Decentralized Sliding Mode Control for Autonomous Collective Transport by Multi-Robot Systems

Hamed Farivarnejad, Sean Wilson, and Spring Berman

Abstract-We present a decentralized sliding mode control strategy for collective payload transport by a team of robots. The controllers only require robots' measurements of their own heading and velocity, and the only information provided to the robots is the target speed and direction of transport. The control strategy does not rely on inter-robot communication, prior information about the load dynamics and geometry, or knowledge of the transport team size and configuration. We initially develop the controllers for point-mass robots that are rigidly attached to a load and prove the stability of the system, showing that the speed and direction of the transported load will converge to the desired values in finite time. We also modify the controllers for implementation on differential-drive mobile robots. We demonstrate the effectiveness of the proposed controllers through simulations with point-mass robots, 3D physics simulations with realistic dynamics, and experiments with small mobile robots equipped with manipulators.

I. INTRODUCTION

One potential application of autonomous robotic swarms is cooperative manipulation in unstructured, uncertain environments that are inaccessible or hazardous to humans. This type of task can arise in scenarios such as construction, assembly in space and underwater, search-and-rescue operations, and disaster response. This application will require the development of robot control strategies that rely on minimal information and are robust to uncertainties in the payload dynamics and to external disturbances. Toward this end, we present, analyze, and implement a novel decentralized control scheme based on a sliding mode control approach. These types of controllers provide robust control of nonlinear systems and only require bounds on uncertainties in the dynamics instead of their precise characterization.

In recent years, various control methods have been proposed for collective transport tasks in which there is no interrobot communication, some of which are leader-follower strategies. For instance, in [2] and [3], a consensus-based approach is presented in which the leader is more powerful than the followers and is provided with a predefined path to the goal, and the followers, which do not know anything about the leader's intention, can effectively attain a consensus on the magnitude and the direction of the force they have to apply to the load. Also, in [4], a leader robot applies a force to move the load over a predefined path, and followers can estimate the direction of the object movement using



Fig. 1. Simulated Pheeno robots [1] performing a collective transport task.

force sensing at the attachment point and apply their forces along this estimated direction in order to assist the leader. Considering the load as the leader is another approach presented in [5], where the followers (the transporting robots) use a path planning approach to preserve their initial position and orientation with respect to the virtual leader (load) during the transport. In other works, all robots in the transport team are assumed to be identical. In [6], a decentralized approach is proposed for cooperative transport when the load is significantly larger than robots, and robots push the load to the goal only if their line of sight to the goal is occluded by the object. In [7], a combination of four controllers are presented by which robots can estimate the centroid of the load in order to rotate and transport it over certain marked points along the way that can be recognized by a guide robot. Transporting a flexible payload is considered in [8] and [9], where the reaction force between the robot and the payload is modeled as the gradient of a nonlinear potential that describes the load deformation. Inspired by ants, group food retrieval is studied in [10], and a force sensing mechanism is fabricated to develop a hybrid dynamical model that can replicate collective behavior observed in ants. In [11], it is supposed that all the robots know the target direction to the goal, and a simple control law, which uses just the velocity of the attachment point, is developed to calculate the force that a robot has to apply to the load. In a similar scenario, decentralized PID controllers were used in [1] for collective transport by three small mobile robots.

In the present work, we consider scenarios that are similar to those in [1] and [11] and design a control approach whose stability can be proven to drive the transport system to a goal position in a target direction. Although sliding mode control has previously been used for cooperative manipulation in [12], [13], [14], [15], these strategies require predefined

This research was supported by DARPA Young Faculty Award D14AP00054.

Hamed Farivarnejad, Sean Wilson, and Spring Berman are with the School for Engineering of Matter, Transport and Energy, Arizona State University (ASU), Tempe, AZ 85281, USA {Hamed.Farivarnejad, Sean.T.Wilson, Spring.Berman}@asu.edu

trajectories for each robot and/or for the payload. In contrast, the control strategy proposed here only requires local robot measurements of their own velocity and heading, and it does not rely on information about the environment, load, or transport team.

II. PROBLEM STATEMENT

We consider a team of identical autonomous ground robots, each equipped with a manipulator arm, that are arranged on a planar surface in an arbitrary configuration around a payload. The robots are all grasping the load and holding it above the ground (as in Fig. 1). We assume that each robot can measure its speed and heading. The robots do not have global localization or communication capabilities, and they lack information about the payload dynamics, the number of robots in the transport team and their distribution around the payload, and the layout of the environment.

Our objective is to design decentralized controllers that will drive the team of robots to collectively transport the load at a desired speed along a straight path in a target direction. We assume that each robot knows the target direction, although they are not assigned predefined trajectories. To enable the robots to act autonomously during transport, we do not assign them reference speed profiles that would require the presence of a global supervisor with knowledge about their positions with respect to the goal and their distribution around the payload. Instead, the controllers must depend only on the minimal information that is available to the robots and should be robust to the uncertainties in the highly nonlinear dynamics of the manipulated payload.

III. DYNAMICAL MODEL

We consider a load that is transported in the plane by a group of N robots, each of which is modeled as a point-mass agent. The position of robot i at time t in an inertial reference frame is given by $\mathbf{X}_i(t) \in \mathbb{R}^2$. The robot's actuating force is denoted by $\mathbf{u}_i \in \mathbb{R}^2$, and the reaction force exerted by the load on the robot is $\mathbf{F}_i \in \mathbb{R}^2$. Given that robot i has mass m_i , the dynamics of the robot are:

$$m_i \ddot{\mathbf{X}}_i = \mathbf{u}_i - \mathbf{F}_i \ . \tag{1}$$

In order to develop a sliding mode controller for robot i, we must be able to write the robot's dynamics in the form

$$\ddot{\mathbf{X}}_i = \mathbf{h} + \mathbf{G}\mathbf{u}_i , \qquad (2)$$

in which G is an input matrix that is a function of the load dynamics, and h is a nonlinear term that describes the effects of both the load dynamics and the forces applied by the other robots. The sliding mode controller will only require bounds on this nonlinear term, not a precise characterization. In the remainder of this section, we show that Equation (1) can be put into the form Equation (2).

The notation for our dynamical model of collective transport is shown in Figure 2. We define an inertial coordinate frame I and a local coordinate frame B that is fixed to the load. The matrix \mathbf{R}_B^I is the rotation matrix from coordinate frame B to coordinate frame I. We define $\ddot{\mathbf{X}}_o^B$ and \mathbf{r}_i^B ,



Fig. 2. Schematic representation of a collective transport task with four robots and the associated coordinate systems.

both expressed in coordinate frame B, as the acceleration of the load's center of gravity (CG) and the vector from the load's CG to the attachment point of robot *i*, respectively. We denote the load's orientation in frame I by θ , its angular velocity by ω , and its angular acceleration by α . We now recall that the cross product of any two non-zero vectors **a** and **b** can be expressed as $\mathbf{a} \times \mathbf{b} = \hat{\mathbf{a}}\mathbf{b}$, where $\hat{\mathbf{a}}$ is a skew-symmetric matrix. Using this notation, we denote the skew-symmetric matrix representations of ω and α by $\hat{\omega}$ and $\hat{\alpha}$, respectively.

Since we assume that each robot is rigidly attached to the load, the acceleration of robot i can be written in terms of the load's angular velocity and angular acceleration as follows:

$$\ddot{\mathbf{X}}_{i} = \mathbf{R}_{B}^{I} (\ddot{\mathbf{X}}_{o}^{B} + \hat{\alpha} \mathbf{r}_{i}^{B} + \hat{\omega} (\hat{\omega} \mathbf{r}_{i}^{B})) .$$
(3)

Noting that

$$\hat{\alpha}\mathbf{r}_i^B = -\hat{\mathbf{r}}_i^B \alpha = (\hat{\mathbf{r}}_i^B)^T \alpha \tag{4}$$

and that $\alpha = \ddot{\theta}$, we can rewrite Equation (3) as:

$$\ddot{\mathbf{X}}_{i} = \mathbf{R}_{B}^{I} \left(\begin{bmatrix} \mathbf{I} & (\hat{\mathbf{r}}_{i}^{B})^{T} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{X}}_{o}^{B} \\ \ddot{\theta} \end{bmatrix} + \hat{\omega}(\hat{\omega}\mathbf{r}_{i}^{B}) \right) , \quad (5)$$

where I is the identity matrix.

We now represent the load's translational and rotational dynamics together in the following matrix form:

$$\begin{bmatrix} m_o \mathbf{I} & 0\\ 0 & I_o \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{X}}_o^B\\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \cdots & \mathbf{I}\\ \hat{\mathbf{r}}_1^B & \cdots & \hat{\mathbf{r}}_N^B \end{bmatrix} \boldsymbol{\mathcal{F}}^B, \quad (6)$$
$$\boldsymbol{\mathcal{F}}^B = \begin{bmatrix} (\mathbf{F}_1^B)^T & \cdots & (\mathbf{F}_N^B)^T \end{bmatrix}^T,$$

where \mathbf{F}_i^B is force \mathbf{F}_i in coordinate frame B, m_o is the mass of the load, and I_o is the load's moment of inertia about the axis normal to the plane of the motion and passing through its CG. Solving for the load's acceleration vector from this equation and substituting it into Equation (5), we obtain the acceleration of robot i as:

$$\ddot{\mathbf{X}}_{i} = \mathbf{R}_{B}^{I} \begin{bmatrix} \mathbf{I} & (\hat{\mathbf{r}}_{i}^{B})^{T} \end{bmatrix} \mathbf{M}_{o}^{-1} \begin{bmatrix} \mathbf{I} & \cdots & \mathbf{I} \\ \hat{\mathbf{r}}_{1}^{B} & \cdots & \hat{\mathbf{r}}_{N}^{B} \end{bmatrix} \boldsymbol{\mathcal{F}}^{B} \\ + \mathbf{R}_{B}^{I} \hat{\omega} (\hat{\omega} \mathbf{r}_{i}^{B}) , \qquad (7)$$

where

$$\mathbf{M}_{o} = \begin{bmatrix} m_{o}\mathbf{I} & 0\\ 0 & I_{o} \end{bmatrix}.$$
(8)

We can rewrite Equation (7) as the sum of two terms, one containing the force applied by robot i and the other containing the forces applied by all the other robots:

$$\ddot{\mathbf{X}}_i = \mathbf{Q} + \mathbf{P}\mathbf{F}_i , \qquad (9)$$

where

$$\mathbf{P} = \mathbf{R}_{B}^{I} \begin{bmatrix} \mathbf{I} & (\hat{\mathbf{r}}_{i}^{B})^{T} \end{bmatrix} \mathbf{M}_{o}^{-1} \begin{bmatrix} \mathbf{I} \\ \hat{\mathbf{r}}_{i}^{B} \end{bmatrix} \mathbf{R}_{I}^{B}$$
(10)

and

$$\mathbf{Q} = \mathbf{R}_B^I \mathbf{Q}_1 \mathbf{Q}_2 \boldsymbol{\mathcal{F}}_r^B + \mathbf{R}_B^I \hat{\omega} (\hat{\omega} \mathbf{r}_i^B) , \qquad (11)$$

$$\mathbf{Q}_{1} = \begin{bmatrix} \mathbf{I} & (\hat{\mathbf{r}}_{i}^{B})^{T} \end{bmatrix} \mathbf{M}_{o}^{-1} ,$$

$$\mathbf{Q}_{2} = \begin{bmatrix} \mathbf{I} & \cdots & \cdots & \cdots & \mathbf{I} \\ \hat{\mathbf{r}}_{1}^{B} & \cdots & \hat{\mathbf{r}}_{i-1}^{B} & \hat{\mathbf{r}}_{i+1}^{B} & \cdots & \hat{\mathbf{r}}_{N}^{B} \end{bmatrix} ,$$

$$\boldsymbol{\mathcal{F}}_{r}^{B} = \begin{bmatrix} (\mathbf{F}_{1}^{B})^{T} & \cdots & (\mathbf{F}_{i-1}^{B})^{T} & (\mathbf{F}_{i+1}^{B})^{T} & \cdots & (\mathbf{F}_{N}^{B})^{T} \end{bmatrix}^{T}$$
(12)

From Equation (9), we can solve for \mathbf{F}_i as

$$\mathbf{F}_i = \mathbf{P}^{-1} (\mathbf{X}_i - \mathbf{Q}) \ . \tag{13}$$

Finally, by substituting this expression into Equation (1), we obtain an equation of the form Equation (2), where

$$\mathbf{h} = \mathbf{M}_a^{-1} \mathbf{P}^{-1} \mathbf{Q} ,$$

$$\mathbf{G} = \mathbf{M}_a^{-1} , \qquad (14)$$

(15)

in which

IV. CONTROLLER DESIGN

 $\mathbf{M}_a = m_i \mathbf{I} + \mathbf{P}^{-1}.$

Our objective is for a team of robots to transport the load at a constant speed v_{des} in a target direction, defined by the angle γ in Figure 1. To achieve this, we will design controllers for each robot that regulate the magnitude of its velocity to v_{des} and the direction of its velocity to γ . Since the controllers for each robot will be identical, we will drop the subscript *i* from the robot state variables and parameters in this section. We define the state vector for a robot as $\mathbf{x} = [\mathbf{X}^T \ \dot{\mathbf{X}}^T]^T$, where $\mathbf{X} = [x \ y]^T$ is the robot's position in inertial coordinate frame *I* and $\dot{\mathbf{X}} = [\dot{x} \ \dot{y}]^T$ is its velocity in frame *I*. The state vector of the robot is then $\mathbf{x} = [x \ y \ \dot{x} \ \dot{y}]^T$.

It will be useful to define a second global coordinate frame I_{γ} , with axes labeled x_I^* and y_I^* , by rotating coordinate frame I by the angle γ (see Figure 1). In this coordinate frame, the axis labeled x_I^* points in the target direction of transport. Denoting the rotation matrix from frame I to frame I_{γ} by $\mathbf{R}_I^{I_{\gamma}}$, the state vector in the coordinates of frame I_{γ} is given by $\mathbf{x}^* = \mathbf{R}_I^{I_{\gamma}}\mathbf{x}$. The components of this state vector are $\mathbf{x}^* = [(\mathbf{X}^*)^T \ (\dot{\mathbf{X}}^*)^T]^T = [x^* \ y^* \ \dot{x}^* \ \dot{y}^*]^T$. The transformed control input is $\mathbf{u}^* = \mathbf{R}_I^{I_{\gamma}}\mathbf{u}$. We can then write the robot dynamics Equation (2) in the frame I_{γ} as

$$\ddot{\mathbf{X}}^* = \mathbf{h}^* + \mathbf{G}^* \mathbf{u}^* , \qquad (16)$$

in which $\mathbf{h}^* = \mathbf{R}_I^{I_{\gamma}} \mathbf{h}$ and $\mathbf{G}^* = \mathbf{R}_I^{I_{\gamma}} \mathbf{G} \mathbf{R}_{I_{\gamma}}^I$.

We denote the components of the vector $\mathbf{u}^* \in \mathbb{R}^2$ by $\mathbf{u}^* = [u_1^* \ u_2^*]^T$, where u_1^* is the robot's actuating force in the desired direction of transport and u_2^* is its actuating force

normal to this direction. We will design each of these control inputs as a sliding mode controller that drives all possible robot trajectories $\mathbf{x}^*(t)$ to enter a *sliding manifold* in the robot's state space in finite time and remain on the manifold thereafter [16]. The robot exhibits a desired dynamical behavior when its state evolves along the manifold. To regulate the robot's speed to v_{des} in the desired direction of transport (along the x_I^* axis), we define a sliding manifold s_1 as

$$s_1 = \dot{x}^* - v_{des} = 0 . (17)$$

To stabilize the direction of the robot's velocity to the angle γ , we define a sliding manifold s_2 that sets the component of the robot's velocity along the y_I^* axis to zero:

$$s_2 = \dot{y}^* = 0$$
 . (18)

Using the approach for sliding mode control design in [16], we define the control laws for u_1^* and u_2^* as

$$u_1^* = -k_1 \, sgn(s_1) \,, \tag{19}$$

$$u_2^* = -k_2 \, sgn(s_2) \,, \tag{20}$$

where k_1 and k_2 are control gains. These gains must be large enough to stabilize the system on the sliding manifolds. We derive lower bounds on the gains in section V. To eliminate chattering on the sliding manifolds without considerably affecting the controller performance, the signum functions in these controllers can be replaced by saturation functions, as proposed in [16] and [17].

We note that since \mathbf{G}^* is not a diagonal matrix, u_2^* affects the motion along x_I^* in addition to y_I^* , and u_1^* influences the motion along y_I^* in addition to x_I^* . We can describe these effects as a bounded nonlinear term that is added to the vector \mathbf{h}^* in Equation (16). Since sliding mode controllers are robust to variations in the \mathbf{h}^* term, these effects will not deteriorate the controller performance.

V. STABILITY ANALYSIS

We first derive some preliminary results that we will need to prove the stability of the system driven by the sliding mode controllers.

Proposition 5.1: The matrix \mathbf{P} *in Equation* (10) *is positive definite.*

Proof: We define a matrix \mathbf{P}_1 as:

$$\mathbf{P}_{1} = \begin{bmatrix} \mathbf{I} & (\hat{\mathbf{r}}_{i}^{B})^{T} \end{bmatrix} \mathbf{M}_{o}^{-1} \begin{bmatrix} \mathbf{I} \\ \hat{\mathbf{r}}_{i}^{B} \end{bmatrix}.$$
 (21)

Then, by Equation (10), $\mathbf{P} = \mathbf{R}_B^I \mathbf{P}_1 \mathbf{R}_I^B$. Since rotation matrices are invertible, the matrices \mathbf{P} and \mathbf{P}_1 are similar, and thus they have the same eigenvalues. Using the definition $\mathbf{r}_i^B = \begin{bmatrix} r_{i,x} & r_{i,y} \end{bmatrix}^T$ and the definition of \mathbf{M}_o from Equation (8), \mathbf{P}_1 can be calculated from Equation (21) as

$$\mathbf{P}_{1} = \begin{bmatrix} \frac{1}{m_{o}} + \frac{r_{i,y}^{2}}{I_{o}} & \frac{-r_{i,x}r_{i,y}}{I_{o}} \\ \frac{-r_{i,x}r_{i,y}}{I_{o}} & \frac{1}{m_{o}} + \frac{r_{i,x}^{2}}{I_{o}} \end{bmatrix}.$$
 (22)

The eigenvalues of \mathbf{P}_1 are

$$\lambda_1 = \frac{1}{m_o}, \quad \lambda_2 = \frac{1}{m_o} + \frac{||\mathbf{r}_i^B||^2}{I_o},$$
 (23)

which are both positive. Since these are also the eigenvalues of \mathbf{P} , the matrix \mathbf{P} is positive definite.

Proposition 5.2: The matrix **G** in Equation (14), and consequently \mathbf{G}^* in Equation (16), is positive definite and has constant eigenvalues.

Proof: Since $\mathbf{G} = \mathbf{M}_a^{-1}$ by Equation (14), we need to show that \mathbf{M}_a , and consequently \mathbf{M}_a^{-1} , is positive definite with constant eigenvalues. Let \mathbf{e}_1 and \mathbf{e}_2 denote the eigenvectors of \mathbf{M}_a , with corresponding eigenvalues μ_1 and μ_2 . Using Equation (15) for \mathbf{M}_a , we obtain

$$\mathbf{M}_{a}\mathbf{e}_{j} = (m_{i}\mathbf{I} + \mathbf{P}^{-1})\mathbf{e}_{j} = \mu_{j}\mathbf{e}_{j}, \quad j = 1, 2.$$
(24)

This equation can be rearranged as

$$\mathbf{P}^{-1}\mathbf{e}_{j} = \mu_{j}\mathbf{e}_{j} - m_{i}\mathbf{e}_{j} = (\mu_{j} - m_{i})\mathbf{e}_{j}, \quad j = 1, 2.$$
 (25)

Hence, the eigenvalues of \mathbf{P}^{-1} are $\mu_1 - m_i$ and $\mu_2 - m_i$. Since the eigenvalues of \mathbf{P} were found to be λ_1 and λ_2 as defined in Equation (23), and the eigenvalues of \mathbf{P}^{-1} are the inverses of the eigenvalues of \mathbf{P} , we have that $\mu_1 - m_i = \lambda_1^{-1}$ and $\mu_2 - m_i = \lambda_2^{-1}$. Therefore, the eigenvalues of \mathbf{M}_a are $\mu_1 = m_i + \lambda_1^{-1}$ and $\mu_2 = m_i + \lambda_2^{-1}$, which are both positive and constant, making \mathbf{M}_a a positive definite matrix with constant eigenvalues.

Lemma 5.3: If all the robots in a transport team apply control forces Equation (19) and Equation (20) to the load, then the angular velocity of the load will remain bounded.

Proof: Since the robots are rigidly attached to the load, the rotational dynamics of the entire system are given by

$$I_s \ddot{\theta} = \sum_{i=1}^N \hat{\mathbf{r}}_i^B \mathbf{R}_{I_\gamma}^I \mathbf{u}_i^*, \quad I_s = I_o + \sum_{i=1}^N m_i \left\| \mathbf{r}_i^B \right\|^2.$$
(26)

We define the angular difference between the load's orientation and the target direction as $\phi = \theta - \gamma$. Since γ is constant, $\dot{\phi} = \dot{\theta}$ and $\ddot{\phi} = \ddot{\theta}$. Writing Equation (26) in terms of ϕ and substituting in the control laws Equation (19) and Equation (20), we obtain:

$$I_{s}\ddot{\phi} = \left(k_{1}\sum_{i=1}^{N}r_{i,y}sgn(s_{i,1}) - k_{2}\sum_{i=1}^{N}r_{i,x}sgn(s_{i,2})\right)\cos(\phi) + \left(k_{1}\sum_{i=1}^{N}r_{i,x}sgn(s_{i,1}) + k_{2}\sum_{i=1}^{N}r_{i,y}sgn(s_{i,2})\right)\sin(\phi),$$
(27)

where $s_{i,1}$ and $s_{i,2}$ are the sliding modes Equation (17) and Equation (18) that are defined in terms of the velocity $\dot{\mathbf{X}}_i^* = [\dot{x}_i^* \ \dot{y}_i^*]^T$ of robot *i* in coordinate frame I_{γ} . Since $\dot{\mathbf{X}}_i^*$ is a function of $\dot{\mathbf{X}}_o = [\dot{x}_o \ \dot{y}_o]^T$, the velocity of the load's CG in coordinate frame *I*, and the load's orientation $\theta = \phi + \gamma$ and angular velocity $\dot{\theta} = \dot{\phi}$, we can write Equation (27) in the following form:

$$\ddot{\phi} = \eta(\dot{x}_o, \dot{y}_o, \phi, \dot{\phi}) \cos\left(\phi\right) + \zeta(\dot{x}_o, \dot{y}_o, \phi, \dot{\phi}) \sin\left(\phi\right), \quad (28)$$

where the coefficients η and ζ are bounded since both are finite summations of signum functions:

$$|\eta| \le \delta_{\eta} , \qquad |\zeta| \le \delta_{\zeta} . \tag{29}$$

To prove the boundedness of the load's angular velocity $\dot{\phi}$ from Equation (28), we can use the comparison lemma presented in [16]. Here, we apply this lemma to the simpler equation $\ddot{\phi} = \eta \cos{(\phi)}$, since a similar approach can be used for the entire Equation (28). We define a function

$$v(t) = \frac{1}{2}\dot{\phi}(t)^2$$
 (30)

whose time derivative can be calculated as

$$\dot{v} = \dot{\phi}\ddot{\phi} = \dot{\phi}\eta\cos(\phi) \le \delta_\eta\dot{\phi}\cos(\phi),$$
 (31)

with the bound on η defined in Equation (29). Using Equation (30), we can write the upper bound in Equation (31) in terms of v(t) as:

$$\dot{v} \leq \delta_{\eta} \sqrt{2v} \cos\left(\int_{0}^{t} \sqrt{2v} d\tau\right).$$
 (32)

We now define another function, w(t), that is the solution to the following equation:

$$\dot{w} = \delta_{\eta} \sqrt{2w} \cos\left(\int_0^t \sqrt{2w} d\tau\right), \quad w(0) = w_0.$$
 (33)

By the comparison lemma from [16], we can conclude that $v(t) \leq w(t)$ for all $t \geq 0$. From the definition of v(t) in Equation (30), this implies that $\dot{\phi}(t)^2/2 \leq w(t)$, and thus we obtain an upper bound on the load's angular velocity,

$$|\dot{\phi}(t)| \leq \sqrt{2|w(t)|}, \quad t \ge 0.$$
 (34)

We can derive an expression for the upper bound in Equation (34) by solving Equation (33) for w(t) and then obtaining an upper bound for |w(t)|. To solve Equation (33), we can use the following change of variables,

$$\xi = \sqrt{2w} \; \Rightarrow \; \dot{\xi} = \frac{\dot{w}}{\sqrt{2w}},$$

and rewrite Equation (33) as:

$$\dot{\xi} = \delta_{\eta} \cos\left(\int_{0}^{t} \xi d\tau\right), \quad \xi(0) = \sqrt{2w_0}.$$
 (35)

Using another change of variables,

$$\psi = \int_0^t \xi d\tau \; \Rightarrow \; \dot{\psi} = \xi, \; \ddot{\psi} = \dot{\xi},$$

Equation (35) can be written as

$$\ddot{\psi} = \delta_{\eta} \cos(\psi), \quad \psi(0) = \psi_0, \ \dot{\psi}(0) = \sqrt{2w_0}.$$
 (36)

This is the equation of motion of a simple pendulum, which can be integrated once to obtain

$$\frac{1}{2}\dot{\psi}^2 - \delta_\eta \sin(\psi) = \frac{1}{2}\dot{\psi}^2(0) - \delta_\eta \sin(\psi_0) \equiv c. \quad (37)$$

Using the fact that $\dot{\psi} = \xi = \sqrt{2w}$, we have the relation $w = \dot{\psi}^2/2$, and so by Equation (37),

$$w = \delta_{\eta} \sin(\psi) - \delta_{\eta} \sin(\psi_0) + w_0. \tag{38}$$

Then, using the triangle inequality and the fact that $w_0 > 0$,

$$|w| \leq 2\delta_{\eta} + w_0.$$

If we use the same procedure for the term containing ζ in Equation (28), we can modify the bound as:

$$|w| \leq 2\delta_{\eta} + 2\delta_{\zeta} + w_0. \tag{39}$$

Substituting Equation (39) into Equation (34) yields the following finite upper bound on the load's angular velocity:

$$|\dot{\phi}(t)| \le \sqrt{2|w(t)|} \le \sqrt{4\delta_{\eta} + 4\delta_{\zeta} + 2w_0}, \quad t \ge 0.$$
 (40)

Note that since $v(0) \le w(0) = w_0$ by the comparison lemma and $v(0) = \dot{\phi}(0)^2/2$ by Equation (30), setting $w_0 = 0$ implies that $\dot{\phi}(0) = 0$, meaning that the load starts with zero angular velocity at t = 0.

The nonlinear term h defined in Equation (14) is a function of the load's angular velocity ω , mass, and geometric properties, as well as the forces applied by the robots. Lemma 5.3 states that ω is bounded, and the other parameters are bounded as well due to the fact that the load has finite mass and dimensions and the robots' forces cannot exceed a saturation limit. This implies that h is bounded, a result that we will use subsequently in our stability analysis.

To analyze the stability of the system, we follow the approach in [16] and define the Lyapunov functions $V_1 = \frac{1}{2}s_1^2$ and $V_2 = \frac{1}{2}s_2^2$, which measure the distance of a robot state trajectory \mathbf{x}^* from the sliding manifolds Equation (17) and Equation (18), respectively. The time derivatives of these functions are:

$$\dot{V}_1 = s_1 \dot{s}_1 = s_1 \ddot{x}^*, \tag{41}$$

$$\dot{V}_2 = s_2 \dot{s}_2 = s_2 \ddot{y}^*.$$
 (42)

In order for the system to be asymptotically stable, these functions should both be negative whenever $|s_1|, |s_2| \neq 0$.

We will conduct the analysis just for \dot{V}_1 , since the analysis for \dot{V}_2 is similar. The expression for \ddot{x}^* can be obtained from the first component of Equation (16) and substituted into Equation (41), yielding

$$\dot{V}_1 = s_1(h_1^* + g_{11}^* u_1^* + g_{12}^* u_2^*),$$
(43)

where h_1^* is the first component of \mathbf{h}^* and g_{ij}^* is the entry of matrix \mathbf{G}^* at row *i* and column *j*. Our finding that \mathbf{h} is bounded implies that $h_1^* < \rho_1$ for some positive constant ρ_1 . In addition, suppose that ϵ_{11} , ϵ_{12} , and g_0 are positive constants such that $g_{11}^* \ge g_0 > 0$ and the following condition is satisfied:

$$\left|\frac{h_1^* + g_{12}^* u_2^*}{g_{11}^*}\right| \le \rho_1 + \epsilon_{11} \left|u_1^*\right| + \epsilon_{12} \left|u_2^*\right|.$$
(44)

Here, the constants ϵ_{ij} incorporate the uncertainties associated with $(g_{12}^*u_2^*)/g_{11}^*$.

By Proposition 5.2, the matrix \mathbf{G}^* is positive definite, which implies that its diagonal elements g_{ii}^* are positive. Therefore, we can multiply both sides of Equation (44) by $g_{11}^* |s_1|$ to obtain:

$$|s_{1}|(h_{1}^{*}+g_{12}^{*}u_{2}^{*}) \leq \rho_{1}g_{11}^{*}|s_{1}|+g_{11}^{*}\epsilon_{11}|u_{1}^{*}||s_{1}| + g_{11}^{*}\epsilon_{12}|u_{2}^{*}||s_{1}|.$$

$$(45)$$

Noting that $s_1 \leq |s_1|$ and that $|u_1^*| = k_1$, $|u_2^*| = k_2$ by Equation (19) and Equation (20), we have that:

$$s_{1}(h_{1}^{*} + g_{12}^{*}u_{2}^{*}) \leq \rho_{1}g_{11}^{*}|s_{1}| + g_{11}^{*}\epsilon_{11}k_{1}|s_{1}| + g_{11}^{*}\epsilon_{12}k_{2}|s_{1}|.$$
(46)

If we add $s_1g_{11}^*u_1^*$ to both sides of this inequality, the term on the left side becomes \dot{V}_1 by Equation (43). Then, noting that $s_1g_{11}^*u_1^* = -g_{11}^*k_1 |s_1|$, this inequality can be written as

$$\dot{V}_1 \leq g_{11}^* |s_1| (\rho_1 + \epsilon_{11}k_1 - k_1 + \epsilon_{12}k_2)$$
 (47)

We can follow the same procedure to compute an upper bound on \dot{V}_2 .

From Equation (47), we can derive the following condition on ρ_1 to ensure that $\dot{V}_1 < 0$ whenever $|s_1|, |s_2| \neq 0$:

$$\rho_1 \le (1 - \epsilon_{11})k_1 - \epsilon_{12}k_2. \tag{48}$$

Similarly, the following condition on a positive constant ρ_2 can be derived to ensure that $\dot{V}_2 < 0$ whenever $|s_1|, |s_2| \neq 0$:

$$p_2 \le (1 - \epsilon_{22})k_2 - \epsilon_{21}k_1,$$
(49)

where ϵ_{21} , ϵ_{22} are positive constants. Now, by defining $\mathbf{b} = [b_1 \ b_2]^T$ in which $b_1, b_2 > 0$, the inequalities Equation (48) and Equation (49) can be written in the form of a matrix equation as:

$$\begin{bmatrix} \rho_1 \\ \rho_2 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \left(\mathbf{I} - \begin{bmatrix} \epsilon_{11} & \epsilon_{12} \\ \epsilon_{21} & \epsilon_{22} \end{bmatrix} \right) \begin{bmatrix} k_1 \\ k_2 \end{bmatrix}.$$
(50)

Define the matrix with entries ϵ_{ij} as **E**. Since k_i , ρ_i , and b_i are positive, $(\mathbf{I} - \mathbf{E})$ must be nonsingular, and all elements of $(\mathbf{I} - \mathbf{E})^{-1}$ must be positive. This implies that $(\mathbf{I} - \mathbf{E})$ is an M-matrix [16]. Then, Equation (50) can be solved for the control gains k_1 and k_2 :

$$\begin{bmatrix} k_1\\ k_2 \end{bmatrix} = (\mathbf{I} - \mathbf{E})^{-1} \begin{bmatrix} \rho_1 + b_1\\ \rho_2 + b_2 \end{bmatrix}.$$
 (51)

By choosing these control gains and substituting them into the upper bound Equation (47) on \dot{V}_1 and the corresponding upper bound on \dot{V}_2 , we obtain the following inequalities:

$$\dot{V}_1 \le -g_0 b_1 |s_1|, \quad \dot{V}_2 \le -g_0 b_2 |s_2|.$$
 (52)

Since $g_0, b_1, b_2 > 0$, it is evident that $\dot{V}_1 < 0$ and $\dot{V}_2 < 0$ when $|s_1|, |s_2| \neq 0$. Hence, the system is asymptotically stable for these gains k_1 and k_2 , meaning that all state trajectories will reach the intersection of the two sliding manifolds in finite time and remain on it thereafter.

VI. SIMULATION RESULTS

We validated our sliding mode control strategies with simulations of point-mass robots in MATLAB and with high-fidelity 3D physics simulations in the robot simulator Webots [18]. The robots in the Webots simulations are 3D models of the small mobile robot platform "Pheeno" that has been developed in our lab [1]. To address the problem of chattering on the sliding manifolds, we used the approach mentioned in section IV: we replaced each signum function



Fig. 3. Collective transport of a load by five point-mass robots.



Fig. 4. Sliding mode parameters $(s_1 \text{ and } s_2)$ of the five robots.

 $sgn(s_i)$ in the controllers Equation (19), Equation (20) with a saturation function $sat(s_i/\epsilon_{b_i})$, where ϵ_{b_i} is a *boundary layer* parameter that gives the bounds on an envelope around $s_i = 0$ within which trajectories can evolve to avoid chattering.

A. Simulation with point-mass robots

We simulated a scenario in which five point-mass robots, marked by the red dots in Figure 3, must transport an asymmetric load to a goal, the heading to which is $\gamma = 30^{\circ}$. The desired load velocity was set to $v_{des} = 0.1 m/s$, and the controller parameters were set to $k_1 = k_2 = 0.4$, $\epsilon_{b_1} = \epsilon_{b_2} = 0.01$. The mass of the load is 1 kg, and its moment of inertia is $0.33 kg/m^2$. Each robot has a mass of 0.1 kg and can apply a maximum force of 0.1 N on the load. The system was simulated for 120 s. As Figure 3 shows, the load and the robots exhibit fairly straight trajectories that are parallel to the desired path to the goal, illustrated by the dashed line.

In addition, Figure 4 plots the values of the sliding mode parameters, s_1 and s_2 , for all the robots during the first 3 s of the transport. These values all quickly converge to the boundary layer ($|s_1| < 0.01$, $|s_2| < 0.01$) within 1 s. Figure 5, which plots the load's angular position and its drift from the desired path, shows that system converges to a stable equilibrium state after a negligible initial load rotation that produces a slight initial drift of about 0.03 mm.



Fig. 5. Load rotation (left) and its drift from the desired path (right).

B. Simulations with a model of Pheeno in Webots

We also developed 3D simulations that incorporate realistic physical effects arising from the robots' wheeled actuation system and the additional degrees of freedom introduced by the manipulator arms. In addition, these simulations required modifications to the sliding mode controllers to account for the fact that Pheeno is a nonholonomic, differential-drive platform. As defined, the controllers require the velocity of the attachment point of a robot to the load, and so they require the velocity of Pheeno's end-effector, which necessitates computing the Jacobian matrix and consequently including the geometry of the manipulator arm in the control commands. However, there is an alternate way to control the heading and velocity of Pheeno during transport, which we pursue here. Let $\dot{\theta}_R$ and $\dot{\theta}_L$ denote the angular velocities of the right and left wheel of Pheeno, respectively, and τ_R and τ_L be the corresponding actuation torques on the wheels. These torques are the control inputs to the robot. We define:

$$\dot{\theta}_{H} = \frac{1}{2} (\dot{\theta}_{R} - \dot{\theta}_{L}), \quad \tau_{H} = \frac{1}{2} (\tau_{R} - \tau_{L}), \dot{\theta}_{V} = \frac{1}{2} (\dot{\theta}_{R} + \dot{\theta}_{L}), \quad \tau_{V} = \frac{1}{2} (\tau_{R} + \tau_{L}).$$
(53)

We diagonalize the linear model developed in [19] for a differential-drive robot and write it in the following form:

$$\begin{bmatrix} A - B & 0 \\ 0 & A + B \end{bmatrix} \begin{bmatrix} \ddot{\theta}_H \\ \ddot{\theta}_V \end{bmatrix} + \begin{bmatrix} K & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} \dot{\theta}_H \\ \dot{\theta}_V \end{bmatrix} = \begin{bmatrix} \tau_H \\ \tau_V \end{bmatrix},$$
(54)

where the constants A and B depend on the geometry and mass properties of the robot, and the constant K is the damping in the wheels.

Equation (54) provides us with two decoupled equations. One equation governs the robot's heading angle φ , which is proportional to $(\theta_R - \theta_L)$, and the other governs the robot's speed v, which is proportional to $(\dot{\theta}_R + \dot{\theta}_L)$. Defining $s_H = \varphi - \gamma$ and $s_V = v - v_{des}$, we can formulate the following sliding mode controllers for the robot's heading and speed:

$$\tau_H = -k_H \, sat\left(\frac{s_H}{\epsilon_{b_H}}\right), \quad \tau_V = -k_V \, sat\left(\frac{s_V}{\epsilon_{b_V}}\right). \tag{55}$$

We implemented these controllers in a Webots simulation in which five Pheeno robots grasp a load, lift it simultaneously, and transport it to a goal location at a heading of $\gamma = 30^{\circ}$. The desired load velocity, load mass and moment of inertia, robot mass, and robot maximum force were all set to the same values as in the point-mass simulations. The system



Fig. 6. Collective transport by five Pheenos simulated in Webots.



Fig. 7. The trajectories of the Pheenos and the load during transport.

was simulated for 120 s. The controller parameters were set to $k_H = 0.03$, $k_V = 0.09$, $\epsilon_{b_H} = 0.01$, and $\epsilon_{b_V} = 0.1$. Snapshots of the simulation are shown in Figure 6. Figure 7 plots the load and the robot trajectories, which are straight and parallel in the desired direction. Figure 8 shows the time evolution of the sliding mode parameters, which all converge within the specified boundary layers.

VII. EXPERIMENTAL RESULTS

To further validate the control strategies, we conducted five experimental trials of collective transport with four Pheeno robots and a rectangular load. The robots and load were marked with 2D binary identification tags to enable real-time tracking of their positions and orientations by an overhead camera. The robots were initially placed in the configuration shown in Figure 9. This configuration was chosen to minimize unwanted effects such as wheel slip and unnecessary stress on the central servo, which controls the yaw angle of the manipulator arm about the central axis of the robot. Each robot updated its state estimate using a basic complementary filter acting on its onboard encoders,



Fig. 8. The sliding mode parameters $(s_V \text{ and } s_H)$ of the Pheenos during transport.



Fig. 9. A zoomed-in screenshot from the overhead camera that tracks the load and robots during the experiment.

compass, and accelerometer.

We implemented controllers similar to those in Equation (55) on the robots. However, instead of using torque inputs, the individual motor accelerations were controlled directly, *i.e.* τ_H and τ_V in Equation (55) were replaced by $\ddot{\theta}_H$ and $\ddot{\theta}_V$, respectively. Thus, the controllers required measurements of the wheel velocities and the robots' heading. The control parameters were set to $k_H = 0.01$, $k_V = 0.05$, and $\epsilon_{b_H} = \epsilon_{b_V} = 0.01$; the gains were lower than the gains in the Webots simulation to avoid causing the motors to accelerate too quickly, which results in wheel slip and odometry drift.

The robots were tasked with transporting the load at a desired velocity of $v_{des} = 10 \ cm/s$ along the x-axis of the global frame defined by the overhead camera. Each trial was run for 30 s. Figure 10 shows the paths of the load and transporting robots during a single experiment, and Figure 11 plots the average and standard deviation of the load's velocity, heading, and trajectory over the five experiments. These plots show that the sliding mode controllers are fairly successful at achieving the control objectives. The slight rotation of the load and its deviation from the desired path in Figure 10, as well as the increasing standard deviations in the plots in Figure 11, are due to unavoidable drift in the onboard odometry caused by wheel slip, sensor noise, and model error, among other factors. Sensor noise can result in discrepancies in the robots' velocities, causing the robots to exert torques on each other through the load, which produces wheel slip and error in the odometry.

VIII. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a decentralized control strategy for multi-robot collective transport based on sliding mode control. The controllers do not require inter-robot communication, knowledge of the load dynamics and geometry, or



Fig. 10. The trajectories of the Pheenos and the transported load during one experiment. The rectangle shows the orientation of the load at several time points. The colored circles mark the robot attachment points at the beginning and end of the experiment.



Fig. 11. The states of the load during the transport experiments. The dark blue lines are the mean states averaged over five trials, and the light blue area shows the standard deviation. *Top Left:* The average velocity of the load during transport. *Top Right:* The average heading of the load during transport. *Bottom:* The average trajectory of the load during transport.

the size and spatial configuration of the transport team. We validated this control strategy in simulations with point-mass robots and with 3D models of robots with realistic dynamics, as well as in experiments with a team of small mobile robots. The simulations and experiments demonstrated the effectiveness of the strategy at driving the transport team to a target speed in a desired direction.

In future work, we will remove the assumption that all the robots know the desired direction of transport. We will consider the scenario where each robot knows its own direction to the goal from its odometric readings, and all robots must use a consensus protocol to reach an agreement on the average direction to the goal. The controllers proposed here could then be updated periodically with new outputs of the consensus algorithm. We will also consider leaderfollower control strategies in which the leader robots know the target direction to the goal, while follower robots must infer the leaders' intentions using tools from adaptive control.

IX. ACKNOWLEDGMENTS

H.F. thanks Konstantinos Tsakalis for his helpful suggestions regarding the use of the comparison lemma.

REFERENCES

- S. Wilson, R. Gameros, M. Sheely, M. Lin, K. Dover, R. Gevorkyan, M. Haberland, A. Bertozzi, and S. Berman, "Pheeno, a versatile swarm robotic research and education platform," *IEEE Robotics and Automation Letters (RA-L)*, vol. 1, no. 2, pp. 884–891, 2016.
- [2] Z. Wang and M. Schwager, "Multi-robot manipulation without communication," in *Int'l. Symposium on Distributed Autonomous Robotic Systems (DARS)*, Nov. 2014, pp. 43–56.
- [3] —, "Multi-robot manipulation with no communication using only local measurements," in *IEEE Conf. on Decision and Control (CDC)*, Dec. 2015, pp. 380–385.
- [4] —, "Kinematic multi-robot manipulation with no communication using force feedback," in *IEEE Int'l. Conf. on Robotics and Automation (ICRA)*, May 2016, pp. 427–432.
- [5] A. Yufka and M. Ozkan, "Formation-based control scheme for cooperative transportation by multiple mobile robots," *Int'l. J. Advanced Robotic Systems*, vol. 12, 2015.
- [6] J. Chen, M. Gauci, W. Li, A. Kolling, and R. Groß, "Occlusion-based cooperative transport with a swarm of miniature mobile robots," *IEEE Trans. on Robotics*, vol. 31, no. 2, pp. 307–321, 2015.
- [7] G. Habibi, K. Zachary, W. Xie, M. Jellins, and J. McLurkin, "Distributed centroid estimation and motion controllers for collective transport by multi-robot systems," in *IEEE Int'l. Conf. on Robotics* and Automation (ICRA), May 2015, pp. 1282–1288.
- [8] H. Bai and J. T. Wen, "Motion coordination through cooperative payload transport," in *American Control Conference (ACC)*, June 2009, pp. 1310–1315.
- [9] —, "Cooperative load transport: A formation-control perspective." *IEEE Trans. on Robotics*, vol. 26, no. 4, pp. 742–750, 2010.
- [10] S. Berman, Q. Lindsey, M. S. Sakar, V. Kumar, and S. Pratt, "Study of group food retrieval by ants as a model for multi-robot collective transport strategies," in *Robotics: Science and Systems (RSS)*, 2010.
- [11] M. Rubenstein, A. Cabrera, J. Werfel, G. Habibi, J. McLurkin, and R. Nagpal, "Collective transport of complex objects by simple robots: Theory and experiments," in *Int'l. Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2013, pp. 47–54.
- [12] N. Yagiz, Y. Hacioglu, and Y. Z. Arslan, "Load transportation by dual arm robot using sliding mode control," *Journal of Mechanical Science* and Technology, vol. 24, no. 5, pp. 1177–1184, 2010.
- [13] W. Yim, M. Selvarajan, and W. R. Wells, "Sliding mode cooperative motion control of dual arm manipulators," *Artificial Life and Robotics*, vol. 3, no. 3, pp. 166–169, 1999.
- [14] V. Parra-Vega, A. Sanchez, C. Izaguirre, O. Garcia, and F. Ruiz-Sanchez, "Toward aerial grasping and manipulation with multiple UAVs," J. Intell. & Robot. Syst., vol. 70, no. 1-4, pp. 575–593, 2013.
- [15] A.-N. Ponce-Hinestroza *et al.*, "Cooperative redundant omnidirectional mobile manipulators: Model-free decentralized integral sliding modes and passive velocity fields," in *IEEE Int'l. Conf. on Robotics and Automation (ICRA)*, May 2016, pp. 2375–2380.
- [16] H. K. Khalil, Nonlinear Systems. Upper Saddle River, N.J.: Prentice Hall, 1996.
- [17] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*. Englewood Cliffs, N.J.: Prentice Hall, 1991.
- [18] O. Michel, "Webots: Professional mobile robot simulation," *Int'l. J. Advanced Robotic Systems*, vol. 1, no. 1, pp. 39–42, Mar. 2004.
- [19] E. Ivanjko, T. Petrinic, and I. Petrovic, "Modelling of mobile robot dynamics," in *7th EUROSIM Congress on Modelling and Simulation*, vol. 2, 2010.