's mass, inertial properties, and the location of its center of mass), which imposes the requirement that every payload be thoroughly calibrated prior to manipulation, in addition to making manipulation teams brittle in the face of parameter uncertainty. Furthermore, many algorithms require a centralized planner, which introduces both latency and a single point of failure to the manipulation system. Recent works have proposed handling payload uncertainty by explicitly estimating physical parameters before manipulation [6], [7], or via an indirect scheme, which combines a consensus-based estimator with a certainty-equivalent controller [8]. We consider a different approach, decentralized adaptive control, where each agent uses a single adaptive controller to perform estimation and control simultaneously.

In this work, we take a minimalist approach to collaborative manipulation, and present a control scheme, which requires no prior payload characterization and can be computed locally on board each agent. To achieve this, we use a controller on SE(3), together with a decentralized adaptation law, which allows the manipulation team to adapt to unknown payload dynamics; we also prove our controller's convergence to tracking a reference trajectory using Barbalat's lemma. We only require that the agents share a common state measurement, which can be achieved by having one agent broadcast its measurements to the entire manipulation team, or instead by having agents communicate and average their state measurements to estimate the state of their centroid. This estimate could also be obtained by running a consensus algorithm [9] online, using a mesh network between agents.

Intuitively, during manipulation each agent executes the proposed controller, with its individual set of parameter estimates (or, equivalently, control gains). The agents then use the common measurement to adapt their local parameters, such that the measurement point asymptotically tracks a desired trajectory. Such an approach leads to a manipulation algorithm, which can

easily scale in the number of agents, and is flexible enough to manipulate a large variety of payloads, while tolerating both parameter uncertainty and single-agent failure.

This article proposes control and adaptation laws for manipulation tasks in both the plane and 3-D space, in addition to a proof of convergence to the desired trajectory, as well as the boundedness of all closed-loop signals. We further validate the controller's performance both numerically, simulating a manipulation task in 3-D, and experimentally, using a team of ground robots performing a planar manipulation task.

## II. PRIOR WORK

Collaborative manipulation has been a subject of both sustained and broad interest in multirobot systems. Early work such as [10] first proposed multiarm manipulation strategies, which used a central computer to control each arm. This early work was followed with a flurry of interest, including [3], [11], where the authors study the dynamics and force allocation of redundant manipulators, and [12], which investigates various control strategies for multiple arms. A number of authors [13]–[15] have used impedance controllers for collaborative manipulation, in order to yield compliant object behavior, as well as to avoid large internal forces on the payload. An approach for characterizing such forces is proposed in [16].

While early work focused almost exclusively on multiarm manipulation, often with centralized architectures and only a few robots, other authors have focused instead on collaborative manipulation with large teams of mobile robots. In [5] and [17], the authors move furniture with small mobile robots, with a focus on reducing the communication and sensing requirements of manipulation algorithms. Other works have studied various decentralized manipulation strategies, including potential fields [18], caging [4], and distributed control allocation via matrix pseudoinverse [19]. In [20], the authors consider the problem of collaborative lifting using quadrotors, which rigidly grasp a common payload; they use a least-squares solution to allocate control efforts between agents.

While most work, including this article, assumes the robots are rigidly attached to the payload, some authors have studied the problem under different grasping assumptions. Lynch [21] studies the problem of collaborative manipulation with thrusters (i.e., agents can only apply force to the body along a fixed line), and investigates the minimum number of thrusters needed to achieve nonlinear controllability of the system. Other authors have studied aerial towing (manipulation via cables) in works such as [22]–[24]. The authors in [25] use a distributed optimization algorithm to perform collaborative manipulation of a cable-suspended load.

While collaborative manipulation has been studied in a variety of settings, many works rely on restrictive assumptions, such as central planning or prior knowledge of the object parameters, which limit the practical applicability of their approaches. More recent work has focused on removing these assumptions. In [26], the authors propose a controller, which allows a group of agents to reach a force consensus without communication, using only the payload dynamics; a leader can steer this consensus to have the object track a desired trajectory. Furthermore,

Wang *et al.* [27] proposed a communication-free controller for collaborative manipulation with aerial robots, using online optimization to allocate control effort. Corah and Michael [6] proposed a centralized estimation scheme, wherein the robots use an active perception strategy to minimize the uncertainty of their estimates of various payload parameters. In [7] and [28], the authors propose to first characterize the payload via an explicit distributed estimation scheme, in which robots use a communication network and local measurements to estimate both the mass and geometric properties of their common payload, before using these estimates to perform manipulation.

Previous work has proposed applying adaptive control to collaborative manipulation. The authors of [29] propose a centralized adaptive control scheme, which they demonstrate experimentally with a pair of robot arms. In [30], the authors propose a centralized robust adaptive controller for a group of mobile manipulators performing a collaborative manipulation task.

Perhaps closest to this work are the work by Liu and Arimoto [31] and Verginis *et al.* [32], which also propose distributed adaptive control schemes for collaborative manipulation. This work differs from previous literature in adapting both inertial *and* geometric parameters of the body; previous methods require exact measurement of manipulators' relative positions from the center of mass in order to invert the grasp matrix [33], which relates wrenches applied at contact points to wrenches applied at the center of mass. We argue that this assumption is quite restrictive, since, in practice, this amounts to knowing exactly how the body's translational and rotational dynamics are coupled. Accurately localizing large numbers of manipulators (especially with respect to the center of mass) can be both time consuming and difficult; it is also unclear why the body's normalized first moment of mass (i.e., the center of mass) is assumed known, while its zeroth (mass) and second moments (inertia tensor) are assumed unknown. In contrast, our method can treat a much broader class of model parameters (including inertial, geometric, and other effects such as gravity and friction) as unknowns, and handle them through online adaptation. We believe that this provides greater robustness to uncertainty in these parameters, and eliminates the need to estimate these quantities before manipulation.

An early version of this work appeared in [34]; this work differs greatly from the original version. The most fundamental difference is that this work builds on the Slotine–Li adaptive controller [35], which allows for much more efficient parameterization than the general model reference adaptive controller in the original version. We do not constrain the measurement point to lie at the center of mass, and further introduce a composite error that allows for trajectory tracking instead of simple velocity control. Finally, we present new simulation and experimental results studying the novel controller.

Our basic algorithm can also be compared to quorum sensing [36], [37], a natural phenomenon in which organisms or cells can exchange information [38] and synchronize themselves using a noisy, shared environment, instead of using direct information exchange. Such mechanisms have been shown to reduce the influence of random noise on synchronization phenomena [36], allowing for meaningful coordination without centralized control or explicit communication. The proposed

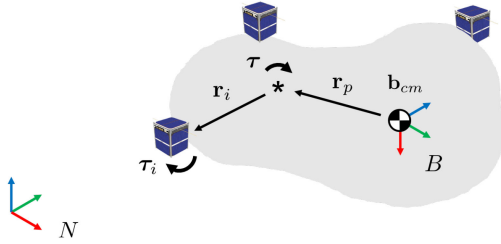


Fig. 1. Schematic of collaborative manipulation task.

algorithm can be seen as analogous to quorum sensing, wherein the common object acts as the shared environment, which allows the agents to coordinate their control actions to achieve tracking.

### III. PROBLEM STATEMENT

We consider a rigid body, which undergoes general motion (i.e., translation and rotation), during collaborative manipulation by a team of  $N$  autonomous agents. There exists a world-fixed (Newtonian) frame  $\mathcal{F}$ , as well as a body-fixed frame  $\mathcal{B}$ , which is centered at  $\mathbf{b}_{cm}$ , the body's center of mass. There also exists a body-fixed measurement point  $P$ , which is located at  $\mathbf{r}_p$  in  $\mathcal{B}$ . The body has mass  $m$  and body-frame inertia of  $\mathbf{I}_{cm}$  about its center of mass. Each agent is rigidly attached to the body at  $\mathbf{r}_i$  from  $P$ , and applies a wrench  $\boldsymbol{\tau}_i$ , i.e., a combined force and torque. Fig. 1 shows a schematic of the system.

We describe the body's pose in  $\mathcal{F}$  using a configuration variable  $\mathbf{q} = (\mathbf{x}, \mathbf{R}) \in \mathbb{R}^n \times \text{SO}(n)$ , where

$$\text{SO}(n) = \{ \mathbf{R} \in \mathbb{R}^{n \times n} \mid \mathbf{R}^T \mathbf{R} = \mathbf{I}, \det(\mathbf{R}) = 1 \}$$

is the group of rotation matrices,  $\mathbf{x}$  is the position of the reference point  $P$  in  $\mathcal{F}$ ,  $\mathbf{R}$  is the body-to-world rotation matrix relating  $\mathcal{B}$  and  $\mathcal{F}$ , and  $n \in \{2, 3\}$ , depending on if the manipulation task is in the plane or in general 3-D space, respectively. For simplicity, we say

$$\mathbb{R}^n \times \text{SO}(n) \cong \text{SE}(n)$$

i.e., we denote this set as the special Euclidean group

$$\text{SE}(n) = \left\{ \begin{bmatrix} \mathbf{R} & \mathbf{v} \\ \mathbf{0} & 1 \end{bmatrix} \mid \mathbf{R} \in \text{SO}(n), \mathbf{v} \in \mathbb{R}^n \right\}$$

which is the set of  $(n+1) \times (n+1)$  transformation matrices, since these spaces are homeomorphic. Abusing notation, we define  $\dot{\mathbf{q}} = [\dot{\mathbf{x}}^T, \boldsymbol{\omega}^T]^T \in \mathbb{R}^{3n-3}$ , which appends  $\dot{\mathbf{x}}$ , the velocity of  $P$  in  $\mathcal{F}$ , with  $\boldsymbol{\omega}$ , the body's angular rate(s) in  $\mathcal{F}$ . We now define the control problem considered in this work.

*Problem 1:* Assume each agent can measure the body's pose  $\mathbf{q}$  and rates  $\dot{\mathbf{q}}$ , but has no knowledge of its mass properties ( $m$ ,  $\mathbf{I}_{cm}$ ) or geometry ( $\mathbf{r}_p$ ,  $\mathbf{r}_i$ ). Design a decentralized control law for each agent, which allows the body's pose  $\mathbf{q}$  to track a desired trajectory  $\mathbf{q}_d(t) \in \text{SE}(n)$  asymptotically.

It is apparent that solving Problem 1 achieves collaborative manipulation under relatively few assumptions on the information available to each agent. However, the assumption of a common measurement ( $\mathbf{q}$ ,  $\dot{\mathbf{q}}$ ) warrants additional motivation.

We aim to collaboratively control both the linear and rotational dynamics of the object; but in defining a reference trajectory

$\mathbf{q}_d(t) \in \text{SE}(n)$ , we must additionally define, which point on the body we wish to track this trajectory. Yet, since we assume the body's geometry is unknown, this problem is circular: even if the agents can measure their own states, they have no way of computing or estimating the state of the measurement point, making the problem ill-posed.

In this work, we chose to resolve this ambiguity by allowing the agents to simply measure the pose and velocity of the reference point. In practice, this could be achieved by one agent broadcasting its measurements to the group; this would require a very simple "broadcast" network architecture that could easily scale by adding more robots equipped to receive the measurement. Alternatively, we note that the average of the robots' individual linear and angular velocities is equal to the body's linear and angular rates at the robots' centroid. Thus, the robots could use either an "all-to-all" communication scheme, or a mesh network and linear consensus [9] to compute the averaged measurement.

However, different assumptions can be made if we still wish to solve Problem 1, but without the use of a common measurement. One option is to restrict the class of desired trajectories  $\mathbf{q}_d$ ; if the body's angular velocity  $\boldsymbol{\omega} = \mathbf{0}$ , then the velocity is uniform across all points on the body, and robots can use their own velocity measurements  $\dot{\mathbf{x}}_i$  in order to track a desired linear velocity  $\dot{\mathbf{x}}_d$ . Similarly, if we only seek to control the body's orientation  $\mathbf{R}$ , if the agents share a reference frame, then their pose measurements are identical, making the problem again well-posed.

Finally, we can eliminate the need for a common measurement if we assume the agents know their position  $\mathbf{r}_i$  with respect to the measurement point, as in [31] and [32]. Using this information, the agents can easily compute the position and velocity of the reference point from their own measurements. However, we argue such an assumption is quite restrictive, since each agent must now have some way of localizing itself on the body before starting a manipulation task. We further note that such a method is not robust to errors in measuring  $\mathbf{r}_i$ , which could introduce unmodeled sources of uncertainty.

To solve Problem 1, we propose a modified form of the Slotine–Li adaptive manipulator controller [35], which is generalized to the multiagent case, and can adapt to the unknown mass and geometric parameters. We achieve tracking of a desired pose trajectory  $\mathbf{q}_d(t) \in \text{SE}(3)$  by introducing a composite error on  $\text{SE}(3)$ , and show that the proposed controller indeed results in asymptotic tracking, while also ensuring all signals in the system remain bounded.

## IV. PRELIMINARIES

### A. $\text{SO}(3)$ and Other Mathematical Preliminaries

In this work, we are interested in controlling both the position and orientation of a rigid body, most generally in 3-D space. While there exist numerous ways to parameterize the orientation of a rigid body, it is well-known that three-parameter representations such as Euler angles are singular, meaning they experience configurations in which the body's orientation and rotational dynamics are not uniquely defined. In this work, we instead

represent orientation using a full rotation matrix  $\mathbf{R} \in \text{SO}(3)$ , which is both unique and nonsingular.

Since  $\text{SO}(3)$  is a Lie group, it admits a Lie algebra, which captures the tangent space of the group at its identity element; this Lie algebra is in fact the set of skew-symmetric matrices  $\mathfrak{so}(3)$ , where

$$\mathfrak{so}(3) = \{\mathbf{A} \in \mathbb{R}^{3 \times 3} \mid \mathbf{A}^T = -\mathbf{A}\}.$$

We further note that  $\mathfrak{so}(3)$  forms a 3-D vector space: any matrix in  $\mathfrak{so}(3)$  can be mapped to a vector in  $\mathbb{R}^3$  and vice versa. Let us denote by  $(\cdot)^\times$  the map from  $\mathbb{R}^3$  to  $\mathfrak{so}(3)$ , i.e., for  $\mathbf{v} = [v_1, v_2, v_3]^T \in \mathbb{R}^3$

$$\mathbf{v}^\times = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix}.$$

We further denote the inverse map as  $(\cdot)^\vee$ .

Let us define the operator  $\mathbb{P}_a : \mathbb{R}^{n \times n} \mapsto \mathbb{R}^{n \times n}$ , where, for  $\mathbf{A} \in \mathbb{R}^{n \times n}$

$$\mathbb{P}_a(\mathbf{A}) = \frac{1}{2} (\mathbf{A} - \mathbf{A}^T)$$

which we call the ‘‘skew symmetric’’ part of  $\mathbf{A}$ . Furthermore, let us define  $\mathbb{P}_s : \mathbb{R}^{n \times n} \mapsto \mathbb{R}^{n \times n}$ , where

$$\mathbb{P}_s(\mathbf{A}) = \frac{1}{2} (\mathbf{A} + \mathbf{A}^T)$$

is the ‘‘symmetric’’ part. It is easy to verify that  $\mathbf{A} = \mathbb{P}_a(\mathbf{A}) + \mathbb{P}_s(\mathbf{A})$ , and for  $n = 3$ ,  $\mathbb{P}_a(\mathbf{A}) \in \mathfrak{so}(3)$ .

We further define the usual inner product on  $\mathbb{R}^{n \times n}$

$$\langle \mathbf{A}, \mathbf{B} \rangle = \text{tr}(\mathbf{A}^T \mathbf{B}).$$

Beyond the usual properties of an inner product, it is easy to verify that  $\langle \mathbb{P}_a(\mathbf{A}), \mathbb{P}_s(\mathbf{A}) \rangle = 0$ , and  $\langle \mathbb{P}_a(\mathbf{A}), \mathbb{P}_a(\mathbf{A}) \rangle = \frac{1}{2} \|\mathbb{P}_a(\mathbf{A})^\vee\|^2$ .

Finally, we say a symmetric matrix  $\mathbf{A} = \mathbf{A}^T \in \mathbb{R}^{n \times n}$  is positive semidefinite if, for all  $\mathbf{x} \in \mathbb{R}^n$ ,  $\mathbf{x}^T \mathbf{A} \mathbf{x} \geq 0$ . We say  $\mathbf{A}$  is positive definite if the inequality holds strictly for all nonzero  $\mathbf{x}$ .

### B. Convergence of Nonlinear Systems

Due its topology, no dynamical system on  $\text{SE}(3)$  has a globally asymptotically stable equilibrium [39]. Thus, we turn to relaxed notions of stability and convergence for systems with rotational dynamics. Here, we state a slightly relaxed notion of convergence, *almost global convergence*.

*Definition 1 (Almost Global Convergence):* Let  $\mathbf{x}_e$  be an equilibrium of a dynamical system  $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ , where  $\mathbf{x} \in \mathcal{X}$ . We say trajectories of the system almost globally converge to  $\mathbf{x}_e$  if, for some open, dense subset of the state space  $\mathcal{A} \subset \mathcal{X}$ , with closure  $\bar{\mathcal{A}} = \mathcal{X}$ , all trajectories with  $\mathbf{x}(0) \in \mathcal{A}$  tend toward  $\mathbf{x}_e$  asymptotically.

Similarly, we say trajectories of the system almost globally exponentially converge to  $\mathbf{x}_e$  if all trajectories with  $\mathbf{x}(0) \in \mathcal{A}$  converge exponentially to  $\mathbf{x}_e$ .

## V. EQUATIONS OF MOTION FOR COLLABORATIVE MANIPULATION

In this section, we will formulate the equations of motion for collaborative manipulation tasks, both in the plane and in three dimensions, and show they can be represented using the Lagrangian equations for robot motion [40], commonly termed ‘‘the manipulator equations.’’

### A. Rigid-Body Dynamics in Three Dimensions

Let us consider the dynamics of a rigid body being manipulated, as shown in Fig. 1.

The body’s linear motion is given by Newton’s law

$$\mathbf{F} = m \mathbf{a}_{\text{cm}} \quad (1)$$

where  $\mathbf{F}$  is the total force applied to the body, and  $\mathbf{a}_{\text{cm}}$  is the acceleration of the body’s center of mass in  $\mathcal{F}$ . We seek to relate  $\mathbf{a}_{\text{cm}}$  to  $\mathbf{a}_p$ , the acceleration of  $P$  in  $\mathcal{F}$ . We first relate the points’ velocities

$$\mathbf{v}_p = \mathbf{v}_{\text{cm}} + \boldsymbol{\omega} \times \mathbf{R} \mathbf{r}_p$$

where  $\boldsymbol{\omega}$  is the angular velocity of  $\mathcal{B}$  in  $\mathcal{F}$ , and  $\mathbf{R}$  is the rotation matrix from  $\mathcal{B}$  to  $\mathcal{F}$ . This can be differentiated to yield

$$\mathbf{a}_p = \mathbf{a}_{\text{cm}} + \boldsymbol{\alpha} \times \mathbf{R} \mathbf{r}_p + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{R} \mathbf{r}_p) \quad (2)$$

where  $\boldsymbol{\alpha}$  is  $\mathcal{B}$ ’s angular acceleration in  $\mathcal{F}$ . We can substitute (2) into (1) to yield

$$\mathbf{F} = m \mathbf{a}_p - m(\boldsymbol{\alpha} \times \mathbf{R} \mathbf{r}_p) - m \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{R} \mathbf{r}_p). \quad (3)$$

The body’s rotational dynamics are given by Euler’s rigid body equation

$$\mathbf{M}_p = \mathbf{R} \mathbf{J}_p \mathbf{R}^T \boldsymbol{\alpha} + \boldsymbol{\omega} \times (\mathbf{R} \mathbf{J}_p \mathbf{R}^T \boldsymbol{\omega}) - m \mathbf{R} \mathbf{r}_p \times \mathbf{a}_p \quad (4)$$

where  $\mathbf{J}_p$  is the body’s inertia matrix about  $\mathbf{b}_p$ , which is given by

$$\mathbf{J}_p = \mathbf{J}_{\text{cm}} + m((\mathbf{r}_p^T \mathbf{r}_p) \mathbf{I} - \mathbf{r}_p \mathbf{r}_p^T)$$

where  $\mathbf{I}$  denotes the identity matrix.

We again define the configuration variable  $\mathbf{q} = [\mathbf{x}, \mathbf{R}] \in \mathbb{R}^3 \times \text{SO}(3)$ , where  $\mathbf{x}$  is the position of the measurement point in  $\mathcal{F}$ . Abusing notation, we can let  $\dot{\mathbf{q}}, \ddot{\mathbf{q}} \in \mathbb{R}^6$ , where  $\dot{\mathbf{q}} = [\dot{\mathbf{x}}^T, \boldsymbol{\omega}^T]^T$  represents the body’s linear and angular rates, and  $\ddot{\mathbf{q}} = [\ddot{\mathbf{x}}^T, \boldsymbol{\alpha}^T]^T$  represents its linear and angular accelerations.

Using these configuration variables, we can now write (2) and (4) more compactly as

$$\boldsymbol{\tau} = \mathbf{H}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} \quad (5)$$

where  $\boldsymbol{\tau}$  is the total wrench applied to the body about the measurement point, where

$$\mathbf{H}(\mathbf{q}) = \begin{bmatrix} m \mathbf{I} & m(\mathbf{R} \mathbf{r}_p)^\times \\ -m(\mathbf{R} \mathbf{r}_p)^\times & \mathbf{R} \mathbf{J}_p \mathbf{R}^T \end{bmatrix} \quad (6)$$

is the system’s inertia matrix (which is symmetric, positive definite), and

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} \mathbf{0}_3 & m \boldsymbol{\omega}^\times (\mathbf{R} \mathbf{r}_p)^\times \\ -m \boldsymbol{\omega}^\times (\mathbf{R} \mathbf{r}_p)^\times & \boldsymbol{\omega}^\times \mathbf{R} \mathbf{J}_p \mathbf{R}^T - m((\mathbf{R} \mathbf{r}_p)^\times \dot{\mathbf{x}})^\times \end{bmatrix}$$

is a matrix which contains centrifugal and Coriolis terms, with the  $3 \times 3$  zero matrix denoted by  $\mathbf{0}_3$ .

Furthermore, we can show that these matrices have a number of interesting properties. Specifically,  $\mathbf{H}(\mathbf{q})$  is positive definite for all  $\mathbf{q}$ , and  $\dot{\mathbf{H}} - 2\mathbf{C}$  is skew-symmetric. These properties can be viewed as matrix versions of common properties of Hamiltonian systems (e.g., conservation of energy). Proof of these properties is included in Appendix A.

One can easily derive similar expressions for the 2-D case, when the body is restricted to move in the plane; we omit these expressions for brevity.

### B. Grasp Matrix and Its Properties

While the dynamics in (5) describe the body's motion under the total wrench  $\boldsymbol{\tau}$  about  $\mathbf{b}_p$ , we must further express  $\boldsymbol{\tau}$  as a function of each robot's applied wrench  $\boldsymbol{\tau}_i$ . We can write  $\boldsymbol{\tau}$  as

$$\boldsymbol{\tau} = \sum_{i=1}^N \mathbf{M}(\mathbf{q}, \mathbf{r}_i) \boldsymbol{\tau}_i \quad (7)$$

where  $\mathbf{M}$  denotes the grasp matrix, which is given by

$$\mathbf{M}(\mathbf{q}, \mathbf{r}_i) = \begin{bmatrix} \mathbf{I} & \mathbf{0}_3 \\ (\mathbf{R}\mathbf{r}_i)^\times & \mathbf{I} \end{bmatrix}$$

where agent  $i$  grasps the object at some point  $\mathbf{r}_i$  from the measurement point. We denote the set of such grasp matrices as  $\mathcal{M}$ .

The grasp matrix has some unique properties, specifically

$$\mathbf{M}(\mathbf{r}_i)^{-1} = \begin{bmatrix} \mathbf{I} & \mathbf{0}_3 \\ -(\mathbf{R}\mathbf{r}_i)^\times & \mathbf{I} \end{bmatrix} = \mathbf{M}(-\mathbf{r}_i)$$

and

$$\begin{aligned} \mathbf{M}(\mathbf{r}_i)\mathbf{M}(\mathbf{r}_j) &= \begin{bmatrix} \mathbf{I} & \mathbf{0}_3 \\ (\mathbf{R}(\mathbf{r}_i + \mathbf{r}_j))^\times & \mathbf{I} \end{bmatrix} \\ &= \mathbf{M}(\mathbf{r}_i) + \mathbf{M}(\mathbf{r}_j) - \mathbf{I} \end{aligned} \quad (8)$$

Thus, interestingly, for two matrices in  $\mathcal{M}$ , their product is simply equal to their sum, minus the identity.

## VI. DECENTRALIZED ADAPTIVE TRAJECTORY TRACKING

We will now propose a decentralized adaptive controller suitable for accomplishing the collaborative manipulation tasks outlined previously, and provide proofs of boundedness of signals, and almost global asymptotic tracking.

### A. Proposed Composite Error on $\text{SE}(3) \times \mathbb{R}^6$

We begin by proposing a composite error for 3-D poses and their derivatives, which lie in  $\text{SE}(3) \times \mathbb{R}^6$ . Adaptive controllers for manipulators in literature (cf., [41, Ch. 9]) typically use a composite error  $\mathbf{s}$ , which linearly combines position and velocity errors, and exhibits linear, exponentially stable dynamics on the surface  $\mathbf{s} = 0$ . Such a variable is appropriate for configuration variables (such as positions and small joint angles), which lie in Cartesian space.

However, since we aim to track both a desired position  $\mathbf{x}_d$  and orientation  $\mathbf{R}_d$ , the system's dynamics do not evolve naturally in Cartesian space. We, thus, propose a composite error  $\mathbf{s}$ , which is nonsingular, inherits the topology of  $\text{SE}(3)$ , and exhibits almost global convergence when on the surface  $\mathcal{D} = \{(\mathbf{q}_e, \dot{\mathbf{q}}_e) \in \text{SE}(3) \times \mathbb{R}^6 \mid \mathbf{s}(\mathbf{q}_e, \dot{\mathbf{q}}_e) = \mathbf{0}\}$ . To achieve this, we adapt the sliding variable on  $\text{SO}(3)$  from [42], and combine it with a typical linear term on  $\mathbb{R}^3$  to define a composite error on  $\text{SE}(3)$ . We then show almost all trajectories of the system converge to the desired trajectory when  $\mathbf{s} = \mathbf{0}$ .

We aim to control the body to track a desired position  $\mathbf{x}_d(t)$ , and a desired orientation,  $\mathbf{R}_d$ , which satisfies

$$\dot{\mathbf{R}}_d = \boldsymbol{\omega}_d^\times \mathbf{R}_d$$

for some desired angular velocity  $\boldsymbol{\omega}_d$ . Let us define the rotational and angular velocity errors as  $\mathbf{R}_e = \mathbf{R}_d^T \mathbf{R}$ , and  $\boldsymbol{\omega}_e = \boldsymbol{\omega} - \boldsymbol{\omega}_d$ , noting

$$(\mathbf{R}_e, \boldsymbol{\omega}_e) = (\mathbf{I}, \mathbf{0}) \iff (\mathbf{R}, \boldsymbol{\omega}) = (\mathbf{R}_d, \boldsymbol{\omega}_d).$$

We can then obtain the rotation error dynamics

$$\begin{aligned} \dot{\mathbf{R}}_e &= \dot{\mathbf{R}}_d^T \mathbf{R} + \mathbf{R}_d^T \dot{\mathbf{R}} = \mathbf{R}_d^T (\boldsymbol{\omega} - \boldsymbol{\omega}_d)^\times \mathbf{R} \\ &= (\mathbf{R}_d^T (\boldsymbol{\omega} - \boldsymbol{\omega}_d))^\times \mathbf{R}_d^T \mathbf{R} = (\mathbf{R}_d^T \boldsymbol{\omega}_e)^\times \mathbf{R}_e. \end{aligned} \quad (9)$$

Finally, we propose a composite error  $\mathbf{s} : \text{SE}(3) \times \mathbb{R}^6 \mapsto \mathbb{R}^6$ , which is given by

$$\mathbf{s} = \begin{bmatrix} \mathbf{s}_\ell \\ \boldsymbol{\sigma} \end{bmatrix} = \begin{bmatrix} \dot{\tilde{\mathbf{x}}} + \lambda \tilde{\mathbf{x}} \\ \boldsymbol{\omega}_e + \lambda \mathbf{R}_d \mathbb{P}_a(\mathbf{R}_e)^\vee \end{bmatrix}$$

where  $\tilde{\mathbf{x}} = \mathbf{x} - \mathbf{x}_d$  is the linear tracking error, and  $\mathbf{s}_\ell : \mathbb{R}^3 \times \mathbb{R}^3 \mapsto \mathbb{R}^3$  and  $\boldsymbol{\sigma} : \text{SO}(3) \times \mathbb{R}^3 \mapsto \mathbb{R}^3$  are composite errors for the object's linear and angular dynamics, respectively.

### B. System Behavior on $\mathbf{s} = \mathbf{0}$

We now consider the system's behavior when constrained to the surface  $\mathbf{s} = \mathbf{0}$ . The behavior of the linear dynamics is obvious; imposing the condition  $\mathbf{s}_\ell = \mathbf{0}$  results in the linear error dynamics

$$\dot{\tilde{\mathbf{x}}} = -\lambda \tilde{\mathbf{x}}$$

which defines a stable linear system. Thus, if the system lies on the surface, the position error  $\tilde{\mathbf{x}}$  tends exponentially to zero, with time constant  $\frac{1}{\lambda}$ .

Thus, we turn our attention to the rotational error dynamics defined by  $\boldsymbol{\sigma} = \mathbf{0}$ . Imposing this condition results in the rotational dynamics

$$\begin{aligned} \dot{\mathbf{R}}_e &= (-\lambda \mathbf{R}_d^T \mathbf{R}_d \mathbb{P}_a(\mathbf{R}_e)^\vee)^\times \mathbf{R}_e \\ &= -\lambda \mathbb{P}_a(\mathbf{R}_e) \mathbf{R}_e \end{aligned}$$

using the rotation error dynamics given by (9). We note that  $(\mathbf{R}_e, \boldsymbol{\omega}_e) = (\mathbf{I}, \mathbf{0})$  both lies on the surface  $\mathbf{s} = \mathbf{0}$ , and is an equilibrium of the reduced-order rotational dynamics.

*Theorem 1:* Almost all trajectories  $(\mathbf{R}_e, \boldsymbol{\omega}_e)$  of the reduced-order system given by  $\boldsymbol{\sigma} = \mathbf{0}$  converge exponentially to the desired equilibrium  $(\mathbf{I}, \mathbf{0})$ .

*Proof:* Consider the Lyapunov-like function

$$V_R(t) = \text{tr}(\mathbf{I} - \mathbf{R}_e) > 0$$

which is lower bounded by zero since  $-1 \leq \text{tr}(\mathbf{R}_e) \leq 3$  for all  $\mathbf{R}_e \in \text{SO}(3)$ , with  $\text{tr}(\mathbf{R}_e) = 3$  iff  $\mathbf{R}_e = \mathbf{I}$ . The time derivative of  $V_R$  is given by

$$\begin{aligned} \dot{V}_R &= -\text{tr}(\dot{\mathbf{R}}_e) = \lambda \langle -\mathbb{P}_a(\mathbf{R}_e), \mathbf{R}_e \rangle \\ &= -\lambda \langle \mathbb{P}_a(\mathbf{R}_e), \mathbb{P}_a(\mathbf{R}_e) + \mathbb{P}_s(\mathbf{R}_e) \rangle \\ &= -\frac{\lambda}{2} \|\mathbb{P}_a(\mathbf{R}_e)^\vee\|^2 \leq 0 \end{aligned}$$

using the matrix inner product  $\langle \mathbf{A}, \mathbf{B} \rangle = \text{tr}(\mathbf{A}^T \mathbf{B})$ , and the fact that  $\langle \mathbb{P}_a(\mathbf{A}), \mathbb{P}_s(\mathbf{A}) \rangle = 0$ , for  $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{3 \times 3}$ .

In order to show almost global exponential convergence to  $(\mathbf{I}, \mathbf{0})$ , we seek a strictly positive constant  $\alpha > 0$  such that

$$\dot{V}_R \leq -\alpha V_R.$$

To find this constant, let us recall that for the rotation matrix  $\mathbf{R}_e$  there is an equivalent unit quaternion representation

$$\mathbf{z}(t) = q_0 + q_1 i + q_2 j + q_3 \tilde{k}$$

with  $\text{tr}(\mathbf{R}_e) = 4q_0^2 - 1$ , and  $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$ . Furthermore, we can rewrite  $\dot{V}_R$

$$\dot{V}_R = \frac{\lambda}{2} \text{tr}(\mathbf{R}_e \dot{\mathbf{R}}_e - \mathbf{I}) = 2\lambda \left( (q_0^2 - q_1^2 - q_2^2 - q_3^2)^2 - 1 \right)$$

where the second equality follows from quaternion multiplication. Since  $\mathbf{z}$  has unit norm, we can write

$$\begin{aligned} \dot{V}_R &= 2\lambda \left( (2q_0^2 - 1)^2 - 1 \right) = 8\lambda q_0^2 (q_0^2 - 1) \\ &= - (2\lambda q_0^2) V_R. \end{aligned}$$

Furthermore, we note  $q_0^2 = 0$  if and only if  $\text{tr}(\mathbf{R}_e) = -1$ , which corresponds to the global maximum of  $V_R$ .

Let us assume our trajectory does not start at this point, i.e., there exists some  $\epsilon > 0$  such that  $4\tilde{\sim} \geq \tilde{\sim} V_R(0) + \epsilon$ . Then, since  $\dot{V}_R \leq 0$

$$q_0^2(t) = 1 - \frac{1}{4} V_R(t) \geq 1 - \frac{1}{4} V_R(0) > \frac{\epsilon}{4} > 0.$$

Thus, for all  $t \geq 0$ , we have

$$\dot{V}_R \leq -\alpha V_R$$

for  $\alpha = \frac{1}{2} \lambda \epsilon > 0$ . Thus, almost all trajectories on  $\sigma = \mathbf{0}$  converge exponentially to the desired equilibrium. ■

Thus, we can conclude that almost all trajectories satisfying  $\mathbf{s} = \mathbf{0}$  will exponentially converge to tracking the desired position  $\mathbf{x}_d$  and orientation  $\mathbf{R}_d^1$ .

We further note that  $\mathbf{s}$  can be interpreted as a “velocity error,” since we can write

$$\mathbf{s} = \dot{\mathbf{q}} - \dot{\mathbf{q}}_r$$

<sup>1</sup>Note the composite error  $\mathbf{s}$  discussed here is closely related to the “sliding variable” commonly used in sliding-mode control [41, Ch. 7], since, when we constrain  $\mathbf{s} = \mathbf{0}$ , the reduced-order system dynamics converge exponentially to tracking. However, controller (13) proposed in the following discussion is not, in fact, a sliding-mode controller because it does not include the discontinuous term typically used to guarantee finite-time convergence to  $\mathbf{s} = \mathbf{0}$ ; we show the proposed controller instead exhibits asymptotic tracking.

where  $\dot{\mathbf{q}}_r$  is a “reference velocity,” which augments the desired velocity  $\dot{\mathbf{q}}_d$  with an additional term based on the pose error

$$\dot{\mathbf{q}}_r = \begin{bmatrix} \dot{\mathbf{x}}_d - \lambda \tilde{\mathbf{x}} \\ \boldsymbol{\omega}_d - \lambda \mathbf{R}_d \mathbb{P}_a(\mathbf{R}_e)^\vee \end{bmatrix}.$$

### C. Decentralized Adaptive Trajectory Control

We now return to Problem 1, decentralized adaptive control of a common payload. We consider the Lagrangian dynamics

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g} = \sum_{i=1}^N \mathbf{M}_i(\mathbf{q})\boldsymbol{\tau}_i \quad (10)$$

which, as shown previously, hold for collaborative manipulation tasks on both SE(2) and SE(3). We note that while the manipulation dynamics do not include a term  $\mathbf{g}$ , which models conservative forces such as gravity, we include it here for completeness, and to conform to the traditional “manipulator equation.”

Let us consider a Lyapunov-like function candidate

$$V(t) = \frac{1}{2} \left[ \mathbf{s}^T \mathbf{H} \mathbf{s} + \sum_{i=1}^N \tilde{\mathbf{o}}_i^T \boldsymbol{\Gamma}_o^{-1} \tilde{\mathbf{o}}_i + \tilde{\mathbf{r}}_i^T \boldsymbol{\Gamma}_r^{-1} \tilde{\mathbf{r}}_i \right] \quad (11)$$

where  $\tilde{\mathbf{o}}_i = \hat{\mathbf{o}}_i - \mathbf{o}_i$  is a parameter estimation error for agent  $i$ , with  $\mathbf{o}_i$  being a vector of physical constants that define the payload’s dynamics, and  $\hat{\mathbf{o}}_i$  being agent  $i$ ’s estimate of these parameters. Similarly, we let  $\tilde{\mathbf{r}}_i$  be the difference between the vector  $\mathbf{r}_i$  from the measurement point to agent  $i$ , and the agent’s estimate  $\hat{\mathbf{r}}_i$ . We further let  $\boldsymbol{\Gamma}_o$  and  $\boldsymbol{\Gamma}_r$  be positive definite matrices, which correspond to adaptation gains.

Differentiating  $V$  yields

$$\dot{V}(t) = \mathbf{s}^T \dot{\mathbf{H}} \mathbf{s} + \frac{1}{2} \mathbf{s}^T \dot{\mathbf{H}} \mathbf{s} + \sum_{i=1}^N \tilde{\mathbf{o}}_i^T \boldsymbol{\Gamma}_o^{-1} \dot{\tilde{\mathbf{o}}}_i + \tilde{\mathbf{r}}_i^T \boldsymbol{\Gamma}_r^{-1} \dot{\tilde{\mathbf{r}}}_i$$

using the fact that the parameter error derivatives  $\dot{\tilde{\mathbf{o}}}_i, \dot{\tilde{\mathbf{r}}}_i$  are equal to the parameter estimate derivatives  $\dot{\hat{\mathbf{o}}}_i, \dot{\hat{\mathbf{r}}}_i$ , since the true parameter values are constant.

Furthermore, since we can write  $\dot{\mathbf{s}} = \ddot{\mathbf{q}} - \ddot{\mathbf{q}}_r$ , we can expand the term

$$\begin{aligned} \mathbf{s}^T \dot{\mathbf{H}} \mathbf{s} &= \mathbf{s}^T \mathbf{H}(\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_r) \\ &= \mathbf{s}^T \left( \sum_{i=1}^N \mathbf{M}_i \boldsymbol{\tau}_i - \mathbf{C} \dot{\mathbf{q}} - \mathbf{g} - \mathbf{H} \ddot{\mathbf{q}}_r \right) \\ &= \mathbf{s}^T \left( \sum_{i=1}^N \mathbf{M}_i \boldsymbol{\tau}_i - \mathbf{H} \ddot{\mathbf{q}}_r - \mathbf{C} \dot{\mathbf{q}}_r - \mathbf{g} - \mathbf{C} \mathbf{s} \right) \end{aligned}$$

using the system dynamics (10) to substitute for  $\mathbf{H} \ddot{\mathbf{q}}$ , and the fact that  $\dot{\mathbf{q}} = \mathbf{s} + \dot{\mathbf{q}}_r$ .

Furthermore, since the system is overactuated, meaning it has many more control inputs than degrees of freedom, there exist many sets of control inputs  $\boldsymbol{\tau}_i$ , which produce the same object dynamics. Thus, it is sufficient for each agent to take some portion of the control effort, which, combined, enables tracking of the desired trajectory. To this end, let us introduce a set of  $N$  positive constants  $\alpha_i$ , with  $\sum_{i=1}^N \alpha_i = 1$ ; a natural choice is  $\alpha_i = \frac{1}{N}$ , which we use throughout this article.

Using these constants, we can further exploit an important property of this system, *linear parameterizability*. Importantly,

for the control tasks considered in this work, the physical parameters to be estimated appear only linearly in the dynamics. Thus, we can define a known regressor matrix  $\mathbf{Y}_o(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r)$ , such that

$$\alpha_i (\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}}_r + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}_r + \mathbf{G}) = \mathbf{Y}_o(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r)\mathbf{o}_i$$

where  $\mathbf{o}_i = \alpha_i \mathbf{o}$  is a vector of the object's physical parameters,  $\mathbf{o}$ , such as mass or moments of inertia, scaled by  $\alpha_i$ . We further note that

$$\begin{aligned} \sum_{i=1}^N \mathbf{Y}_o \mathbf{o}_i &= \sum_{i=1}^N \alpha_i (\mathbf{H}\ddot{\mathbf{q}}_r + \mathbf{C}\dot{\mathbf{q}}_r + \mathbf{g}) \\ &= \mathbf{H}\ddot{\mathbf{q}}_r + \mathbf{C}\dot{\mathbf{q}}_r + \mathbf{g}. \end{aligned}$$

In short, the constants  $\alpha_i$  are a mathematical convenience used to allocate control effort among the agents. Since they now seek to estimate the scaled physical parameters  $\mathbf{o}_i$ , the constants need not be known *a priori*. Recalling the analogy between our proposed controller and quorum sensing [37], we can interpret these constants as akin to coupling weights between the agents.

Using this regressor, we can write

$$\begin{aligned} \dot{V}(t) &= \sum_{i=1}^N \mathbf{s}^T (\mathbf{M}_i \boldsymbol{\tau}_i - \mathbf{Y}_o \mathbf{o}_i) + \frac{1}{2} \mathbf{s}^T (\dot{\mathbf{H}} - 2\mathbf{C}) \mathbf{s} \\ &\quad + \tilde{\mathbf{o}}_i^T \boldsymbol{\Gamma}_o^{-1} \dot{\tilde{\mathbf{o}}} + \tilde{\mathbf{r}}_i^T \boldsymbol{\Gamma}_r^{-1} \dot{\tilde{\mathbf{r}}}_i. \end{aligned} \quad (12)$$

As noted previously,  $\dot{\mathbf{H}} - 2\mathbf{C}$  is skew-symmetric, thus, its quadratic form is equal to zero for any vector  $\mathbf{s}$ .

We will now outline the proposed control and adaptation laws. Let

$$\boldsymbol{\tau}_i = \widehat{\mathbf{M}}_i^{-1} \mathbf{F}_i \quad (13)$$

where  $\mathbf{M}_i(\hat{\mathbf{r}}_i)^{-1}$  is the inverse of a grasp matrix in  $\mathcal{M}$  with a moment arm of  $\hat{\mathbf{r}}_i$ , and

$$\mathbf{F}_i = \mathbf{Y}_o \hat{\mathbf{o}}_i - \mathbf{K}_D \mathbf{s} \quad (14)$$

where  $\mathbf{K}_D$  is some positive definite matrix. Intuitively, the term  $\mathbf{F}_i$  combines a feedforward term using the estimated object dynamics with a simple proportional-derivative (PD) term,  $-\mathbf{K}_D \mathbf{s}$ . This control law alone would lead to successful tracking if all agents were attached at the measurement point. Each agent then multiplies  $\mathbf{F}_i$  by  $\widehat{\mathbf{M}}_i^{-1}$  in an effort to compensate for the torque its applied force generates about the measurement point.

We further note, using (8), that the product

$$\mathbf{M}_i \widehat{\mathbf{M}}_i^{-1} = (\mathbf{I} - \widetilde{\mathbf{M}}_i)$$

where  $\widetilde{\mathbf{M}}_i = \widehat{\mathbf{M}}_i - \mathbf{M}_i$ . We can further observe that the product  $\widetilde{\mathbf{M}}_i \mathbf{F}_i$  is linearly parameterizable, with

$$-\widetilde{\mathbf{M}}_i \mathbf{F}_i = \mathbf{Y}_g(\mathbf{F}_i, \mathbf{q}) \tilde{\mathbf{r}}_i.$$

Using this expression, we propose the adaptation laws

$$\dot{\tilde{\mathbf{o}}}_i = -\boldsymbol{\Gamma}_o \mathbf{Y}_o(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r)^T \mathbf{s} \quad (15)$$

$$\dot{\tilde{\mathbf{r}}}_i = -\boldsymbol{\Gamma}_r \mathbf{Y}_g(\mathbf{F}_i, \mathbf{q})^T \mathbf{s}. \quad (16)$$

Note that all terms in the proposed control and adaptation laws can be computed using only local information (using the desired

trajectory, shared measurement, and local parameter estimates). We now reach the main theoretical result of this article, i.e., the boundedness of all closed-loop signals, and almost-global asymptotic convergence of the system trajectory to the desired trajectory.

*Theorem 2:* Consider the control laws (13) and (14), and the adaptation laws (15) and (16). Under the proposed adaptive controller, all closed-loop signals remain bounded, and the system converges to the desired trajectory from almost all initial conditions.

*Proof:* Substituting these control and adaptation laws into the previous expression (12) of  $\dot{V}(t)$  yields

$$\begin{aligned} \dot{V}(t) &= \sum_{i=1}^N \mathbf{s}^T (\mathbf{F}_i + \mathbf{Y}_g \tilde{\mathbf{r}}_i - \mathbf{Y}_o \mathbf{o}_i) \\ &\quad + \tilde{\mathbf{o}}_i^T \boldsymbol{\Gamma}_o^{-1} \dot{\tilde{\mathbf{o}}} + \tilde{\mathbf{r}}_i^T \boldsymbol{\Gamma}_r^{-1} \dot{\tilde{\mathbf{r}}}_i \\ &= \sum_{i=1}^N -\mathbf{s}^T \mathbf{K}_D \mathbf{s} + \tilde{\mathbf{o}}_i^T (\mathbf{Y}_o \tilde{\mathbf{o}}_i + \boldsymbol{\Gamma}_o^{-1} \dot{\tilde{\mathbf{o}}}_i) \\ &\quad + \tilde{\mathbf{r}}_i^T (\mathbf{Y}_g \tilde{\mathbf{r}}_i + \boldsymbol{\Gamma}_r^{-1} \dot{\tilde{\mathbf{r}}}_i) \end{aligned}$$

which, upon substituting the adaptation laws (15) and (16), yields

$$\dot{V}(t) = N (-\mathbf{s}^T \mathbf{K}_D \mathbf{s}) \leq 0. \quad (17)$$

As shown in Section VI-B, when  $\mathbf{s} = 0$ , almost all trajectories converge exponentially to tracking the desired trajectory. Thus, if we have  $\mathbf{s} \rightarrow \mathbf{0}$ , this is sufficient to show the body tracks the desired trajectory asymptotically. To this end, we now show that  $\dot{V}(t) \rightarrow 0$  as  $t \rightarrow \infty$ , since from (17), it is apparent that this is sufficient for  $\mathbf{s} \rightarrow \mathbf{0}$ .

To do this, we again invoke Barbalat's Lemma [41, Ch. 4.5], which states that  $\dot{V} \rightarrow 0$  if  $V$  has a finite limit, and  $\dot{V}$  is uniformly continuous. From (17), we see that  $V$  has a finite limit; thus, we show  $\dot{V}$  is bounded.

From the expression of  $\ddot{V} = -2N(\dot{\mathbf{s}}^T \mathbf{K}_D \dot{\mathbf{s}})$ , we have  $\ddot{V}$  is bounded if  $\mathbf{s}, \dot{\mathbf{s}}$  are bounded. Since  $\dot{V} \leq 0$ , and  $V$  is lower bounded, we have that  $V$  is bounded, which implies  $\mathbf{s}, \tilde{\mathbf{o}}_i, \tilde{\mathbf{r}}_i$  are bounded for all  $i$ .

Furthermore, we can write

$$\begin{aligned} \mathbf{H}\dot{\mathbf{s}} &= \mathbf{H}(\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_r) \\ &= \sum_{i=1}^N (\mathbf{F}_i + \mathbf{Y}_g \tilde{\mathbf{r}}_i) - \mathbf{Y}_o \mathbf{o}_i - \mathbf{C}\mathbf{s} \\ &= \sum_{i=1}^N (\mathbf{Y}_o \tilde{\mathbf{o}}_i + \mathbf{Y}_g \tilde{\mathbf{r}}_i) - (N\mathbf{K}_D + \mathbf{C}) \mathbf{s} \end{aligned}$$

noting that all signals on the right-hand side are bounded (since the desired trajectory  $\mathbf{q}_d$  and its derivatives are bounded by design). Finally, since  $\mathbf{H}$  is positive definite, we know  $\mathbf{H}^{-1}$  exists, which implies that  $\dot{\mathbf{s}}$  is bounded.

Thus, we have shown all signals are bounded. Furthermore, since  $\mathbf{s} \rightarrow \mathbf{0}$  as  $t \rightarrow \infty$ , then, except for the singularity  $\text{tr}(\mathbf{R}_e(0)) \neq -1$ , we have  $\tilde{\mathbf{q}}, \tilde{\mathbf{x}} \rightarrow \mathbf{0}, \mathbf{R}_d^T \mathbf{R} \rightarrow \mathbf{I}$  in the limit, which completes the proof. ■

Similarly to [35], Theorem 2 shows boundedness of all closed-loop signals and almost-global asymptotic convergence to the desired trajectory. Many variations and extensions of [35] have been proposed. For instance, Spong *et al.* [43] demonstrated that an additional term could be added to the Lyapunov-like function of [35] to show that the desired trajectory and true parameters are stable in the sense of Lyapunov. However, this result relied on the particular linear structure of the composite error  $\mathbf{s}$ , which does not hold in our case.

#### D. Lack of Persistent Excitation

While we have shown that under the proposed controller, the tracking errors approach zero in the limit, we have not reasoned about the asymptotic behavior of the parameter errors  $\tilde{\mathbf{o}}_i, \tilde{\mathbf{r}}_i$ . While these signals must remain bounded, it is unclear under what conditions the parameter estimates converge to their true values.

For most (centralized) adaptive controllers, a condition termed ‘‘persistent excitation’’ is sufficient to show the parameter errors are driven to zero asymptotically.

*Definition 2 (Persistent Excitation):* Consider  $\mathbf{W}(t) \in \mathbb{R}^{m \times n}$ , a matrix-valued, time-varying signal. We say  $\mathbf{W}(t)$  is persistently exciting if, for all  $t > 0$ , there exist  $\gamma, T > 0$  such that

$$\frac{1}{T} \int_t^{t+T} \mathbf{W}(\tau)^T \mathbf{W}(\tau) d\tau > \gamma \mathbf{I}.$$

Let us first consider the case of single-agent manipulation, i.e., the case where  $N = 1$ . Furthermore, let us define  $\mathbf{Y}_1 = [\mathbf{Y}_o, \mathbf{Y}_r]$ , and  $\tilde{\mathbf{a}}_1 = [\tilde{\mathbf{o}}^T, \tilde{\mathbf{r}}^T]^T$ , meaning we stack the parameter errors row-wise, and their respective regressors column-wise.

In this case, we can again write the closed-loop dynamics as

$$\mathbf{H}\dot{\mathbf{s}} + (\mathbf{C} + \mathbf{K}_D)\mathbf{s} = \mathbf{Y}_1 \tilde{\mathbf{a}}_1$$

which, since we know  $\mathbf{s}, \dot{\mathbf{s}} \rightarrow \mathbf{0}$  in the limit, implies that  $\tilde{\mathbf{a}}_1$  converges to the nullspace  $\mathcal{N}(\mathbf{Y}_1)$  of  $\mathbf{Y}_1$ ; this does not, however, imply that  $\tilde{\mathbf{a}}_1 \rightarrow \mathbf{0}$ .

Instead, it is sufficient to require that the matrix  $\mathbf{Y}_{1,d}(t)$  is persistently exciting [44], where  $\mathbf{Y}_{1,d}(t) = \mathbf{Y}_1(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d)$ . However, while there exist simple conditions on the reference signal to guarantee persistent excitation for linear, time-invariant systems, to date, such simple conditions do not exist for non-linear systems. In practice, however, reference signals with a large amount of frequency content are typically used to achieve persistent excitation.

We now turn to the multiagent case. Let us define  $\mathbf{Y}_N = [\mathbf{Y}_o, \mathbf{Y}_r, \dots, \mathbf{Y}_o, \mathbf{Y}_r]$  and  $\tilde{\mathbf{a}}_N = [\tilde{\mathbf{o}}_1^T, \tilde{\mathbf{r}}_1^T, \dots, \tilde{\mathbf{o}}_N^T, \tilde{\mathbf{r}}_N^T]^T$ , i.e., we stack  $\mathcal{F}$  sets of the regressors  $\mathbf{Y}_o, \mathbf{Y}_r$  column-wise, and append all  $\mathcal{F}$  sets of parameter errors row-wise. This allows us to again write the closed-loop dynamics as

$$\mathbf{H}\dot{\mathbf{s}} + (\mathbf{C} + \mathbf{K}_D)\mathbf{s} = \mathbf{Y}_N \tilde{\mathbf{a}}_N.$$

Thus, to ensure that all parameter errors  $\tilde{\mathbf{a}}_i$  converge to zero, we hope to find conditions under which  $\mathbf{Y}_{N,d}(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d)$  is persistently exciting. Thus, for all  $t > 0$ , we require that there

exist  $T, \gamma > 0$  such that

$$\int_t^{t+T} \mathbf{Y}_{N,d}^T(\tau) \mathbf{Y}_{N,d}(\tau) d\tau \geq \gamma \mathbf{I}.$$

However, examining the structure of the inner product  $\mathbf{Y}_{N,d}^T \mathbf{Y}_{N,d}$  reveals

$$\mathbf{Y}_{N,d}^T(t) \mathbf{Y}_{N,d}(t) = \begin{bmatrix} \mathbf{Y}_o^T \mathbf{Y}_o & \mathbf{Y}_o^T \mathbf{Y}_r & \dots & \mathbf{Y}_o^T \mathbf{Y}_r \\ \mathbf{Y}_r^T \mathbf{Y}_o & \mathbf{Y}_r^T \mathbf{Y}_r & \dots & \mathbf{Y}_r^T \mathbf{Y}_r \\ \vdots & & \ddots & \vdots \\ \mathbf{Y}_r^T \mathbf{Y}_o & \mathbf{Y}_r^T \mathbf{Y}_r & \dots & \mathbf{Y}_r^T \mathbf{Y}_r \end{bmatrix}$$

which we note has the same two (block) rows repeated  $\mathcal{F}$  times. Furthermore, we note the integral

$$\int_t^{t+T} \mathbf{Y}_{N,d}^T(\tau) \mathbf{Y}_{N,d}(\tau) d\tau$$

will have the same block structure, meaning the integrated matrix must be rank deficient.

Thus, regardless of the reference signal  $\mathbf{q}_d$ ,  $\mathbf{Y}_{N,d}$  cannot be persistently exciting; therefore, we can make no claim on the asymptotic behavior of the individual parameter errors  $\tilde{\mathbf{a}}_i$ , except that they remain bounded, and their derivative  $\dot{\tilde{\mathbf{a}}}_i$  converges to zero.

However, we can also write the closed-loop dynamics as

$$\begin{aligned} \mathbf{H}(\mathbf{q})\dot{\mathbf{s}} + (\mathbf{C} + \mathbf{K}_D)\mathbf{s} &= \sum_{i=1}^N \mathbf{Y}_i \tilde{\mathbf{a}}_i \\ &= N \mathbf{Y}_1 \left( \frac{1}{N} \sum_{i=1}^N \tilde{\mathbf{a}}_i \right) \end{aligned}$$

which implies, if  $\mathbf{Y}_1$  is persistently exciting, that the average parameter error  $\frac{1}{N} \sum_{i=1}^N \tilde{\mathbf{a}}_i$  converges to zero.

Thus, we can interpret the lack of a persistently exciting signal  $\mathbf{q}_d$  as a reflection of the redundancy inherent in the system; since there exists a subspace of individual wrenches  $\boldsymbol{\tau}_i$ , which produce the same total wrench  $\boldsymbol{\tau}$  on the body, even a signal which would be persistently exciting for a single agent can only guarantee that the average parameter error vanishes.

Interestingly, this is in many ways dual to the results in [45], wherein the authors consider the problem of adaptive control for cloud-based robots. In this case, the authors have a large number of decoupled, but identical, manipulators, which they aim to control adaptively using a shared parameter estimate. The authors find that parameter convergence is guaranteed if the sum of the regressor matrices is persistently exciting, a much weaker condition than any single robot executing a persistently exciting trajectory. On the other hand, in this setting, we have a single physical system, which is controlled collaboratively by many robots using individual parameter estimates; we find that in this case persistent excitation requires not just a stronger condition on  $\mathbf{q}_d$ , but is in fact impossible to achieve.

#### E. Friction and Dissipative Dynamics

Finally, in some systems (such as the planar manipulation task), there are nonconservative forces, which act on the payload due to effects such as friction. The simplest model treats these



forces as an additive body-frame disturbance  $\mathbf{D}(\mathbf{q})\dot{\mathbf{q}}$ , which is linear in the velocity of the measurement point, where  $\mathbf{D}(\mathbf{q}) = \mathbf{R}\boldsymbol{\Lambda}_D\mathbf{R}^T$ , for a constant positive semidefinite matrix of friction coefficients  $\boldsymbol{\Lambda}_D$ . It can be easily verified that such a model is appropriate for viscous sliding friction.

This friction can be compensated quite easily using an additional term in the control law, namely

$$\mathbf{F}_i = \mathbf{Y}_o\hat{\mathbf{o}}_i + \mathbf{Y}_f\hat{\mathbf{f}}_i - \mathbf{K}_D\mathbf{s}$$

where  $\mathbf{Y}_f(\mathbf{q}, \dot{\mathbf{q}}_r)\hat{\mathbf{f}}_i = \alpha_i\mathbf{D}(\mathbf{q})\dot{\mathbf{q}}_r$ . If we additionally use the adaptation law

$$\dot{\hat{\mathbf{f}}}_i = -\Gamma_f\mathbf{Y}_f^T(\mathbf{q}, \dot{\mathbf{q}}_r)\mathbf{s}$$

with positive definite  $\Gamma_D$ , one can show this results in

$$\dot{V} = -\mathbf{s}^T(N\mathbf{K}_D + \mathbf{D}(\mathbf{q}))\mathbf{s}$$

recalling that  $\mathbf{D}\mathbf{s} = \mathbf{D}\dot{\mathbf{q}} - \mathbf{D}\dot{\mathbf{q}}_r$ . Thus, in effect, we have increased the PD gain with an additional positive semidefinite term  $\mathbf{D}(\mathbf{q})$ , yielding a higher “effective”  $\mathbf{K}_D$  matrix without increasing the gains directly.

However, since such an expression is not obtainable in general, e.g., in the case of Coulomb friction, we turn instead to the case when frictional forces are modeled as being applied at the attachment point of each agent. In this case, the frictional forces are also premultiplied by the grasp matrices, resulting in the dynamics

$$\mathbf{H}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{g} = \sum_{i=1}^N \mathbf{M}_i(\boldsymbol{\tau}_i - \mathbf{D}\mathbf{v}_i - \mathbf{D}_C \text{sgn}(\mathbf{v}_i))$$

where  $\mathbf{v}_i = \mathbf{M}_i^T\dot{\mathbf{q}}$  is the linear and angular velocity of the robot in the global frame,  $\mathbf{D}$  and  $\mathbf{D}_C$  are positive semidefinite matrices of friction coefficients, and  $\text{sgn}(\cdot)$  is the element-wise sign function.

These terms can again be compensated, now by augmenting the “shifted” control law (13)

$$\boldsymbol{\tau}_i = \widehat{\mathbf{M}}_i^{-1}\mathbf{F}_i + \mathbf{Y}_D\hat{\mathbf{d}}_i + \mathbf{Y}_c\hat{\mathbf{c}}_i$$

where we have

$$\mathbf{Y}_D(\mathbf{q}, \dot{\mathbf{q}}_r)\hat{\mathbf{d}}_i = \mathbf{M}_i\mathbf{D}\mathbf{M}_i^T\dot{\mathbf{q}}_r$$

and

$$\mathbf{Y}_C(\mathbf{q}, \mathbf{v}_i)\hat{\mathbf{c}}_i = \mathbf{M}_i\mathbf{D}_C \text{sgn}(\mathbf{v}_i)$$

which can be computed assuming each agent can measure its own linear and angular rate  $\mathbf{v}_i$ .

Using this new control law, along with adaptation laws

$$\begin{aligned} \dot{\hat{\mathbf{d}}}_i &= -\Gamma_D\mathbf{Y}_D^T\mathbf{s} \\ \dot{\hat{\mathbf{c}}}_i &= -\Gamma_C\mathbf{Y}_C^T\mathbf{s} \end{aligned}$$

results in

$$\dot{V} = -\mathbf{s}^T(N\mathbf{K}_D + \mathbf{M}_i\mathbf{D}\mathbf{M}_i^T)\mathbf{s}$$

which again adds a positive definite term  $\mathbf{M}_i\mathbf{D}\mathbf{M}_i^T$  to  $\dot{V}$ , using the viscous friction applied at each point. We note that the Coulomb friction must be canceled exactly, due to the presence of unknown parameters (namely  $\mathbf{r}_i$ ) in the  $\text{sgn}(\cdot)$  nonlinearity.

TABLE I  
PARAMETER VALUES FOR SIMULATION IN SE(3)

parameter	value
$m$	$1.89 \times 10^4$ kg
$I_{xx}$	$1.54 \times 10^4$ kg m <sup>2</sup>
$I_{yy}$	$1.54 \times 10^4$ kg m <sup>2</sup>
$I_{zz}$	$2.37 \times 10^3$ kg m <sup>2</sup>
$\mathbf{r}_p$	[0, 0, 1.5] m
$\mathbf{r}_1$	[0, 0, 1.5] m
$\mathbf{r}_2$	[0, 0, -1.5] m
$\mathbf{r}_3$	[0.5, 0, 0] m
$\mathbf{r}_4$	[-0.5, 0, 0] m
$\mathbf{r}_5$	[0, 0.5, 0] m
$\mathbf{r}_5$	[0, -0.5, 0] m

## VII. SIMULATION RESULTS

We simulated the performance of the proposed controller for a collaborative manipulation task in SE(3) to verify the previous theoretical results. To this end, we constructed a simulation scenario where a group of  $N = 6$  space robots are controlling a large, cylindrical rigid body (such as a rocket body) to track a desired periodic trajectory  $\mathbf{q}_d(t)$ .

Table I lists the parameters used in the simulation, along with their values. The payload was a cylinder with a height of 1.5 m and radius 0.5 m. The agents were placed symmetrically along the principal axes of the body, with the measurement point  $\mathbf{r}_p$  collocated with the first agent,  $\mathbf{r}_1$ , replicating the scenario when one agent broadcasts its measurements to the group. The reference trajectory  $\mathbf{q}_d$  had desired linear and angular rates  $\dot{\mathbf{q}}_d$ , which were sums of sinusoids, e.g.,

$$v_{x,d}(t) = \sum_{i=1}^{n_f} \alpha_i \cos(\omega_i t)$$

with the number of frequencies  $n_f = 5$ . The desired pose  $\mathbf{q}_d$  started from an initial state  $\mathbf{q}_d(0)$  and forward-integrated the dynamics according to the desired velocity.

The simulations were implemented in Julia 1.1.0. All continuous dynamics were discretized and numerically integrated using the Heun’s method [46], a two-stage Runge–Kutta method, using a step size of  $h = 1 \times 10^{-2}$  s. All forward integration steps involving rotation matrices were implemented using the exponential map

$$\mathbf{R}_{t+h} = \exp(h\boldsymbol{\omega}_t^\times)\mathbf{R}_t$$

which ensures that  $\mathbf{R}$  remains in SO(3) for all time. Since the proposed adaptation laws are designed for continuous-time systems, we modify them by adding a deadband in which no adaptation will occur. For all simulations, we stop adaptation when  $\|\mathbf{s}\|_2 \leq 0.01$ . This technique is well-studied, and is commonly used to improve the robustness of adaptive controllers to unmodeled dynamics.

The physical parameters of the rigid body are given in Table I. The gain matrix  $\Gamma_o$  was set to  $0.3 \cdot \text{diag}(|\mathbf{a}_o| + 0.01)$ , where, for a vector  $\mathbf{v} \in \mathbb{R}^n$ ,  $\text{diag}(\mathbf{v})$  returns the matrix in  $\mathbb{R}^{n \times n}$  with  $\mathbf{v}$  on its diagonal, and zeros everywhere else. We place the

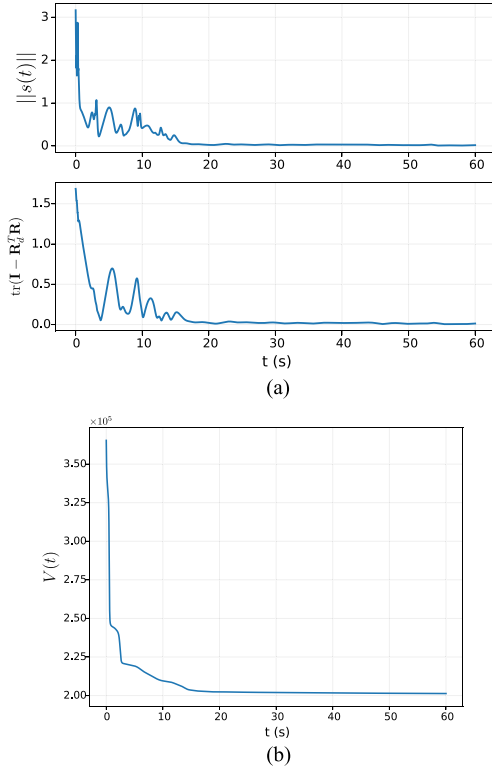


Fig. 2. Simulation results for a group of  $N = 6$  agents controlling a body in SE(3). (a) Simulated values of the composite error norm  $\|s(t)\|$  and rotation error. We notice both the composite and rotation errors converge to zero. (b) Simulated value of the Lyapunov-like function  $V(t)$ . We notice it is nonincreasing (i.e.,  $\dot{V}(t) \leq 0$ ), and converges to a constant, nonzero value, since the parameter errors do not vanish

correct parameters on the gain diagonal to scale the adaptation gains correctly, since the parameters range in magnitude from  $\times 10^{-1}$  to  $\times 10^4$ . For  $\Gamma_r$ , we use  $\times 10^{-3} * \mathbf{I}$ . Finally, we let  $\lambda = 1.5$ , and  $\mathbf{K}_D = \text{diag}(\mathbf{K}_F, \mathbf{K}_M)$ , where  $\mathbf{K}_F = 5 \times 10^4 * \mathbf{I}$ , and  $\mathbf{K}_M = 5 \times 10^3 * \mathbf{I}$ . The initial parameter estimates  $\hat{\mathbf{o}}_i$  and  $\hat{\mathbf{r}}_i$  were drawn from zero-mean multivariate Gaussian distributions with variances of  $\Sigma_o = \mathbf{I}$  and  $\Sigma_r = 2\mathbf{I}$ , respectively. Source code of the simulation is publicly available here.<sup>1</sup>

Fig. 2 plots the error signals and Lyapunov-like function value over the course of the simulation. We can see that the composite error quickly converges to zero. We also notice the rotation error converges to zero, implying  $\mathbf{R} \rightarrow \mathbf{R}_d$ , as expected. We can further see that the Lyapunov-like function  $V(t)$  is strictly decreasing (implying  $\dot{V}(t) \leq 0$ ), and converges to a constant, nonzero value. This implies that the parameter errors  $\tilde{\mathbf{o}}_i, \tilde{\mathbf{r}}_i$  do not converge to zero, which matches intuition due to the lack of a persistently exciting signal, as shown in Section VI-D.

We further compared the controller's performance against some simple baselines; namely, we studied the controller's performance when agents only adapt to the common object parameters  $\mathbf{o}_i$ , as studied in [31] and [32], and in the case of no adaptation, which reduces to the case of a simple PD controller, since all parameter values are initialized to zero. The

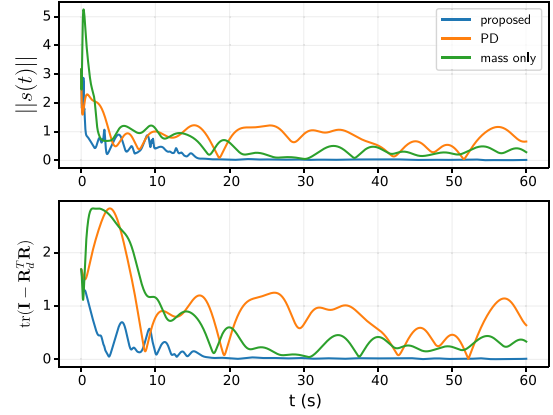


Fig. 3. Simulated baseline comparisons for the SE(3) manipulation task. In green, we plot the error signals in the case where the agents estimate and adapt to only the object parameters  $\hat{\mathbf{o}}_i$ , as in [31] and [32]. In this case, neither the composite error  $\mathbf{s}$  nor the rotation error converges to zero, even if simulated over a long horizon. The controller may also diverge for some initial conditions. In orange, the agents implement a PD controller by setting their parameter estimates  $\hat{\mathbf{o}}_i, \hat{\mathbf{r}}_i$  to zero for all time. We note the error does not converge to zero, and the tracking performance is generally poor.

error signals for both cases are plotted in Fig. 3. In the first case ( $\Gamma_r = \mathbf{0}$ ), we can see that that controller performance greatly suffers due to the inaccurate moment arm estimates. Specifically, in this case, the composite error  $\mathbf{s} \not\rightarrow \mathbf{0}$ ; even over a very long time horizon, we found the error never vanishes. We also found that simply doubling the variance of the distribution from which the moment arm initializations  $\hat{\mathbf{r}}_i$  were sampled, i.e., sampling  $\hat{\mathbf{r}}_i \sim \mathcal{N}(\mathbf{0}, 2\sqrt{2}\mathbf{I})$ , would usually cause the controller to diverge.

This behavior is as expected, since when the moment arm estimates are poor, there is a large, unmodeled coupling between the rotational and translational dynamics, which controller cannot compensate. In the second case, we plot the performance, which results from only using the PD term in the control law (14), to demonstrate the value of the proposed adaptation routine. We can clearly see that the PD term alone is unable to send the composite error  $\mathbf{s}$  to zero; thus, the proposed adaptation is essential to ensure that the tracking error vanishes.

We were also interested in how the controller performs in the case of single-agent failure (i.e., if one or more agents are deactivated at some time  $t_d$  during the manipulation task). We note that this is an instantaneous shift in the system parameters (since new  $\alpha_i$  must be chosen such that the constants still sum to one). This is, in turn, equivalent to restarting the controller with new system parameters  $\mathbf{o}_i$ , and a different initial condition; thus, the boundedness and convergence results still hold. Fig. 4 plots the system behavior in the case that three agents (half the manipulation team) are deactivated at  $t_d = 30$  s. As expected, when the agents are dropped, we see a discontinuity in the Lyapunov-like function  $V(t)$ , due to the parameter change, and a slight jump in the composite error  $\mathbf{s}$ . However, the remaining three agents are still able to regain tracking, as expected.

Finally, as shown recently in [47] and [48], the parameter error terms in the Lyapunov-like function (11),  $\tilde{\mathbf{a}}_i^T \Gamma_i^{-1} \tilde{\mathbf{a}}_i$  can be generalized by instead using the Bregman divergence  $d_{\psi}(\Gamma_i \mathbf{a}_i \parallel$

<sup>1</sup>[https://github.com/pculbertson/hamilton\\_ac](https://github.com/pculbertson/hamilton_ac)

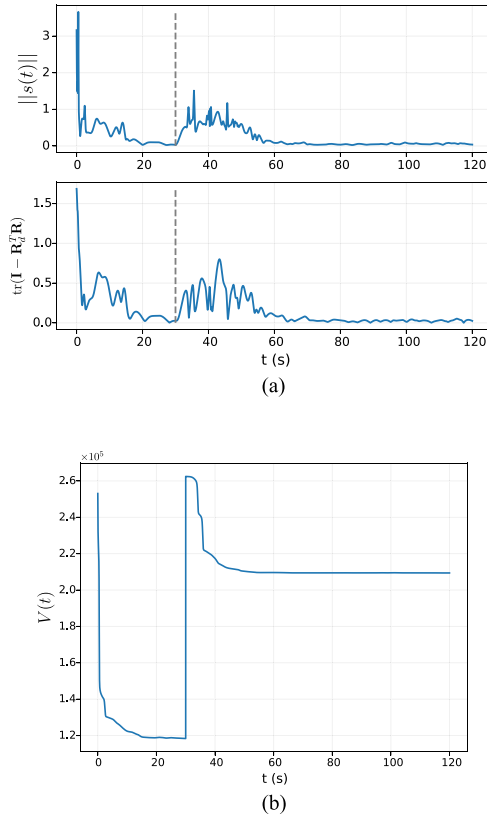


Fig. 4. Simulation results for SE(3) task, where 3 of the  $N = 6$  agents are deactivated at time  $t = 30$ . (a) Simulated values of the composite error norm  $\|s(t)\|$  and rotation error. The “drop time” is shown with a dashed grey line. After the drop, the remaining agents return to tracking the reference trajectory. (b) Simulated value of the Lyapunov-like function  $V(t)$ . We note the function is nonincreasing, except for the discontinuity at  $t = 30$ , which occurs due to the instantaneous switch of parameter values

$\Gamma_i \hat{\mathbf{a}}_i$ ), where, for some strictly convex function  $\psi$ , we write the Bregman divergence as

$$d_\psi(\mathbf{x} \parallel \mathbf{y}) = \psi(\mathbf{y}) - \psi(\mathbf{x}) - (\mathbf{y} - \mathbf{x})^T \nabla \psi(\mathbf{x}).$$

If we use the adaptation law

$$\hat{\mathbf{a}}_i = -\Gamma_i^{-1} (\nabla^2 \psi(\Gamma_i \hat{\mathbf{a}}))^{-1} \Gamma_i^{-1} \mathbf{Y}_i^T \mathbf{s}$$

we can easily reproduce the boundedness and convergence results shown previously. In fact, one can show the usual quadratic penalty is simply a Bregman divergence for  $\psi(\mathbf{x}) = \frac{1}{2} \|\Gamma^{-\frac{1}{2}} \mathbf{x}\|_2^2$ .

As a cursory study of this behavior, we substituted the usual quadratic penalty for a Bregman divergence with  $\psi(\mathbf{x}) = \|\mathbf{x}\|_1$ , the  $\ell_1$  norm. In [49], the authors show the  $\ell_1$  norm leads to sparse solutions, meaning the parameter estimates converge to vectors, which can achieve tracking with few nonzero values.

For this experiment, we increased the number of agents to  $N = 100$  to exaggerate the overparameterization of the system. While the tracking performance and transient error were quite similar for both regularization strategies, they give rise to quite different estimates of the object parameters. Fig. 6 plots histograms of the estimated parameters under both the  $\ell_1$  norm and  $\ell_2$  norm divergences. Here, the parameters are scaled so all values are on the same order. We can indeed see the  $\ell_1$  solution

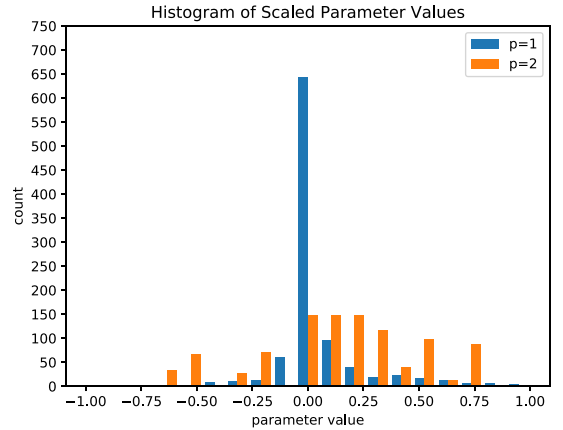


Fig. 5. Histogram of estimated parameter values for  $\ell_1$  and  $\ell_2$  regularization. When using the  $\ell_1$  norm (blue), we see the resulting parameter estimate is quite sparse, with a large spike centered at 0. When using the  $\ell_2$  norm (orange), we instead see the distribution of parameter estimates is less peaked, but takes on fewer large values.

is much sparser, with many values equal to zero, and with more outliers than the  $\ell_2$  case.

Fig. 6 shows a visualization of the estimated parameters. We can again see that the  $\ell_1$  solution is quite sparse, with many entries equal to zero, in addition to the few outliers, which are have relatively large values. In contrast, the  $\ell_2$  solution shows a “banded” structure that implies agreement between agents, where all agents arrive at parameter estimates that are somewhat small in magnitude, and roughly equal across agents.

Again, while these experiments are quite cursory, they demonstrate that, due to the overparameterization of the system, we can design various forms of regularization to express preferences about which parameter estimates, among those that achieve tracking, are desirable. The question of how various regularization terms affect the system’s robustness to noise, or the overall control effort, could prove interesting directions for future research.

## VIII. EXPERIMENTAL RESULTS

We further studied the performance of the proposed controller experimentally, using a group of  $N = 6$  ground robots performing a manipulation task in the plane. Fig. 7 shows one of the robots used in this experiment, a Oujjabot [50], which is a holonomic ground robot designed specifically for collaborative manipulation. The robots were equipped with current sensors on each motor, which allows them to estimate and control the wrench they are applying to a payload. Specifically, the robots used a low-level PID controller to generate motor torques, based on the desired wrench set by the higher level adaptive controller, with feedback from the current sensors. They are also equipped with a Raspberry Pi 3B+, which allows them to perform onboard computation, as well as receive state measurements via robot operating system (ROS). The robots did not use any other communication capabilities.

Fig. 8 shows the payload which the robots manipulated, which was an asymmetric wood piece that weighed roughly 5.5 kg, which was too heavy for any individual agent to move.

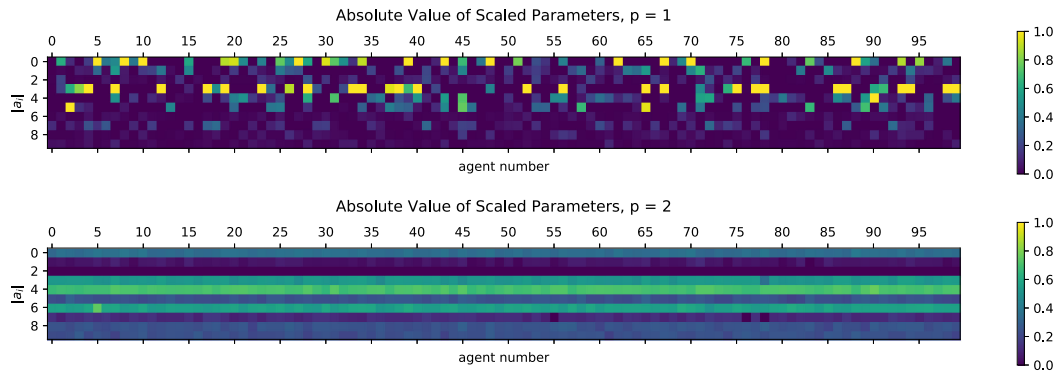


Fig. 6. Visualization of estimated parameters for each regularization term. Top: Visualization of the estimated parameters when using the Bregman divergence for the  $\ell_1$  norm. In this case, we can see this regularization does indeed promote sparsity, with many entries being equal or close to zero, with a few outliers taking on larger absolute values. Bottom: Visualization of the estimated parameters when using the  $\ell_2$  norm. In this case, the parameters take on much smaller values, but are roughly equal across all agents.

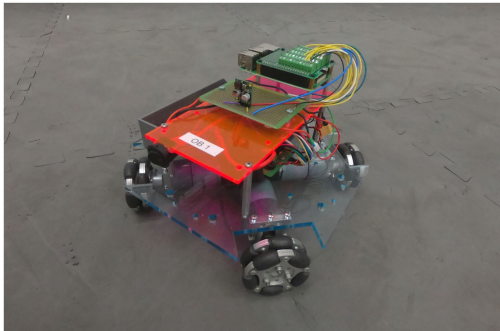


Fig. 7. Photo of a Oujibot, the holonomic ground robot used to conduct the experiments presented in this work. The robot is equipped with current sensors on its motors, which allows it to close a feedback loop to provide a desired force and torque to an external payload.

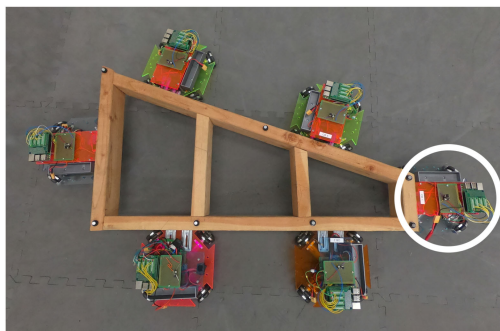


Fig. 8. Experimental setup for planar manipulation task. A team of  $N = 6$  ground robots manipulate a shared wood payload. The measurement point for the payload coincides with one of the agents, circled in white, and is located far from the geometric center of the body, and its center of mass.

The robots were rigidly attached to the piece, which remained roughly 4 cm from the ground. As seen in the figure, the measurement point  $\mathbf{r}_p$  coincided with one of the robots, and was located quite far from the center of mass, resulting in long moment arms  $\mathbf{r}_i$  as well as nonnegligible coupling between the rotational and translational dynamics. The robots received position measurements from this point using Optitrack, a motion capture system; the velocity measurements were computed by numerically differentiating the position measurements. All

signals were passed through a first-order filter to reduce the effects of measurement noise.

Fig. 9 plots the results of the experimental study. The desired trajectory  $\mathbf{q}_d$  of the measurement point corresponded to a “Figure-8” in the plane, with the desired rotation  $\theta_d$  oscillating between  $-\frac{\pi}{4}$  and  $\frac{\pi}{4}$  at successive waypoints. The experiment included 25 trials of the manipulation task, with all initial parameter estimates set to zero. Fig. 9(a) plots the composite error norm, averaged over the 25 trials, with the deadband of 0.2, under which adaptation was stopped, plotted as a dotted line. We can see that the controller converges to be within the deadband in about 45 s. Fig. 9(c) shows the ground track of all trials, with the reference shown in orange, and baseline (PD) response shown in green. We note the agents typically overshoot the first turn (top right), and then converge to reasonable tracking, although it proved difficult to achieve perfect tracking through all turns. Fig. 9(d) plots the heading of the body, averaged over all trials; shaded is the maximum and minimum heading value at each time. We again note the controller achieves reasonable tracking after a short transient, which improves over the trial; the PD controller alone gave nearly no response.

We note this experiment introduced considerable sources of modeling error, which were not parameterized by the adaptive controller. Specifically, the wrenches computed by the high-level controller were only setpoints for a low-level PID controller. Furthermore, while the control law included terms for both viscous and Coulomb friction, as outlined in Section VI-E, this of course did not capture the complex frictional relationship between the robot wheels and foam floor of the experimental space, which depended on wheel speed, direction, etc. Finally, all control signals were subject to time delays, discretization error, and measurement noise. We found our controllers (which were implemented as robot operating system (ROS) nodes) ran at a frequency of 20 Hz.

## IX. CONCLUDING REMARKS

This work has focused on the problem of collaborative manipulation without knowledge of payload parameters or centralized control. To this end, we proposed a feedback control

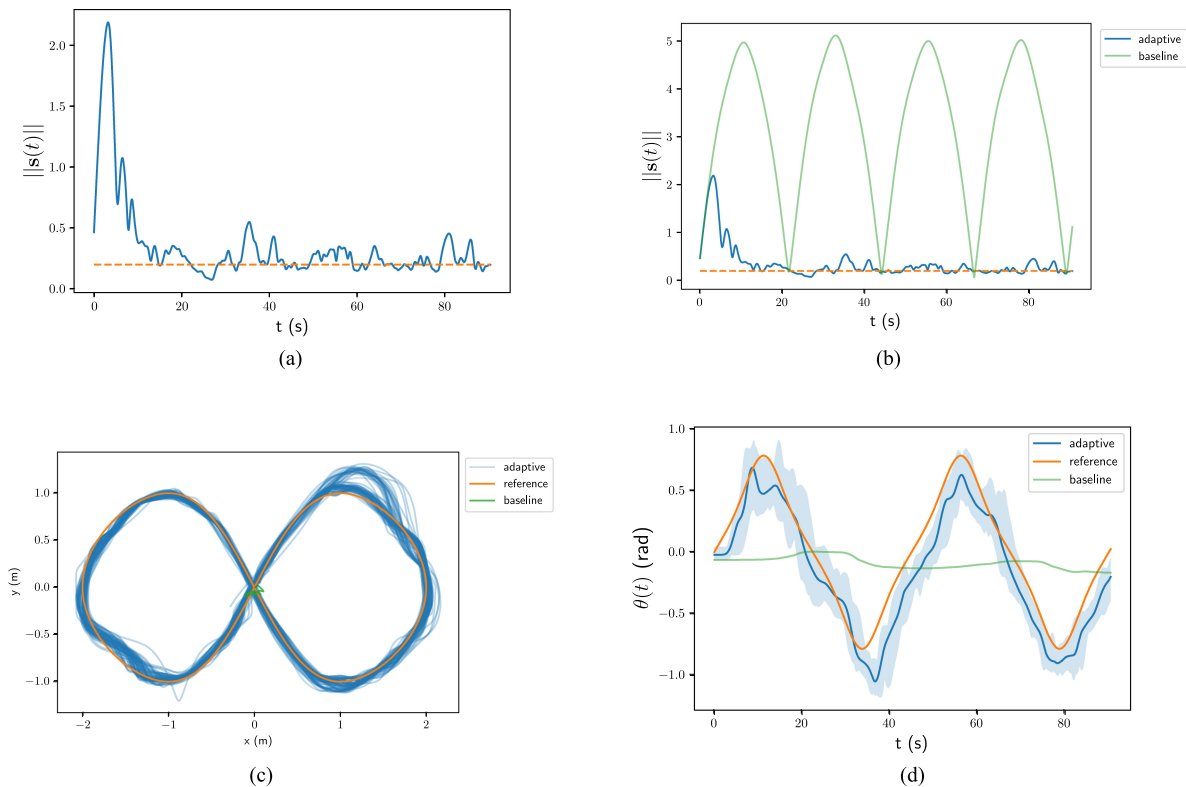


Fig. 9. Experimental results for a group of  $N = 6$  ground robots performing a planar manipulation task. (a) Measured value of the composite error, averaged across 25 trials. (b) Composite error, plotted with the values measured when only using the PD term of the control law. (c) Overhead view of the measurement point trajectory for all trials. (d) Plot of average payload angle (blue), with shading between maximum and minimum values across all trials. Plotted with reference trajectory (orange), and PD baseline response (green)

architecture on  $SE(3)$ , together with a distributed adaptation law, which enabled the agents to gradually adapt to the unknown object parameters to track a desired trajectory. We further provided a Lyapunov-like proof of the asymptotic convergence of the system to the reference trajectory under our proposed controller, and studied its performance for practical manipulation tasks using both numerical simulations and hardware experiments.

There exist numerous directions for future work. An important open question is how to incorporate actuation constraints, such as saturation, into the proposed control framework. In fact, tight actuation constraints typically are a strong motivation for collaborative manipulation, but in this work, we assume the agents may apply any desired force and torque to the object. This is because, in general, it is quite difficult to incorporate saturation into adaptive control design. While our hardware experiments demonstrate that the proposed controller can achieve a good performance despite these unmodeled actuation constraints, handling these theoretically remains an important open problem.

Another interesting direction is to understand more clearly the connections between this work, and other distributed control architectures such as quorum sensing or the control of networked systems. As mentioned previously, studying the effect of different regularization strategies, or of using higher order adaptation laws, on the overall system performance could also prove valuable. Furthermore, while the proposed controller is

robust to slow changes in the object's mass and geometric properties, another interesting direction would be to study how and if the proposed architecture can be extended to the manipulation of articulated or flexible bodies. Finally, while we assume the agents travel through free space, we are quite interested in enabling distributed manipulation in the presence of obstacles, and studying how agents can coordinate themselves through object motion to avoid obstacles they cannot directly observe.

## APPENDIX

### A. Proofs of Matrix Properties

Here, we demonstrate some properties of  $\mathbf{H}$ ,  $\mathbf{C}$ , and their derivatives, which we use to prove Theorem 2. These properties are well-known for general Lagrangian equations of motion [41, Ch. 9]; here, we prove them for our choices of  $\mathbf{H}$ ,  $\mathbf{C}$ .

*Lemma 1:*  $\mathbf{H}(\mathbf{q})$ , as given in (6), is positive definite.

*Proof:* For a block matrix of the form

$$\mathbf{M} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C} \end{bmatrix}$$

using the properties of the Schur complement [51], we can say  $\mathbf{M} \succ 0$  if and only if  $\mathbf{A} \succ 0$  and  $\mathcal{S} := \mathbf{C} - \mathbf{B}^T \mathbf{A}^{-1} \mathbf{B} \succ 0$ . Thus, since  $m\mathbf{I} \succ 0$  by inspection,  $\mathbf{H} \succ 0$  if and only if  $\mathcal{S} = \mathbf{R}\mathbf{I}_p\mathbf{R}^T + m(\mathbf{R}\mathbf{r}_p)^\times(\mathbf{R}\mathbf{r}_p)^\times \succ 0$ .

We can further use the fact that for  $\mathbf{v} \in \mathbb{R}^3$

$$\mathbf{v}^\times \mathbf{v}^\times = \mathbf{v}\mathbf{v}^T - (\mathbf{v}^T \mathbf{v})\mathbf{I}$$

to write

$$\begin{aligned} \mathcal{S} &= \mathbf{R} (\mathbf{I}_{\text{cm}} \pm m(\mathbf{r}_p^T \mathbf{r}_p - \mathbf{r}_p \mathbf{r}_p^T)) \mathbf{R}^T \\ &= \mathbf{R} \mathbf{I}_{\text{cm}} \mathbf{R}^T \end{aligned}$$

so we have  $\mathcal{S} \succ 0$ , since  $\mathbf{I}_{\text{cm}} \succ 0$  for bodies of finite size. Thus, we conclude  $\dot{\mathbf{H}} \succ 0$ . ■

**Lemma 2:**  $\dot{\mathbf{H}} - 2\mathbf{C}$  is skew symmetric.

**Proof.**

$\dot{\mathbf{H}}$  is given by

$$\dot{\mathbf{H}} = \begin{bmatrix} \mathbf{0}_3 & \mathbf{A} - \mathbf{A}^T \\ \mathbf{A}^T - \mathbf{A} \omega^\times \mathbf{R} \mathbf{I}_p \mathbf{R}^T - \mathbf{R} \mathbf{I}_p \mathbf{R} \omega^\times & \end{bmatrix}$$

where  $\mathbf{A} = m\omega^\times (\mathbf{R} \mathbf{r}_p)^\times$ , and using the fact  $\dot{\mathbf{R}} = \omega^\times \mathbf{R}$ . Furthermore

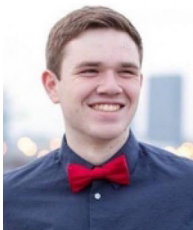
$$\dot{\mathbf{H}} - 2\mathbf{C} = \begin{bmatrix} \mathbf{0}_3 & -\mathbf{B} \\ \mathbf{B} & \mathbf{D} \end{bmatrix}$$

where  $\mathbf{B} = \mathbf{B}^T = \mathbf{A} + \mathbf{A}^T$  and  $\mathbf{D} = -\omega^\times \mathbf{R} \mathbf{I}_p \mathbf{R}^T - \mathbf{R} \mathbf{I}_p \mathbf{R} \omega^\times - m((\mathbf{R} \mathbf{r}_p)^\times \dot{\mathbf{x}})^\times$ . We can see  $\dot{\mathbf{H}} - 2\mathbf{C}$  is skew-symmetric, since  $\mathbf{D}$  is skew-symmetric, and the off-diagonal blocks are symmetric and of opposite sign. ■

## REFERENCES

- [1] O. Khatib, K. Yokoi, O. Brock, K. Chang, and A. Casal, "Robots in human environments: Basic autonomous capabilities," *Int. J. Robot. Res.*, vol. 18, no. 7, pp. 684–696, 1999.
- [2] O. Khatib *et al.*, "Force strategies for cooperative tasks in multiple mobile manipulation systems," in *Robotics Research*, G. Giralt and G. Hirzinger, eds. London, U.K.: Springer, 1996, pp. 333–342.
- [3] O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, and A. Casal, "Coordination and decentralized cooperation of multiple mobile manipulators," *J. Robot. Syst.*, vol. 13, no. 11, pp. 755–764, Nov. 1996.
- [4] J. Fink, M. A. Hsieh, and V. Kumar, "Multi-robot manipulation via caging in environments with obstacles," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2008, pp. 1471–1476.
- [5] D. Rus, B. Donald, and J. Jennings, "Moving furniture with teams of autonomous robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. Human Robot Interact. Cooperative Robots*, 1995, pp. 235–242.
- [6] M. Corah and N. Michael, "Active estimation of mass properties for safe cooperative lifting," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 4582–4587.
- [7] A. Franchi, A. Petitti, and A. Rizzo, "Distributed estimation of state and parameters in multiagent cooperative load manipulation," *IEEE Control Netw. Syst.*, vol. 6, no. 2, pp. 690–701, Jun. 2019.
- [8] H. Lee and H. J. Kim, "Constraint-based cooperative control of multiple aerial manipulators for handling an unknown payload," *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 2780–2790, Dec. 2017.
- [9] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1520–1533, Sep. 2004.
- [10] C. Alford and S. Belyeu, "Coordinated control of two robot arms," in *Proc. IEEE Int. Conf. Robot. Autom.*, Mar. 1984, vol. 1, pp. 468–473.
- [11] O. Khatib, "Object manipulation in a multi-effector robot system," in *Proc. 4th Int. Symp. Robot. Res.*, 1988, pp. 137–144.
- [12] J. T. Wen and K. Kreutz-Delgado, "Motion and force control of multiple robotic manipulators," *Automatica*, vol. 28, no. 4, pp. 729–743, 1992.
- [13] R. C. Bonitz and T. C. Hsia, "Internal force-based impedance control for cooperating manipulators," *IEEE Trans. Robot. Autom.*, vol. 12, no. 1, pp. 78–89, Feb. 1996.
- [14] F. Caccavale, P. Chiacchio, A. Marino, and L. Villani, "Six-dof impedance control of dual-arm cooperative manipulators," *IEEE/ASME Trans. Mechatronics*, vol. 13, no. 5, pp. 576–586, Oct. 2008.
- [15] S. Erhart, D. Sieber, and S. Hirche, "An impedance-based control architecture for multi-robot cooperative dual-arm mobile manipulation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nov. 2013, pp. 315–322.
- [16] S. Erhart and S. Hirche, "Internal force analysis and load distribution for cooperative multi-robot manipulation," *IEEE Trans. Robot.*, vol. 31, no. 5, pp. 1238–1243, Oct. 2015.
- [17] K. Böhlinger, R. Brown, B. Donald, J. Jennings, and D. Rus, "Distributed robotic manipulation: Experiments in minimalism," in *Exp. Robot. IV*, O. Khatib and J. K. Salisbury, eds. Berlin, Germany: Springer, 1997, pp. 11–25.
- [18] P. Song and V. Kumar, "A potential field based approach to multi-robot manipulation," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2002, vol. 2, pp. 1217–1222.
- [19] S. G. Faal, S. T. Kalat, and C. D. Onal, "Towards collective manipulation without inter-agent communication," in *Proc. 31st Annu. ACM Symp. Appl. Comput.*, 2016, pp. 275–280.
- [20] D. Mellinger, M. Shomin, N. Michael, and V. Kumar, *Cooperative Grasping and Transport Using Multiple Quadrotors*. Berlin, Germany: Springer, 2013, pp. 545–558.
- [21] K. M. Lynch, "Controllability of a planar body with unilateral thrusters," *IEEE Trans. Autom. Control*, vol. 44, no. 6, pp. 1206–1211, Jun. 1999.
- [22] K. Sreenath and V. Kumar, "Dynamics, control and planning for cooperative manipulation of payloads suspended by cables from multiple quadrotor robots," in *Proc. Robot.: Sci. Syst. IX. Robot.: Sci. Syst. Found.*, Jun. 2013.
- [23] S. Tang, K. Sreenath, and V. Kumar, "Multi-robot trajectory generation for an aerial payload transport system," in *Robotics Research*, N. M. Amato, G. Hager, S. Thomas, and M. Torres-Torriti, eds. Cham, Switzerland: Springer International Publishing, 2020, pp. 1055–1071.
- [24] N. Michael, J. Fink, and V. Kumar, "Cooperative manipulation and transportation with aerial robots," *Auton. Robots*, vol. 30, no. 1, pp. 73–86, Jan. 2011.
- [25] B. E. Jackson, T. A. Howell, K. Shah, M. Schwager, and Z. Manchester, "Scalable cooperative transport of cable-suspended loads with UAVs using distributed trajectory optimization," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 3368–3374, Apr. 2020.
- [26] Z. Wang and M. Schwager, "Force-amplifying n-robot transport system (force-ants) for cooperative planar manipulation without communication," *Int. J. Robot. Res.*, vol. 35, no. 13, pp. 1564–1586, 2016.
- [27] Z. Wang, S. Singh, M. Pavone, and M. Schwager, "Cooperative object transport in 3 d with multiple quadrotors using no peer communication," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2018, pp. 1064–1071.
- [28] A. Petitti, A. Franchi, D. Di Paola, and A. Rizzo, "Decentralized motion control for cooperative manipulation with a team of networked mobile manipulators," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2016, pp. 441–446.
- [29] Y.-R. Hu, A. A. Goldenberg, and C. Zhou, "Motion and force control of coordinated robots during constrained motion tasks," *Int. J. Robot. Res.*, vol. 14, no. 4, pp. 351–365, 1995.
- [30] Z. Li, S. S. Ge, and Z. Wang, "Robust adaptive control of coordinated multiple mobile manipulators," in *Proc. IEEE Int. Conf. Control Appl.*, Oct. 2007, pp. 71–76.
- [31] Y.-H. Liu and S. Arimoto, "Decentralized adaptive and nonadaptive position/force controllers for redundant manipulators in cooperations," *Int. J. Robot. Res.*, vol. 17, no. 3, pp. 232–247, 1998.
- [32] C. K. Verginis, M. Mastellaro, and D. V. Dimarogonas, "Robust quaternion-based cooperative manipulation without force/torque information," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 1754–1759, 2017.
- [33] D. Prattichizzo and C. J. Trinkle, *Grasping*. Berlin, Germany: Springer, 2008, pp. 671–700.
- [34] P. Culbertson and M. Schwager, "Decentralized adaptive control for collaborative manipulation," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2018, pp. 278–285.
- [35] J.-J. E. Slotine and W. Li, "On the adaptive control of robot manipulators," *Int. J. Robot. Res.*, vol. 6, no. 3, pp. 49–59, 1987.
- [36] N. Tabareau, J.-J. Slotine, and Q.-C. Pham, "How synchronization protects from noise," *PLoS Comput. Biol.*, vol. 6, no. 1, Jan. 2010, Art. no. e1000637.
- [37] G. Russo and J. J. E. Slotine, "Global convergence of quorum-sensing networks," *Phys. Rev. E*, vol. 82, no. 4, Oct. 2010, Art. no. 041919.
- [38] C. M. Waters and B. L. Bassler, "Quorum sensing: Cell-to-cell communication in bacteria," *Annu. Rev. Cell Develop. Biol.*, vol. 21, no. 1, pp. 319–346, 2005. PMID: 16212498.
- [39] S. P. Bhat and D. S. Bernstein, "A topological obstruction to global asymptotic stabilization of rotational motion and the unwinding phenomenon," in *Proc. Amer. Control Conf.*, 1998, vol. 5, pp. 2785–2789.

- [40] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. Boca Raton, FL, USA: CRC Press, Dec. 2017.
- [41] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1991.
- [42] G. C. Gómez Cortés, F. Castaños, and J. Dávila, "Sliding motions on  $so(3)$ , sliding subgroups," in *Proc. IEEE 58th Conf. Decis. Control*, 2019, pp. 6953–6958.
- [43] M. W. Spong, R. Ortega, and R. Kelly, Comments on "Adaptive manipulator control: A case study" *IEEE Trans. Autom. Control*, vol. 35, no. 6, pp. 761–762, Jun. 1990.
- [44] J.-J. E. Slotine and W. Li, "Theoretical issues in adaptive manipulator control," in *Proc. 5th Yale Workshop Appl. Adaptive Syst. Theory*, 1987, pp. 252–258.
- [45] P. M. Wensing and J. Slotine, "Cooperative adaptive control for cloud-based robotics," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 6401–6408.
- [46] J. Stoer and R. Bulirsch, *Introduction to Numerical Analysis*. New York, NY, USA: Springer, 1980.
- [47] T. Lee, J. Kwon, and F. C. Park, "A natural adaptive control law for robot manipulators," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 1–9.
- [48] P. M. Wensing and J.-J. E. Slotine, "Beyond convexity—Contraction and global convergence of gradient descent," *Plos One*, vol. 15, no. 8, 2020, Art. no. e0236661.
- [49] N. M. Boffi and J.-J. E. Slotine, "Implicit regularization and momentum algorithms in nonlinearly parameterized adaptive control and prediction," *Neural Comput.*, vol. 33, no. 3, pp. 590–673, Mar. 2021.
- [50] Z. Wang, G. Yang, X. Su, and M. Schwager, "OuijaBots: Omnidirectional robots for cooperative object transport with rotation control using no communication," in *Distributed Autonomous Robotic Systems*. Cham, Switzerland: Springer International Publishing, 2018, pp. 117–131.
- [51] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.



**Preston Culbertson** (Student Member, IEEE) received the B.S. degree in mechanical engineering in 2016 from the Georgia Institute of Technology, Atlanta, GA, USA, in 2016, and the M.S. degree in mechanical engineering in 2020 from Stanford University, Stanford, CA, USA, where he is currently working toward the Ph.D. degree in mechanical engineering.

His research interests include multirobot manipulation and grasping, distributed optimization and control, and learning and adaptation in robotic systems,

especially among groups of interacting agents.



**Jean-Jacques Slotine** received the Ph.D. degree in estimation and control from the Department of Aeronautics and Astronautics from the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, in 1983.

After working with Bell Labs, Computer Research Department, he joined the MIT faculty in 1984. He is currently a Professor of Mechanical Engineering and Information Sciences, a Professor of Brain and Cognitive Sciences, and the Director of the Nonlinear Systems Laboratory, MIT. His research focuses on developing rigorous but practical tools for nonlinear systems analysis and control. These have included key advances and experimental demonstrations in the contexts of sliding control, adaptive nonlinear control, adaptive robotics, machine learning, and contraction analysis of nonlinear dynamical systems. He is the coauthor of two graduate textbooks, *Robot Analysis and Control* (Wiley, 1986), and *Applied Nonlinear Control* (Prentice-Hall, 1991).

Dr. Slotine was a member of the French National Science Council from 1997 to 2002 and of Singapore's A\*STAR SigN Advisory Board from 2007 to 2010. He currently is a member of the Scientific Advisory Board of the Italian Institute of Technology and a Distinguished Visiting Faculty at Google AI. He is the recipient of the 2016 Oldenburger Award.



**Mac Schwager** (Member, IEEE) received the B.S. degree from Stanford University, Stanford, CA, USA, in 2000, the M.S. and Ph.D. degrees from the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, in 2009 and 2005, respectively, all in mechanical engineering.

He is currently an Assistant Professor with the Aeronautics and Astronautics Department, Stanford University. He was a Postdoctoral Researcher working jointly with the GRASP Lab, University of Pennsylvania, Philadelphia, PA, USA, and CSAIL, MIT, Cambridge, MA, USA, from 2010 to 2012, and was an Assistant Professor with Boston University, Boston, MA, USA, from 2012 to 2015. His research interests are in distributed algorithms for control, perception, and learning in groups of robots, and models of cooperation and competition in groups of engineered and natural agents.

Dr. Schwager was the recipient of the NSF CAREER award in 2014, the DARPA YFA in 2018, and a Google Faculty Research Award in 2018, and the IROS Toshio Fukuda Young Professional Award in 2019.