The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Decentralized Cooperative Manipulation with a Swarm of Mobile Robots

Joel M. Esposito

Dept. of Systems Engineering
U.S. Naval Academy, MD, USA

Abstract— **In this paper we consider cooperative manipulation problems where a large group (swarm) of non-articulated mobile robots is trying to cooperatively control the velocity of some larger rigid body by exerting forces around its perimeter. We consider a second-order dynamic model for the object but use a simplified contact model. We seek solutions that require minimal information sharing among the swarm members. We present a velocity control law that is asymptotically stable. In the case of a constant desired velocity, it is shown that no coordination is required between the swarm members. For more complex trajectories we introduce a decentralized feed-forward component that uses an online consensus estimate of the swarm's configuration. The results are illustrated in simulation.**

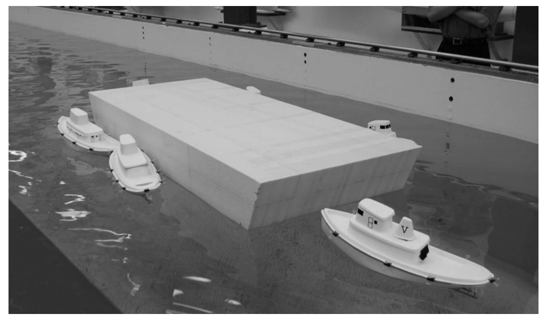

Fig. 1. Experimental test bed. A group of 6 unmanned tugboats (0.5 meters long) and a scale model flat bottomed barge (2 meters long). The tugs have articulated magnetic attachment devices used to grab the barge.

I. INTRODUCTION

Robot swarms[1] are large groups of small, relatively unsophisticated, robots working in concert to achieve objectives that are beyond the capability of a single robot. One example of an application that can benefit from this approach is *non-prehensile cooperative manipulation*, where a group of non-articulated mobile robots attempts to transport a larger object in the plane, by applying forces to its perimeter. The advantages of the swarm are: (1) its ability to distribute applied forces over a large area, achieving an enveloping grasp on large objects; and (2) the maximum wrench the swarm can exert increases linearly as the number of swarm members increases. We are particularly interested in marine applications involving autonomous tugboats such as towing disabled ships (ex. *U.S.S. Cole*), transporting components of large offshore structures (ex. oil platforms), or positioning littoral protection equipment (ex. hydrophone arrays). Figure 1 depicts our marine test-bed. However, most of our work can be extended to ground robots.

Behavior-based approaches to non-articulated cooperative manipulation yield interesting results but lack performance guarantees [15], [12], [5], [7]. On the other hand, so called "caging" approaches such as [17], [20], and [13], design controllers which force robots to surround the object. Inter-robot spacing is constrained to be small enough that it is impossible for the object to "escape" – meaning that one can prove as the robots move, so must the object. However, the problem is reduced to a positioning problem; limiting its applicability to manipulation problems that are not essentially kinematic, such as marine problems.

[1] `esposito@usna.edu`. Supported by ONR N0001405WRY20391.

More applicable to marine problems are methods that consider the full second order dynamics of the system such as [16], [18], [14], and [3]. However these approaches all require some centralized decision making or global knowledge of all swarm member's actions. We seek solutions that consider full second order dynamics, are fully distributed, and provide performance and stability guarantees such as the flocking control strategies presented in [10], [19], [4]. Eventually we hope to extend the work non-trivial contact models.

In this paper we assume a group of agents has already established contact with an object to be manipulated. Here, we do not explicitly consider the synthesis of the grasp (see our previous work [2]) or motion control of the swarm prior to establishing contact with the object. We describe the system and communications model in Sect. II. The primary contribution of this paper is a novel control law that lets each agent compute an applied force such that the swarm is able to drive the velocity of the object to some desired value, with minimal information sharing. In the case of a constant desired velocity and no drag we show that absolutely no communication between agents is required (Sect. III). In the case where the desired velocity varies with time, and drag forces are present, the controller (Sect. IV) utilizes a decentralized estimate of the feed forward terms based on the consensus protocol [9]. Simulation results are provided. In Sections V and VI we discuss the implications of the control laws.

II. MODEL

Consider a rigid body (see Fig. 2) with pose $p = [x, y, \psi]^T \in SE(2)$ defined relative to a global reference frame. A body fixed reference frame is attached to the center of mass. Linear velocity v and angular velocity ω are defined in the body frame. $R(\psi)$ is a rotation matrix converting body velocities to global velocities. N agents are attached to the body at points $r_1, \ldots, r_N \in \mathbb{R}^2$, where r_i is assumed to be time invariant in the body frame. Each of the agents can apply an input force $F_i \in \mathbb{R}^2$ written in body frame coordinates. We assume the agents are rigidly attached to the object and can therefore apply a force of any magnitude, in any direction (i.e. the grasp is not friction assisted). The system dynamics are:

$$\dot{p} = R(\psi)[v,\ \omega]^T \tag{1}$$

$$\dot{v} = -\omega \times v + F_{drag} + M^{-1}\sum_{i=1}^{N} F_i \tag{2}$$

$$\dot{\omega} = \tau_{drag} + J^{-1}\sum_{i=1}^{N} r_i \times F_i. \tag{3}$$

Where the positive scalars M and J are the effective mass and planar moment of inertia of the object along with the attached agents. We assume the products of inertia are zero (i.e. left/right and fore/aft symmetry). F_{drag} and τ_{drag} are drag forces and torques the object experiences. The appropriate model for these is application dependent (eg. ground-based vs. marine, low vs. high-speed, etc.)

Note that under these assumptions, in general there must be at least 2 agents, $N > 1$, which are not co-located, $r_1 \neq r_2$, to ensure small time local controllability. Under the contact model described here, this is equivalent to requiring a force closure grasp [8].

For some tasks the agents will need to share information via some type of *ad hoc* wireless radio or optical communication link. The communication network is modeled as a graph $G = (V, E)$. Each vertex in the graph, $i \in V$, represents an agent and each edge, $e_{ij} \in E$, represents a wireless communication link between agent i and j. We assume the set of links in the network (its topology) is static, message transfer is synchronous, that each edge permits an unlimited data transfer rate, and that there are no time delays or noise in transmission. A network G is said to be *connected* if the communication graph G is connected (*i.e.* if for any node pair i, j there exists an edge path of arbitrary length between them). Occasionally we discuss the network *neighbors* of agent i, the set of all nodes a single hop away, defined as

$$\mathcal{N}_i = \{j \in V | \exists e_{ij}\}. \tag{4}$$

III. CONSTANT VELOCITY CONTROLLER

Assume the desired velocity of the object is a constant in the global frame, v_d^G, as well as the desired angular velocity ω_d^G; and that $F_{drag} = [0,0]^T$, and $\tau_{drag} = 0$. In this section

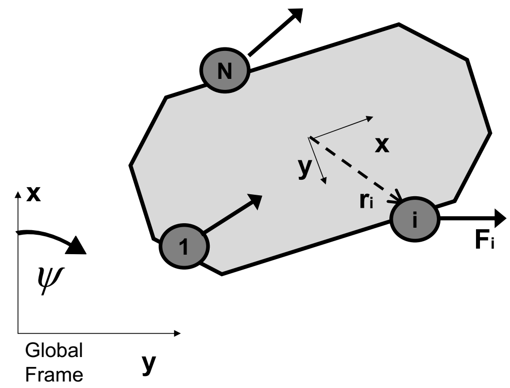

Fig. 2. Swarm Manipulation Scenario. N agents (dark circles) are attached to the object (shaded polygon). The fixed position of the i^{th} tug boat is defined by the vector r_i. It may apply a force F_i in any direction.

we show that if each agent works to regulate the velocity at point r_i

$$v_i = v + \omega \times r_i. \tag{5}$$

to the desired value of $v_d + \omega_d \times r_i$ the overall motion of the body converges to the desired velocity. No coordination between the agents is required.

Proposition 3.1: Assume v_d^G and ω_d are constants in the global frame. Note that since the system is planar $\omega_d^G = \omega_d$. Define error signals $e_v = R^T(\psi)v_d^G - v$ and $e_\omega = \omega_d - \omega$. If each agent applies the control law:

$$F_i \leftarrow \bar{F}_i = e_v + e_\omega \times r_i, \tag{6}$$

then $e_v, e_\omega \to 0$ as $t \to \infty$.

PROOF: To verify this, define the Lyapunov Function

$$W = \frac{1}{2}e_v^T M e_v + \frac{1}{2}e_\omega^T J e_\omega. \tag{7}$$

The derivative is

$$\begin{aligned} \dot{W} &= e_v^T M \dot{e_v} + e_\omega^T J \dot{e_\omega} \\ &= e_v^T M\{-\omega \times R^T v_d^G + \omega \times v - M^{-1}\sum_{i=1}^{N} F_i\} \\ &\quad -e_\omega^T J\{J^{-1}\sum_{i=1}^{N} r_i \times F_i\}. \end{aligned} \tag{8}$$

Substituting the control law $\bar{F}_i$ from (6) and noting that

$$e_v^T(e_\omega \times r_i) = e_\omega^T(r_i \times e_v),$$

and

$$e_\omega^T(r_i \times (e_\omega \times r_i)) = (e_\omega^T e_\omega)(r_i^T r_i),$$

the derivative of W simplifies to

$$\begin{aligned} \dot{W} &= -e_v^T M \omega \times e_v - \sum_{i=1}^{N} e_v^T e_v \\ &\quad -2e_\omega^T \sum_{i=1}^{N} r_i \times e_v - \sum_{i=1}^{N}(e_\omega^T e_\omega)(r_i^T r_i). \end{aligned} \tag{9}$$

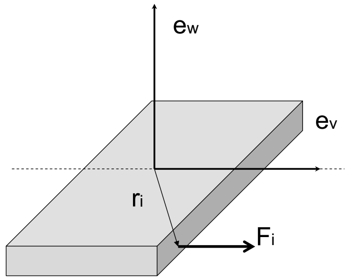

Fig. 3. Physical meaning of the triple scalar product. For agents such as i on the near side of the midline, defined by e_v, the triple scalar product is righthanded, meaning that they can apply forces that simultaneously decrease both error signals. While on the far side, e_ω and e_v represent conflicting goals.

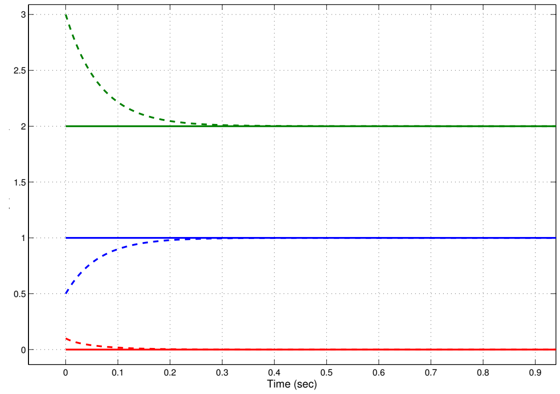

Fig. 4. Simulation of the constant velocity controller. Actual global frame velocities are dashed lines. Desired global frame velocities are solid: $v_x(m/s)$ (top, green), $v_y(m/s)$ (middle, blue), and ω (rad/s) (bottom, red).

Note that the first term is zero since M is a scalar and e_v and ω are always perpendicular. The second and fourth terms are clearly non-positive; and only zero where the error terms vanish. The third term is a triple scalar product whose magnitude is bounded by $\sum \|e_\omega\|\|r_i\|\|e_v\|$. However, its sign cannot be determined (see Fig. 3 for a graphical intuition). Fortunately the last three terms can be collectively bounded by

$$\dot{W} \leq -\sum_{i=1}^{N}(\|e_v\| - \|r_i\|\|e_\omega\|)^2. \quad (10)$$

The only scenario in which $\dot{W} = 0$ would be if all the following conditions held true for all $i = 1, \ldots, N$: (1) r_i is perpendicular to e_v; (2) $e_w^T r_i \times e_v < 0$; and (3) $\|e_v\| - \|r_i\|\|e_\omega\| = 0$. Only a single point on a rigid body satisfies all of these conditions. Since we assume $N > 1$ and $r_1 \neq r_2$, then $\dot{W} < 0$.

Remark 3.2: While the form of (6) is convenient for the proof, in practice it is much easier to measure the agent's velocity rather than the object's. (6) can be rewritten as

$$\bar{F}_i = v_d + \omega_d \times r_i - v_i. \quad (11)$$

Remark 3.3: A friction-based contact-model can easily be accommodated in this scenario, assuming the robots are in a force closure configuration. Each robot simply computes a desired force from (6), and checks to see if it is inside the set of admissible contact forces defined by the friction cone, $\mathcal{F}_i$. If $\bar{F}_i \in \mathcal{F}_i$, $F_i \leftarrow \bar{F}_i$; else $F_i \leftarrow [0, 0]^T$. The stability proof is not qualitatively altered in anyway; however the rate of convergence may be slowed since at any given time there are effectively less than N agents who are applying non-zero forces in (10).

Figure 4 shows a simulation of the control law, with non-zero initial velocity. The object is a rectangle, $2(m) \times 1(m)$, with $M = 1(kg)$. 15 agents are distributed randomly around the perimeter employing the control law (6). The desired velocities are $v_x = 2(m/s)$, $v_y = 1(m/s)$, and $\omega = 0(rad/s)$. The velocities quickly converge to the desired values.

IV. Time Varying Velocity Controller

It is more difficult to track time varying velocities, possibly in the presence of drag forces, in a distributed fashion since it generally requires using feed forward terms. Allocating the effects of those terms among the swarm members requires some coordination.

Assume v_d^G and ω_d are not constant in the global frame; and that $\dot{v}_d^G$ and $\dot{\omega}_d$ are also specified. Now the derivatives contain additional terms

$$\begin{aligned} \dot{e}_v &= -\omega \times R^T(\psi) v_d^G + R^T(\psi)\dot{v}_d^G \quad (12) \\ &\quad + \omega \times v - F_{drag} - M^{-1}\sum_{i=1}^{N} F_i \\ \dot{e}_\omega &= \dot{\omega}_d - \tau_{drag} - J^{-1}\sum_{i=1}^{N} r_i \times F_i. \quad (13) \end{aligned}$$

Define a new control law

$$\hat{F}_i = \bar{F}_i + \tilde{F}_i^d. \quad (14)$$

Using the Lyapunov Function defined previously, and our result from the constant velocity case, the derivative now contains two new terms:

$$\begin{aligned} \dot{W} &\leq -\sum_{i=1}^{N}(\|e_v\| - \|r_i\|\|e_\omega\|)^2 + \quad (15) \\ &\quad e_v^T \left\{ M R^T(\psi)\dot{v}_d^G - F_{drag} - \sum_{i=1}^{N}\tilde{F}_i^d \right\} + \\ &\quad e_\omega^T \left\{ J\dot{\omega}_d - \tau_{drag} - \sum_{i=1}^{N} r_i \times \tilde{F}_i^d \right\}. \end{aligned}$$

Suggesting each agent should compute $\tilde{F}_i^d$ such that

$$labeleq: feedforward \begin{bmatrix} \sum_{i=1}^{N}\tilde{F}_i^d \\ \sum_{i=1}^{N} r_i \times \tilde{F}_i^d \end{bmatrix} = \begin{bmatrix} M R^T(\psi)\dot{v}_d^G - F_{drag} \\ J\dot{\omega}_d - \tau_{drag} \end{bmatrix} \quad (16)$$

which can be re-written in matrix form.

Problem 4.1: Given the desired net feed forward wrench $F^d \in \mathbb{R}^3$ compute $\tilde{F}^d \in \mathbb{R}^{2N}$ such that

$$B\tilde{F}^d = F^d \tag{17}$$

where $B = [B_1 \ldots B_i \ldots B_N]$, and

$$B_i = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -r_i^y & r_i^x \end{bmatrix}.$$

Clearly, under the assumption that there are two or more agents that are not collocated, B is full rank and a solution exists. In fact,

$$\tilde{F}^d = B^\dagger F^d, \tag{18}$$

is a minimum effort solution, where $B^\dagger$ is the pseudo-inverse $B'(BB')^{-1}$. This solution was used in [1] and is analogous to a redundant manipulator controller.

The issue we consider here is: is it possible to implement this solution in an entirely distributed fashion; or, if not, what information much be shared among the agents to implement the solution. Let $\bar{B} = (BB') \in \mathbb{R}^{3\times3}$. Thenagent i's component of (18) is

$$\tilde{F}_i^d = \begin{bmatrix} 1 & 0 & -r_i^y \\ 0 & 1 & r_i^x \end{bmatrix} \bar{B}^{-1} F^d \tag{19}$$

where

$$\bar{B} = \begin{bmatrix} N & 0 & -m^y \\ 0 & N & m^x \\ -m^y & m^x & m^{zz} \end{bmatrix}. \tag{20}$$

Therefore each agent maintain an estimate of $\bar{B}$, called $\hat{B}_i$, based on the following quantities.

- F^d the feed-forward wrench computed from (**??**).
- r_i^x and r_i^y: its own location in the body frame (assumed known).
- N: the number of agents in contact with the object (assumed known).
- $m^y = \sum_{i=1}^N r_i^y$ and $m^x = \sum_{i=1}^N r_i^x$: The first moments of the swarm's configuration (estimated).
- $m^{zz} = \sum_{i=1}^N (r_i^y)^2 + (r_i^x)^2$: The second moment of the swarms's configuration about the rotational axis (estimated).

In order to estimate the values of the moments in a distributed fashion, we employ the consensus protocol discussed in [9]. Let $\hat{m}_i(t) = [\hat{m}_i^x, \hat{m}_i^y, \hat{m}_i^{zz}]^T$ be the i^{th} agent's estimate of the appropriate moment at time t. Estimates are updated according to the following dynamics

$$\dot{\hat{m}}_i = \sum_{j\in\mathcal{N}_i} (\hat{m}_j - \hat{m}_i) \tag{21}$$

using initial condition $\hat{m}_i(0) = [Nr_i^x, Nr_i^y, N(r_x^i)^2 + N(r_y^i)^2]$. In [11] it was shown that, provided the underlying graph is connected, such protocols are globally exponentially stable; and that the equilibrium value is a consensus equal to the mean of the agents's initial conditions. Therefore $\hat{m}_i(t) \to [m^x, m^y, m^{zz}]$ as $t \to \infty$.

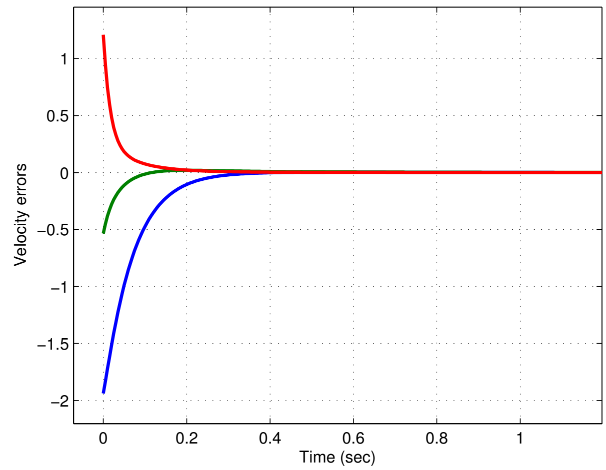

Fig. 5. Simulation of the time varying desired velocity controller. The plot depicts the: x component of e_v (m/s) (middle, green), y component of e_v (bottom, blue), and e_ω (rad/s) (top, red).

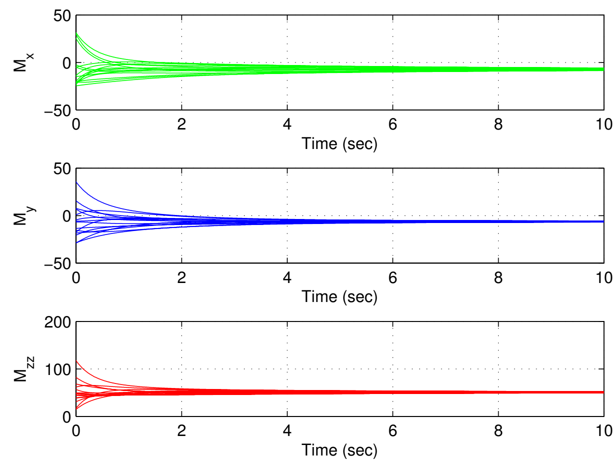

Fig. 6. Each agents estimate of the swarm's moments. m^x (top, green), m^y (middle, blue), and m^{zz} (bottom, red).

Since the dynamics of estimates are decoupled from the dynamics of the object, to prove the decentralized controller's stability it suffices to show that the feed forward term is always bounded. That is equivalent to ensuring the determinant of $\hat{B}$

$$N\hat{m}_i^{zz} - (\hat{m}_i^x)^2 - (\hat{m}_i^x)^2 \neq 0. \tag{22}$$

The only situations in which this happens are when all the agents are co-located (or there is only a single agent). Our assumption is that the agents are in a force closure configuration, which precludes this possibility. Therefore the equilibrium (consensus) value never causes the determinant to be zero. However the initial condition does violate this condition. One possible approach, used in the simulation below, is to not activate the pushing control laws until after the consensus protocol has gone through at least one iteration.

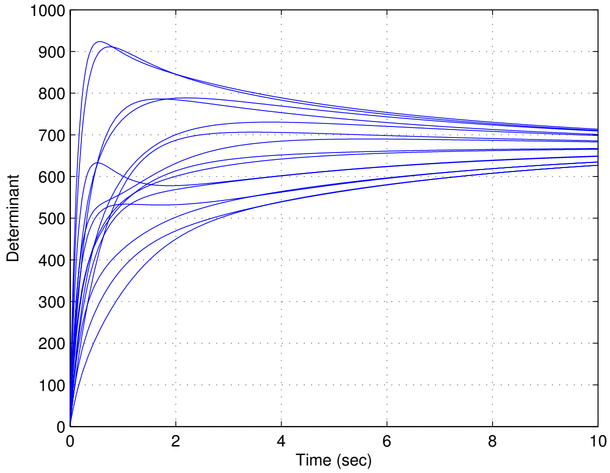

Fig. 7. The determinant of each agent's $\hat{B}$ matrix. Note they are non-zero after the first iteration.

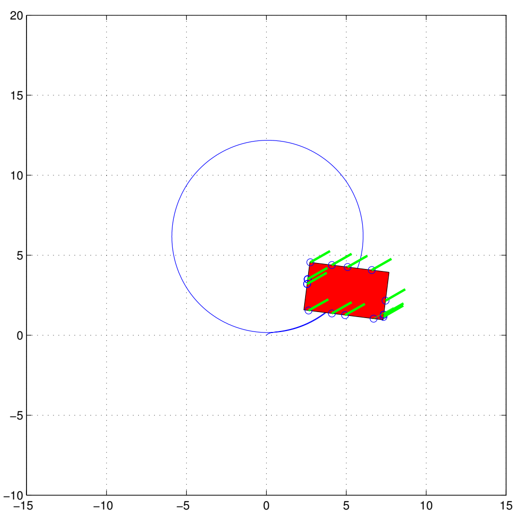

Fig. 8. A screen shot of the simulation. Green lines represent the agents forces. The path of the object's center of mass is a circle.

Figure 5 shows a simulation of the time varying velocity control law, with non-zero initial velocity. The velocity tracking errors quickly converge to zero. The object was a rectangle $2(m) \times 1(m)$ with $M = 1(kg)$. 15 agents are distributed randomly around the perimeter employing the control law (14) and (19). The desired velocities are $v_x = t(m/s)$, $v_y = t(m/s)$, and $\omega = 0.1t(rad/s)$. The communication network graph is a ring (i.e. the set of edges E contains only $e_{i(i+1)}$ for all $i < N-1$ and e_{1N}). Figure 6 show each agent's estimate of the swarm's configuration (moments $\hat{m}^x(t)$, $\hat{m}^y(t)$, $\hat{m}^{zz}(t)$); while Figure 7 shows that the determinant of BB' is non-zero for all but the initial estimates, implying that the distributed system converges to the desired velocity. Figure 8 shows another example scenario. Here, the desired velocity causes the object to move in a circular pattern. The figure illustrates the path of the object's center of mass, and shows the force vector of each agent.

Remark 4.2: It is more difficult to accommodate a friction-based contact-model in this scenario because, instead of simply depending on N (a constant), the controller must estimate the number of agents who are capable of pushing the object, subject to friction constraints, at any given instant in time. This is a topic of ongoing work.

V. Discussion of Information Requirements

Regarding the constant velocity control law in Sect. III, each agent needs to know:

- the desired velocities v_d and ω_d;
- the actual velocity of the object v and ω (alternatively, its own velocity v_i);
- and its own location relative to the center of mass r_i.

No knowledge of M, J, the number of agents N, or the other agent's actions are required for stable velocity control. Regarding the rate of convergence, note from (10) that adding more agents never decreases the convergence rate – and generally improves it.

The conservative bound in (10) stems from the case depicted in Figure 3. In this case the error signals e_v and e_ω are essentially parsimonious requirements for agent i; however, for agents on the far side of the midline e_v and e_ω represent competing requirements.

Regarding the time varying velocity control law in Sect. IV, in addition to the information requirements for the constant velocity control law, each agent needs knowledge of:

- the object's mass M and inertia J;
- the desired accelerations $\dot{v}_d$ and $\dot{\omega}_d$; and
- the number of agents in contact with the object N.

Finally the rate of convergence of the estimates $\hat{m}$ is related to the second smallest Eigenvalue of the graph LaPlacian, $\lambda_2(L_G)$, also known as the algebraic connectivity of the graph – a measure of how strongly connected the graph is. The ring topology used in the example has rather weak connectivity (small $\lambda_2(L_G)$) while a complete graph has very strong connectivity. In general adding links increases $\lambda_2(L_G)$ improving the convergence rate of $\hat{m}$. While we only consider networks with static topology and no network delays, analogous protocols have been defined for networks where those assumptions are relaxed [11]. They have been shown to have similar convergence properties and could be applied to this problem.

VI. Conclusion

In this paper we consider cooperative manipulation problems where a large group (swarm) of non-articulated mobile robots is trying to cooperatively control the velocity of some larger rigid body by exerting forces around its perimeter. We consider a second order dynamic model for the object but use a simplified contact model. We present two asymptotically stable control laws. In the case of a constant desired velocity and no drag, it is shown that no coordination is required between the swarm members and no knowledge of the dynamic parameters of the object is needed. For more complex trajectories, and drag forces, we introduce a decentralized feed-forward component that requires some knowledge of the object's parameters and the swarm's geometric configuration. An online distributed consensus protocol is used to estimate swarm's configuration. An area of future work is to employ a distributed method to estimate the number of agents in contact with the object, N, online – a *census* algorithm [6].

REFERENCES

[1] B.E. Bishop. Swarm-based object manipulation using redundant manipulator analogs. In *IEEE International Conference on Robotics and Automation*, pages 1495 – 1500, 2008.

[2] J.M. Esposito. Distributed grasp synthesis for swarm manipulation. In *IEEE International Conference on Robotics and Automation*, pages 1489 –1494, 2008.

[3] J.M. Esposito, M.G. Feemster, and E. Smith. Cooperative manipulation on the water using a swarm of autonomous tugboats. In *IEEE International Conference on Robotics and Automation*, pages 1501 – 1506, 2008.

[4] V. Gazi and K.M. Passiano. Stability analysis of swarms. *IEEE Transactions on Automatic Control*, 48(4):692–696, 2003.

[5] A.J. Ijspeert, A. Martinoli, A. Billard, and L.M. Gambardella. Collaboration through the exploitation of local interactions in autonomous collective robotics: The stick pulling experiment. *Autonomous Robots*, 11(2):149–171, 2001.

[6] D. Kostoulas, D. Psaltoulis, I. Gupta, K. Birman, and A. Demers. Decentralized schemes for size estimation in large and dynamic groups. In *Proceedings of the IEEE International Symposium on Network Computing and Applications*, pages 41–48, 2005.

[7] R. Kube and E. Bonabeau. Cooperative transport by ants and robots. *Robotics and Autonomous Systems*, 30(1-2):85–101, 2000.

[8] R. Murray, Z. Li, and S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.

[9] R. Olfati-Saber, J.A. Fax, and R.M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215 – 233, 2007.

[10] R. Olfati-Saber and R.M. Murray. Flocking with obstacle avoidance: cooperation with limited communication in mobile networks. In *IEEE Conference on Decision and Control*, pages 2022 – 2028.

[11] R. Olfati-Saber and R.M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520 – 1533, 2004.

[12] C.A. Parker, H. Zhang, and R. Kube. Blind bulldozing: Multiple robot nest construction. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2010–2015.

[13] G. Pereira, V. Kumar, and M. Campos. Decentralized algorithms for multi-robot manipulation via caging. *International Journal of Robotics Research*, 23:783 – 795, 2004.

[14] D. Rus. Coordinated manipulation of objects in the plane. *Algorithmica*, 19(1):129–147, 1997.

[15] P. Song and V. Kumar. A potential field based approach to multi-robot manipulation. In *IEEE International Conference on Robotics and Automation*, pages 1217–1222, 2002.

[16] D.J. Stilwell and J.S. Bay. Optimal control for cooperating mobile robots bearinga common load. In *IEEE International Conference on Robotics and Automation*, pages 58–63, 1994.

[17] A. Sudsang and J. Ponce. A new approach to motion planning for disc shaped objects in the plane. In *IEEE Conference on Robotics and Automation*, pages 1068–1075, 2005.

[18] T. Sugar and V. Kumar. Control of cooperating mobile manipulators. *IEEE Transactions on Robotics and Automation*, 18(1):94 – 103, 2002.

[19] H.G. Tanner, A. Jadbabaie, and G.J. Pappas. Stable flocking of mobile agents, part I: fixed topology. In *IEEE Conference on Decision and Control*, pages 2010 – 2015.

[20] Z.D. Wang, E. Nakano, and T. Takahashi. Solving function distribution and behavior design problem for cooperative object handling by multiple mobile robots. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, 33(5):537–549, 2003.