2008 IEEE International Conference on
Robotics and Automation
Pasadena, CA, USA, May 19-23, 2008

Multi-Robot Manipulation via Caging in Environments with Obstacles

Jonathan Fink, M. Ani Hsieh, and Vijay Kumar

Abstract— **We present a decentralized approach to multi-robot manipulation where the team of robots surround and trap an object and transport it, by dragging or pushing, to the goal configuration in an environment with obstacles. The proposed feedback controllers are obtained by sequentially composing vector fields or behaviors and are decentralized in the sense that robots do not exchange each other's state information. Rather, cooperative manipulation is achieved by relying solely on each robot's local information and a global knowledge of the task. We present computer simulations and experimental results obtained using our multi-robot testbed.**

I. Introduction

As the number of robots in a team increases, it is necessary to think of control paradigms where robots are programmed with simple but identical behaviors that can be realized with limited on-board computational, communication and sensing resources. We often see such swarming behaviors in biological systems composed of large numbers of organisms that individually lack either the communication or computational capabilities required for centralized control. Examples of these behaviors can be seen in the group dynamics of bee [1] and ant [2] colonies, bird flocks, and fish schools [3].

We are interested in developing strategies for multi-robot manipulation where coordination can be achieved in a decentralized manner without requiring knowledge of team size and robot identities. Such strategies provide robustness to failures and facilitate command and control. Additionally, we are interested in strategies that rely solely on local information that can be obtained with minimal communication and sensing requirements. This is especially critical as team sizes increase since it is often impractical for large numbers of robots to share information while providing individual robots with access to global information.

Previous approaches in cooperative manipulation mostly use the notions of force and form closure when manipulating large objects [4]–[6]. Force closure is a condition that implies the grasp can resist any external force applied to the object, while form closure can be seen as the condition that guarantees force closure without requiring the contacts to be frictional [7]. Generally, robots are the ones that induce contacts with the object, and, as such, are the only source of grasp forces. However, when external forces acting on the object, *i.e.* gravity and friction, are used in conjunction with contact forces to produce force closure, we get conditional force closure. It is possible to use conditional closure to transport an object by pushing it from an initial position to a goal position [8], [9]. Caging or object closure is a variation on the form closure theme. It requires the less stringent condition that the object be trapped/caged by the robots and confined to a compact set in configuration space. Motion planning for circular robots manipulating a polygonal object is considered in [10]. Decentralized control policies for a group of robots to move toward a goal position while maintaining a condition of object closure is presented in [11].

In our work, we build on [12], [13] and present multi-robot manipulation of non-circular objects and cooperative manipulation in environments with obstacles via the composition of caging behaviors with feedback controllers derived from a global navigation function. Furthermore, our strategy requires little communication since collision and obstacle avoidance are achieved by sensing the relative positions/velocities of neighbors. While we assume that agents are holonomic, it is possible to extend our methodology to include non-holonomic robots using feedback linearization techniques. We present experimental and simulation results for various scenarios using our multi-robot testbed.

This paper is divided into the following sections: In Section II, we formulate the problem and provide some background to our approach. Section III outlines the implementation of our control methodology. Experimental and simulation results for various scenarios are presented in Section IV. Discussion of our results and directions for future work are provided in Section V.

II. Problem Formulation

We consider a group of N planar, fully actuated robots each with kinematics given by

$$\dot{q}_i = u_i \tag{1}$$

where $q_i = (x_i, y_i)^T$ and u_i denote the i^{th} agent's position and control input. We assume the agents are disk–shaped with radius r_i and can localize themselves in some global coordinate frame. In addition, we assume agents are able to sense the proximity of their teammates and/or obstacles within the environment. Thus, the neighborhood of q_i is given by the range and field of view of the sensing hardware and we denote the set of neighbors in this region by Γ_i. For collision/obstacle avoidance purposes, we assume a circular influence range, R_i, such that collision/obstacle avoidance maneuvers are active when agents are within this range.

Our objective is to design control inputs to enable an N robot team to approach, surround, and transport an object,

This work was supported by ARO DAAD19-02-01-0383, W911NF-05-1-0219, W911NF-04-1-0148 and NSF CCR02-05336, IIS-0413138, IIS-0427313.

Jonathan Fink and Vijay Kumar are with the GRASP Laboratory, University of Pennsylvania, Philadelphia, PA 19104, USA {jonfink,kumar}@grasp.upenn.edu
M. Ani Hsieh is with the Engineering Department, Swarthmore College, Swarthmore, PA 19081, USA mhsieh1@swarthmore.edu

in an environment with circular obstacles, from one location to another. To achieve this, we propose to construct artificial potential functions, φ, that can stabilize the agents onto a one-dimensional boundary (curve) such that they cage the desired object and achieve closure by orbiting around the object, all the while avoiding collisions. Once closure is obtained, the robots would then transport the object from an initial position to a goal location using a global navigation function, Φ, while maintaining closure. We outline our methodology in the following section.

III. Methodology

A. Assumptions

Given a convex workspace $\mathcal{W}$, we assume the boundary, denoted by $\partial\mathcal{W}$, can be described by some implicit function of the form $s_{\mathcal{W}_0} - s_{\mathcal{W}}(x, y) = 0$. For the given $\mathcal{W}$, we denote the set of circular obstacles in $\mathcal{W}$ as $\mathcal{O}$ and let $q_k^{\mathcal{O}}$ and ρ_k be the center and the radius of each obstacle $k \in \mathcal{O}$ respectively. Furthermore, given an object whose centroid is denoted as q_{obj}, we assume there exists a smooth star shape, $\mathcal{S}$, whose boundary, $\partial\mathcal{S}$, is a smooth, regular, simple, closed curve of the form $s(x, y) = 0$ such that the desired object is contained within $\partial\mathcal{S}$. In general, we can always find such a shape $\mathcal{S}$ by considering the following two parameters of the object: define (1) $D_{min}(obj)$ as the smallest gap through which the object will fit, and (2) $D_{max}(obj)$ as the maximum distance between any two points on the object. Then, for any given object, the circular boundary with radius given by:

$$r_{cage} = \frac{1}{2} D_{max}(obj) + \max_i r_i + \epsilon, \tag{2}$$

where $\epsilon > 0$ is a constant scalar, will always contain the object. We refer to this circle as the *caging* circle.

Given the caging radius r_{cage} and a team of N robots each with radius $r_i > 0$ and influence range $R_i > 0$, we define $r = \max_{r_i \forall i} r_i$ and $R = \max_{R_i \forall i} R_i$. Our goal is to synthesize decentralized controllers that will allow the team to surround the object while avoiding collisions. Therefore, the length of $\partial\mathcal{S}$, L, will naturally impose an upper bound on the number of robots, e.g. $N_{max} > 0$, that can fit on this boundary. Additionally, for a given r_{cage} and $D_{min}(obj)$, there must be at least $N_{min} > 0$ number of agents to ensure object closure. Thus, we make the following assumptions and let $d(\cdot, \cdot)$ denote the Euclidean distance between any two positions in $\mathcal{W}$:

1) $N_{min} \leq N < N_{max}$;
2) $r_{cage} > R$;
3) $d_{min}(q_{obj}^0, \partial\mathcal{W}) > (r_{cage} + 2r)$, where q_{obj}^0 denotes the inital position of the obstacle;
4) $d_{min}(q_{obj}^0, q_k) > (r_{cage} + 2R)$ for all $k \in \mathcal{O}$;
5) $d_{min}(q_j^{\mathcal{O}}, q_k^{\mathcal{O}}) > (\rho_j + \rho_k + 2R)$ for all $j,k \in \mathcal{O}$; and
6) $d_{max}(q_{obj}, q_i) \leq (r_{cage} + r)$ for $i = 1, ..., N$ in the Transport behavior.

Assumption 1 ensures the agents be able to surround the object and achieve closure. Assumption 2 ensures convergence of the team to the boundary surrounding the object. Assumptions 3 and 4 ensure the object is initially located at a position where agents can surround and orbit it without colliding with $\partial\mathcal{W}$ or other obstacles in the environment. Assumption 5 ensures the agents will have the ability to maneuver around the obstacles in the environment. Finally, assumption 6 ensures the agents will be able to maintain closure when transporting the object around the environment.

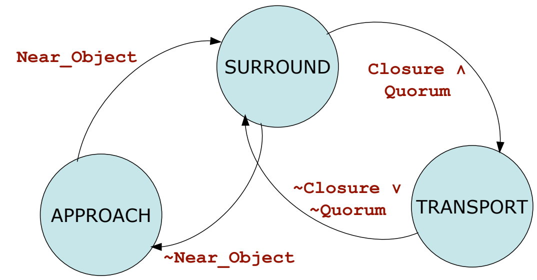


Fig. 1. Behavior Architecture. The software for each robot is identical and consists of several simple modes and the sequential composition of these modes.

B. Behaviors

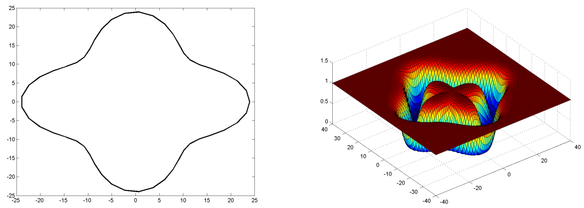
Our approach to caging and manipulation of objects can be summarized by the behavior architecture in Figure 1. The architecture relies on three behaviors.

1) **Approach**: The robot approaches the object while avoiding collisions with obstacles and other robots in the environment;
2) **Surround**: The robot stabilizes to a trajectory that orbits the object while avoiding collisions with other robots; and
3) **Transport**: The robot moves toward the goal configuration following a global navigation function or tracks a reference trajectory derived from the object's reference trajectory.

As shown in Fig. 1, transitions between behaviors are based on simple conditions derived from simple sensor abstractions. If a robot is near an object, a sensor sets its `Near_Object` flag causing the robot to switch to Surround mode. A `Quorum` flag is set based on the number of neighbors within its field of view. The `Closure` flag is set when the robot team surrounds the object. When `Closure` and `Quorum` are both set, the robot enters Transport mode and starts transporting the object. Thus, reseting the flags can cause the robot to regress into a different mode.

C. Shape Control

To enable the robots to surround and orbit the object to be manipulated while avoiding collisions with other robots and obstacles in the environment, we base our decentralized feedback controllers on the ones described in [12]. This controller has been shown to be scalable to large teams and stability and convergence properties have been established with collision and obstacle avoidance guarantees for a certain class of boundaries. We summarize the controller in this section.



(a) (b)

Fig. 2. (a) A star shape boundary described by $r-(20+\sin(5\theta+\pi/2)) = 0$. (b) The shape navigation function for the boundary given in (a) with a circular world boundary centered around $(0, 0)$ with a radius of 70.

Let $\partial\mathcal{S}$ be described by $s(x, y) = 0$ such that $s(x, y) < 0$ for all (x, y) in the interior of $\partial\mathcal{S}$ and $s(x, y) > 0$ for all (x, y) in the exterior of $\partial\mathcal{S}$, let $\gamma = s(x, y)$, and $\beta_0 = s^2_{\mathcal{W}_0} - s^2_{\mathcal{W}}(x, y)$, we define the *shape navigation function*, φ, as

$$\varphi(q) = \frac{\gamma^2}{[\gamma^2 + \beta_0]}. \tag{3}$$

By construction, φ is positive semi-definite, $\varphi = 0$ if and only if $s(x, y) = 0$, is uniformly maximal on $\partial\mathcal{W}$ and real analytic. Fig. 2 shows a star shaped boundary and its shape navigation function. The shape navigation function will generate an input to drive each agent towards $\partial\mathcal{S}$.

To enable the agents to surround and orbit the object in a counter-clockwise direction, let

$$\psi = \left[0,\, 0,\, \frac{\gamma}{\sqrt{\gamma^2 + \beta_0}}\right]^T$$

and we impose an additional input given by $-\nabla \times \psi$, where $\nabla \times \psi$ is a vector tangent to the level set curves of φ which drives each agent to travel along the boundary in a counter-clockwise direction. To enable avoidance of circular obstacles in the environment, a third input is included to drive the robots around any obstacles within their influence range. Thus, the decentralized shape control law is given by:

$$\begin{aligned} u_i &= -K_N \nabla_i \varphi_i \cdot f(N_i) - \nabla_i \times \psi_i \cdot g(T_i) \\ &\quad - \sum_{j \in \Gamma_i, \|v_j\|=0} \frac{\nabla_i \times \hat{\beta}_j}{d_{ij}^{p_2}} h(q_i, q_j) \end{aligned} \tag{4}$$

where ∇_i denotes differentiation with respect to agent i's coordinates, $p_2 > 0$ is a constant scalar, Γ_i denotes the set of neighbors for agent i, $d_{ij} = \|q_i - q_j\| - (r_i + r_j)$, $\varphi_i = \varphi(q_i)$, $\psi_i = \psi(q_i)$, and $\hat{\beta}_j$ is defined as:

$$\hat{\beta}_j = \left[0,\, 0,\, \frac{d_{ij}}{\sqrt{d_{ij}^2 + \beta_0}}\right]^T.$$

The functions $f(N_i), g(T_i) \in [0, 1]$ and $h(q_i, q_j) \in [-1, 1]$ are scalar functions used to independently modulate each term in (4). As such, each robot has the ability to either slow down or speed up depending on the positions of its neighbors and the obstacles in the environment. As such, the first term of (4) drives the agents towards $\partial\mathcal{S}$, the second term drives them along the level set curves of φ in a counter-clockwise direction, and the third term enables them to avoid collision with obstacles in their neighborhood.

We note since the φ and ψ is common among all agents, robots do not have to exchange information. Instead, the positions of the neighbors can be obtained via sensing alone.

D. Approach

In our methodology, the APPROACH behavior is characterized by a larger gain K_N on the descent component of (4) to yield agent trajectories that efficiently approach the object from a distance while avoiding collisions with other agents.

E. Surround

In the SURROUND mode, agents are near the desired shape and K_N is decreased so that the agents can distribute themselves around the object. Given enough robots, this behavior will lead to object closure. For a given r_{cage} and $D_{min}(obj)$, the minimum necessary number of robots to achieve object closure is

$$N_{min} = \frac{2\pi r_{cage}}{2r + D_{min}(obj)}. \tag{5}$$

Additionally, to ensure convergence to the boundary of the desired shape, the size of the team must be no greater than

$$N_{max} = \frac{\pi r_{cage}}{r}. \tag{6}$$

In practice, we do not always require this condition as the state transition events are robust to excess robots.

F. Transport

We propose two controllers for the TRANSPORT mode. The first one, relies on the parameterization of the smooth shape $s(x, y)$ with the reference trajectory. Given a trajectory for the object $(x^d_{obj}(t), y^d_{obj}(t))$, the reference trajectory for the shape is written as $s(x - x^d_{obj}(t), y - y^d_{obj}(t)) = 0$. The vector field for the robots is otherwise unchanged. The reference trajectory adds a time-varying component to the vector field that is computed independently by each robot.

The second controller relies on composing the first two terms of (4) with a descent direction derived from a global navigation function [14]. Given the workspace, $\mathcal{W}$, with the boundary $s_{\mathcal{W}_0} - s_{\mathcal{W}}(x, y) = 0$, the set of circular obstacles $\mathcal{O}$ in $\mathcal{W}$, and a goal configuration q_{goal}, let

$$\beta_k(q) = \left\|q - q^{\mathcal{O}}_k\right\|^2 - (\rho_k + 2r + r_{cage})^2,$$

where $k \in \mathcal{O}$. Then the global navigation function is given by

$$\Phi(q) = \frac{\|q - q_{goal}\|}{\left[\|q - q_{goal}\|^{2\kappa} + \Pi_{k=0}^{|\mathcal{O}|} \beta_k\right]^{1/\kappa}} \tag{7}$$

with the feedback controller for the team given by

$$u_{obj} = -\nabla_{obj} \Phi(q_{obj}). \tag{8}$$

Here, $\Phi(q)$ is constructed by expanding the boundaries of the obstacles and by decreasing the world boundary by $(2r + r_{cage})$ so as to ensure collisions do not occur between the team and the environment.

G. Composition of Behaviors

As described above, our multi-robot manipulation system is based on a decentralized shape controller and a global navigation function, both, with proven stability and convergence results [12], [14]. By varying certain properties of our controller, *i.e.* the sequential composition of (4) and (8), each mobile agent can operate in one of several modes: APPROACH, SURROUND, or TRANSPORT. Composition of these modes result in a global vector field such that, when combined with local interactions, achieves the desired task. Transitions between these modes, as well as exceptions in the case of failure, can be defined robustly while keeping individual agent decisions a function of local sensing.

In general, each agent's transition between modes will result from local observations of its neighbors as well as its distance to the manipulated object. An agent will initialize to the APPROACH mode if $D_{\text{obj}}(q_i) > D_{\texttt{near\_object}}$ (i.e. it is far from the object). As the agent approaches the desired caging shape, $D_{\text{obj}}(q_i) \leq D_{\texttt{near\_object}}$ will result in a transition to the SURROUND mode.

In the SURROUND mode, the orbiting term of the shape controller is favored so the robots are distributed around the object to be manipulated. Given at least N_{min} agents, this mode converges to an equilibrium where object closure is attained. While closure is a global property of the system, we propose an algorithm for local estimation of closure.[1].

1) Quorum: To locally define quorum, we introduce the concept of a *forward* neighborhood $\hat{\Gamma}^+$ and a *backward* neighborhood $\hat{\Gamma}^-$ with respect to the manipulated object and the SURROUND mode. For agent i, the SURROUND mode introduces an approach component, *i.e.* $\nabla_i \varphi_i$, and rotation component, *i.e.* $\nabla_i \times \psi_i$, so that we can define some set of robots to be *in front of* agent i and another set *behind*. If a neighborhood $\hat{\Gamma}_i$ represents the agents within a distance $D_{min}(obj)$, then

$$\hat{\Gamma}_i^{\pm} = \{j \in \Gamma_i \mid 0 < \pm(q_j - q_i)^T (\nabla_i \times \psi_i)\}. \qquad (9)$$

Furthermore, we can define agents from $\hat{\Gamma}_i^+$ and $\hat{\Gamma}_i^-$,

$$i^+ = \text{argmax}_{k \in \hat{\Gamma}_i^+} \frac{(q_k - q_i)^T \nabla_i \varphi_i}{\|q_k - q_i\|}, \qquad (10)$$

$$i^- = \text{argmax}_{k \in \hat{\Gamma}_i^-} \frac{-(q_k - q_i)^T \nabla_i \varphi_i}{\|q_k - q_i\|}, \qquad (11)$$

to be the adjacent agents in the potential cage around the object as depicted in Fig. 3.

We are interested in strategies requiring little to no communication among the agents, as such, consider the following update rule for $\texttt{quorum}_i$,

$$\texttt{quorum}_i = \begin{cases} 0 & \text{if } (\hat{\Gamma}_i^+ = \emptyset) \vee (\hat{\Gamma}_i^- = \emptyset), \\ N_{min} & \text{if } f(i^+, i^-) > N_{min}, \\ f(i^+, i^-) & \text{otherwise}, \end{cases} \qquad (12)$$

[1]Closure can also be determined in a distributed manner by computing homology groups for the network [15], [16]

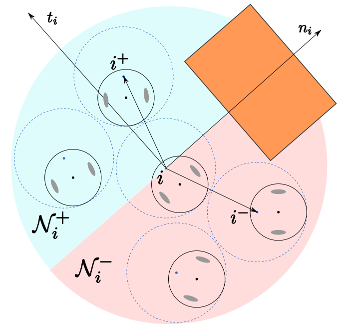


Fig. 3. Agent i's neighborhoods $\hat{\Gamma}_i^+$ and $\hat{\Gamma}_i^-$ with i^+ and i^-

with $f(j,k) = \min(\texttt{quorum}_j, \texttt{quorum}_k) + 1$. This strategy only requires the exchange of small bits of information and can be easily handled by agents with limited communication resources. We shall use $\texttt{quorum}_i$, $\texttt{quorum}_{i^+}$, and $\texttt{quorum}_{i^-}$ to determine when there is object closure.

2) Closure: If there is no closed loop around the object, $\texttt{quorum}_i$ will converge to the minimum of N_{min} and the shorter of the forward/backward chain of agents. On the other hand, if there is a complete loop around the object, $\texttt{quorum}_i$ will grow as large as the imposed bound N_{min}.

We define *local* closure to be

$$\begin{aligned} \texttt{closure}_i = &(\texttt{quorum}_i \geq N_{min}) \wedge \\ &(\texttt{quorum}_i = \texttt{quorum}_{i^+}) \wedge \\ &(\texttt{quorum}_i = \texttt{quorum}_{i^-}). \end{aligned} \qquad (13)$$

Our condition for *local* closure will coincide with *global* closure for any situation where up to $2N_{min}$ agents are used to form a cage around the object.

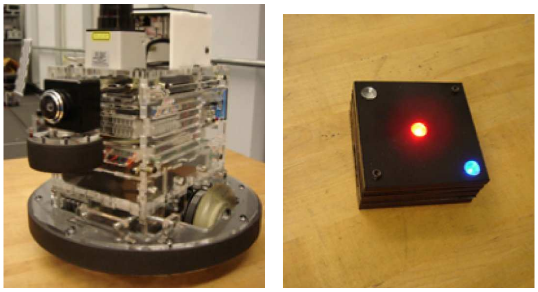
When an agent estimates that *local* closure has been attained, it will switch to the TRANSPORT mode and begin manipulation of the object. Should `closure` be lost during manipulation, each agent in the system will return to the SURROUND mode to reaquire the object.

IV. EXPERIMENTAL RESULTS

We illustrate the proposed controller with some experimental and simulation results.

A. Experimental Setup

Our experimental testbed [13], consists of a number of ground robots and an overhead tracking system that provides localization information to each agent. The SCARAB differential drive robot, shown in Fig. 4(a), is equipped with a 1 GHz on-board computer, power management, and 802.11a/b/g wireless communication capabilities. Eight color cameras and two processing computers make up the overhead tracking system which track LED markers on top of each robot as shown in Fig. 4(b). The marker is programmable and able to transmit a unique identification code via its blinking LEDs. Our entire system is connected using the PLAYER/STAGE/GAZEBO project architecture [17] so that identical code modules can be used on all the robots whether



(a) (b)

Fig. 4. The $20 \times 13.5 \times 22.2$ cm^3 SCARAB platform is shown in Figure 4(a). Figure 4(b) depicts a LED target used for localization.

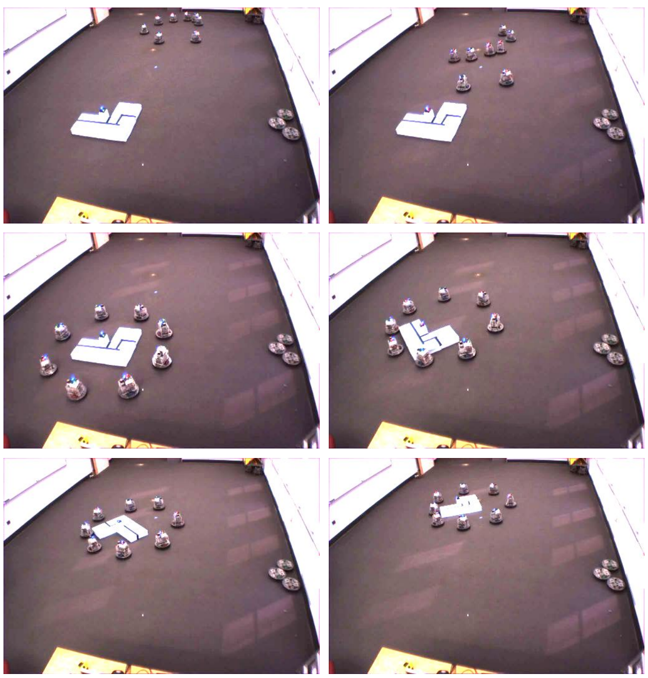
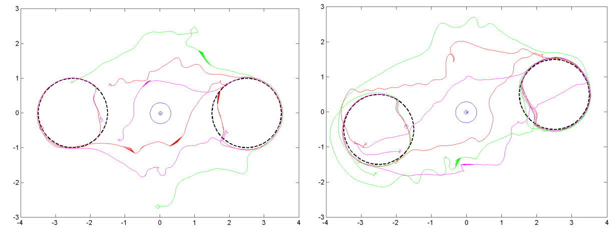


Fig. 5. Eight robots manipulating an L–shaped object in an environment with no obstacles.



(a) (b)

Fig. 6. Four robots with $r = 30$ cm, converging to two circular boundaries each with radius of 1 m. Dotted black lines denote boundaries. Solid lines denote robot trajectories. $\times$s denote initial positions. $\bigcirc$s denote final positions. Middle circle denotes the obstacle, *i.e.* fourth robot.

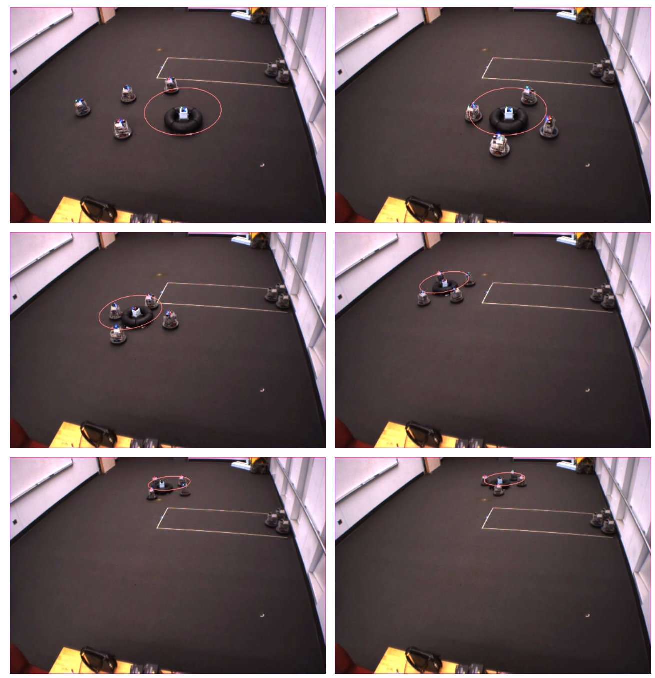


Fig. 7. Four robots manipulating a circular object around an imaginary wall whose boundary is shown by the white lines in the ground plane. The overlayed circle denotes the boundary the robots are converging to.

in experimentation or simulation. All the experiments described here were conducted at the GRASP Lab's SWARMS Robot Playpen. A schematic of the space is shown in Fig. 8.

B. Experimental Results

Our first experiment, shown in Fig. 5, consisted of eight robots approaching, surrounding, and transporting an L–shaped object in an environment with no obstacles. As such, the robots are only concerned with caging the object while avoiding collisions with each other during the task. The L–shaped object is manipulated using the first TRANSPORT methodology described in Section III-F.

The next set of experiments, shown in Fig. 6, involved four robots converging onto a circular boundary while avoiding collisions with obstacles in the environment represented by the fourth robot which has been immobilized due to failure. Fig. 6(a) shows the trajectories of the team converging first to a circular boundary centered at $(2.5, 0)$ m and then to a circular boundary centered at $(-2.5, 0)$, each with a radius of 1 m. Similarly, Fig. 6(b) shows the trajectories of the same team of robots converging first to a circular boundary centered at $(2.5, -0.5)$ m and then to a circular boundary centered at $(-2.5, 0.5)$, each with a radius of 1 m. In these experiments, each agent executed the controller given by (4).

Our next experiment, shown in Fig. 7, consisted of a team of four robots approaching, surrounding, and transporting a circular object in an environment with obstacles. Fig. 8 shows the trajectory of the object as it was manipulated around an imaginary wall inside the Playpen. The shaded region around the trajectory denotes the area occupied by the object and the robots during the execution of the task. The number of agents executing the behaviors APPROACH, SURROUND, and TRANSPORT during the experiment is shown in Fig. 9.

Lastly, the simulation results for four robots transporting the circular object from various initial positions within the Playpen into the hallway are shown in Fig. 10.

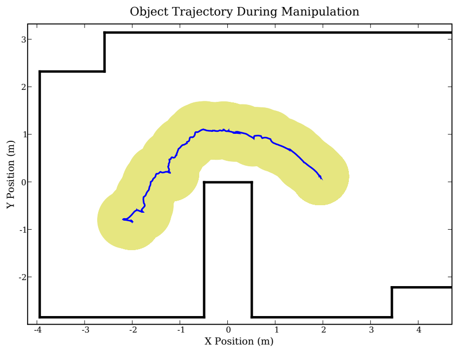


Fig. 8. Object trajectory and area occupied by the object and the robots during task execution.

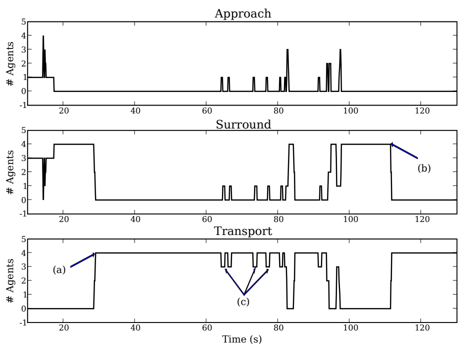


Fig. 9. The number of agents in each mode over the duration of the experiment. Marker (a) points to the first instance when all agents have entered the TRANSPORT mode. Marker (b) denotes a time when closure was lost and the robots attempt to re–acquire the object. Marker (c) identifies instances where individual robots detect loss of closure, however, closure is regained before the object escapes.

V. DISCUSSION

We have presented a decentralized strategy for multi-robot caging and manipulation. The team of robots form patterns to trap an object and manipulate it by dragging or pushing it to the goal configuration. Our decentralized controllers for multi-robot caging and manipulation are obtained via sequential composition of vector fields or behaviors, *i.e.* APPROACH, SURROUND, and TRANSPORT, each with proven stability, convergence, collision, and obstacle avoidance guarantees [12], [14]. We presented simulation and experimental results using our multi-robot test-bed.

While we have not shown formal guarantees for our system, the experimental and simulation results do in fact show the robustness of our methodology. However, one direction for future work is to establish stability and convergence properties for our switched controller. We believe this can be achieved by first extending the results for the APPROACH and SURROUND behaviors to time varying boundaries. Once this has been established, we should be able to employ Lyapunov theory to show that the multi-robot system can, in fact, stably manipulate the object to the desired goal configuration. Additionally, we would like to formally show that global consensus can in fact be reached using our quorum based approach. Lastly, we would like to conduct more experiments

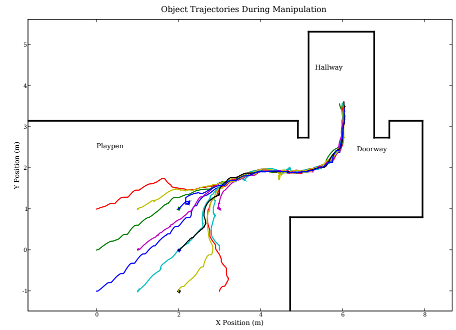


Fig. 10. Object trajectories for various initial positions.

in other environments with obstacles to further demonstrate the robustness of our approach.

REFERENCES

[1] N. F. Britton, N. R. Franks, S. C. Pratt, and T. D. Seeley, “Deciding on a new home: how do honeybees agree?” *Proc R Soc Lond B*, vol. 269, 2002.

[2] S. C. Pratt, “Quorum sensing by encounter rates in the ant temnothorax albipennis,” *Behavioral Ecology*, vol. 16, no. 2, 2005.

[3] J. K. Parrish, S. V. Viscido, and D. Grunbaum, “Self-organized fish schools: An examination of emergent properties,” *Biol. Bull.*, vol. 202, June 2002.

[4] K. Kosuge, Y. Hirata, H. Asama, H. Kaetsu, and K. Kawabata, “Motion control of multiple autonomous mobile robot handling a large object in coordination,” in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, Detroit, Michigan, May 1999, pp. 2666–2673.

[5] D. Rus, “Coordinated manipulation of objects in a plane,” *Algorithmica*, vol. 19, no. 1, pp. 129–147, 1997.

[6] T. Sugar and V. Kumar, “Multiple cooperating mobile manipulators.” in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, Detroit, Michigan, May 1999, pp. 1538–1543.

[7] J. K. Salisbury and B. Roth, “Kinematic and force analysis of articulated hands,” *ASME Journal of Mechanisms, Transmissions, and Automation in Design*, vol. 104, no. 1, pp. 33–41, 1982.

[8] M. J. Mataric, M. Nilsson, and K. Simsarian, “Cooperative multi-robot box-pushing,” in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Pittsburgh, Pennsylvania, August 1995, pp. 556–561.

[9] K. M. Lynch and M. T. Mason, “Stable pushing: Mechanics, controllability, and planning,” *The Int. Journal of Robotics Research*, vol. 16, no. 6, pp. 533–556, 1996.

[10] A. Sudsang and J. Ponce, “Grasping and in-hand manipulation: Experiments with a reconfigurable gripper,” *Advanced Robotics*, vol. 12, no. 5, pp. 509–533, 1998.

[11] G. A. S. Pereira, V. Kumar, and M. F. Campos, “Decentralized algorithms for multi-robot manipulation via caging,” *The Int. Journal of Robotics Research*, vol. 23, no. 7/8, pp. 783–795, 2004.

[12] M. A. Hsieh, “Communication maintenance and motion control strategies for robot teams in environments with obstacles,” Ph.D. dissertation, Univ. of Pennsylvania, Philadelphia, PA, August 2007.

[13] J. Fink, N. Michael, and V. Kumar, “Composition of vector fields for multi-robot manipulation via caging,” in *Proc. 2007 Robotics: Science and Systems III*, Atlanta, GA, 2007.

[14] E. Rimon and D. E. Koditschek, “Exact robot navigation using artificial potential functions,” *IEEE Transactions on Robotics and Automation*, vol. 8, no. 5, pp. 501–518, October 1992.

[15] V. de Silva and R. Ghrist, “Homological sensor networks,” *Notices of the American Mathematical Society*, vol. 54, no. 1, pp. 10–17, 2007.

[16] A. Muhammad and A. Jadbabaie, “Decentralized computation of homology groups in networks by gossip,” in *Proc. of the American Control Conf.*, New York, July 2007, To Appear.

[17] B. Gerkey, R. T. Vaughan, and A. Howard, “The Player/Stage project: Tools for multi-robot and distributed sensor systems,” in *Proc. of the Int. Conf. on Advanced Robotics*, Coimbra, Portugal, June 2003, pp. 317–323.