Object Manipulation and Position Control Using a Swarm With Global Inputs

Shiva Shahrokhi and Aaron T. Becker

***Abstract*— Manipulating objects with a swarm of simple robots with global control inputs is difficult because they are highly under actuated, the number of robots in contact with object changes dynamically, the robots contacting each other changes dynamically, and the robots disperse over time and must be recollected.**

Micro and nano robots are suited for targeted drug delivery and micro scale manufacturing because they are small enough to navigate the passageways of the body. However, due to their small size, robots cannot contain onboard processing for autonomy nor onboard power. Instead they are controlled by an external signal such as a magnetic field. Because each robot can only provide a small amount of force or transport a small amount of material, large swarms of robots are required, all controlled by the same external field.

This work presents controllers and algorithms for steering such an under-actuated swarm to manipulate objects. Previous work showed that mean and variance of the swarm is controllable. This was used to manipulate convex polygonal objects through a simple maze. A key remaining challenge is controlling the torque applied to an object. Torque control is necessary for manipulating objects as well as for aligning sensors or emitters, and for exerting forces. This work first proves that swarm torque control is possible, then presents algorithms to automate the task. The paper concludes with experimental results using 100 hardware robots to manipulate rectangles with large aspect ratios.

I. Introduction

Micro- and nano-robots can be manufactured in large numbers. Our vision is for large swarms of robots remotely guided 1) through the human body, to cure disease, heal tissue, and prevent infection and 2) ex vivo to assemble structures in parallel. For each application, large numbers of micro robots are required to deliver sufficient payloads, but the small size of these robots makes it difficult to perform onboard computation. Instead, these robots are often controlled by a global, broadcast signal. The biggest barrier preventing these implementations is a lack of control techniques that can reliably exploit large populations despite significant underactuation.

Previous work proved that the mean position of a swarm is controllable and that, with an obstacle, the swarm's position variance orthogonal to rectangular boundary walls is also controllable (these are σ_x and σ_y for a workspace with axis-aligned walls). The usefulness of these techniques was demonstrated by several automatic controllers. One controller steered a swarm of robots to push a larger block through a 2D maze [1]. We also showed methods to control a

S. Shahrokhi and A. Becker are with the Department of Electrical and Computer Engineering, University of Houston, Houston, TX 77204-4005 USA {sshahrokhi2, atbecker}@uh.edu

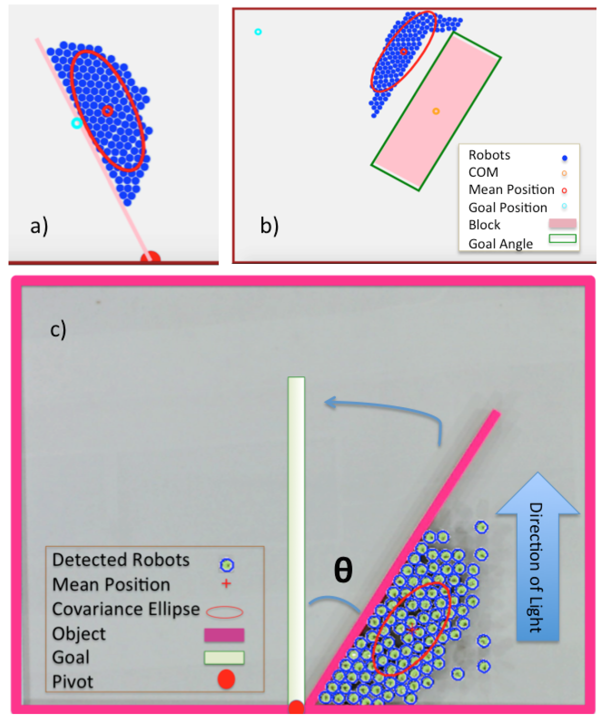

Fig. 1. Torque control of an object is essential for manipulating objects to their goal position when there are narrow passageways and for aligning sensors, emitters, or other objects. This paper provides feedback control laws to achieve required torques and forces using a highly under-actuated system where all robots are controlled globally with the same input. (a) Simulation of robots exerting torque on a hinged "door". (b) Simulation of swarm orientation manipulation (c) 100 hardware robots applying torque to an object. These robots have light sensors and are programmed to move toward the brightest light in the room. This light is a shared control input that operates globally on the swarm.

swarm's position covariance to enable navigating the swarm through a workspaces with narrow corridors. Another challenge remains. Object manipulation often requires controlling torque on an object, for example changing the orientation of an object with a large aspect ratio to thread through narrow corridors. Torque control is also necessary for a variety of alignment tasks from retroreflectors, mirrors for solar incinerators, to targeted radiation therapy.

Accurate torque control is difficult. Predicting the force a swarm exerts on an object requires knowing the location of every robot in contact with the object. This information is often hard to gather, because remote sensing tiny particles is difficult. Particles may be smaller than the minimum resolution of MRI, PET, and ultra sound, but these sensing modalities can still return aggregate data. Form this aggregate data, some statistics—including mean and variance—are easy to obtain. A swarm with n agents has $2n$ degrees of freedom. This swarm, when steered toward an object, begins interacting with object at different times. The number of robots touching this object as a function of time is difficult to predict and often impossible to directly measure. Stochastic effects make long-term prediction challenging. Even when it is possible to predict which agents will hit the object first, as agents interact with the object, the swarm's configuration changes. The challenge is not only limited to swarm-object interaction, but also to swarm-swarm interactions when the swarm self-collides or is split into multiple components. As a result, the force the swarm will exert on the object is not easy to predict. Rather than design open-loop algorithms, this paper focuses on feedback control strategies using just two statistics of the swarm, the mean and variance of the swarm's position.

II. Related Work

Robotic manipulation by pushing has a long and successfull history [2]–[5]. Key developments introduced the notion of a friction cone and stable pushes. A *stable push* is a pushing operation by a robot with a flat plate pushing element in which the object does not change orientation relative to the pushing robot [2]. The *friction cone* is the set of vector directions a robot in contact with an object can push that object with a stable push.

add refs

Unlike *caging* manipulation, where robots form a rigid arrangement around an object [6], [7], our swarm of robots is unable to grasp the blocks they push, and so our manipulation strategies are similar to *nonprehensile manipulation* techniques, e.g. [2], where forces must be applied along the center of mass of the moveable object. A key difference is that our robots are compliant and tend to flow around the object, making this similar to fluidic trapping [8], [9].

Controlling the *shape*, or relative positions, of a swarm of robots is a key ability for a myriad of applications. Correspondingly, it has been studied from a control-theoretic perspective in both centralized, e.g. virtual leaders in [10], and decentralized approaches, e.g. control-Lyapunov functions gradient based decentralized controllers in [11]. Most approaches assume a level of intelligence and autonomy in the individual robots that exceeds the capabilities of current micro- and nano-robots [12], [13].

Instead, this paper focuses on centralized techniques that apply the same control input to each member of the swarm, as in [14].

III. Torque Control

A key feature of an object is the orientation of its major axis. The orientation is especially relevant when a swarm is manipulating a non-symmetric object, in narrow corridors. Orientation is controllable by applying torque to the object. In Eq. 1, to change the output torque τ, we can choose the direction and magnitude of the force applied f, and r, the moment arm from the object's center of mass (COM) to the point of contact.

$$\tau = F \times r \tag{1}$$

The swarm version of (1) is the summation of the forces contributed by individual robots.

$$\tau_{total} = \sum_{i=1}^{n} \rho_i F_i \times (P_i - O) \tag{2}$$

$$F_{total} = \sum_{i=1}^{n} \rho_i F_i \tag{3}$$

Here F_i is the force that the ith robot applies. If all robots are identical and the control input is uniform, the force is equivalent for every robot and $F_i = F_c$. Not all robots are in contact with the object. ρ_i is an indicator variable. ρ_i is 1 if the robot is in direct contact with the object, or touching a chain of robots where at least one robot is in contact with the object. Otherwise $\rho_i = 0$. The moment arm is the robot's position P_i to the object's COM O.

IV. Simulation

This section examples four main challenges for pose and torque control of an object, arranged in increasing difficulty. Each task uses a PD controller that uses the mean position and mean velocity to regulate the swarm's mean position, as in [1]. The control input is the global force applied to each robot:

$$\begin{aligned} u_x &= K_p(goal_x - \bar{x}) + K_d(0 - \bar{v}_x) \\ u_y &= K_p(goal_y - \bar{y}) + K_d(0 - \bar{v}_y) \end{aligned} \tag{4}$$

here K_p is the proportional gain, and K_d is the derivative gain. The swarm's average position is $(\bar{x}, \bar{y})$ and mean velocity is $(\bar{v}_x, \bar{v}_y)$. Each task uses a different algorithm to select the swarm's goal position $(goal_x, goal_y)$.

a) Pure torque control: An object with a pivot point can rotate, but not translate. A door is a common example of an object with a pivot. A door can have an angular velocity but cannot translate. If there was only one robot touching the object, the robot should push at the point which maximizes the moment arm, at the extreme end of the object furthest from the pivot point. The optimal pushing location provides the maximum force, because it maximizes r in (1). However, given a swarm of robots, maximizing r is no longer the optimal solution. If the swarm hit the object with its mean position at the extreme edge, half of the robots will miss the object and the swarm will be torn apart. Because few robots remain, the force is significantly decreased and torque is not maximized. In our simulation, the swarm apples torque until the swarm's mean position is beyond the object. At this point, the swarm will be regathered in a corner, a time consuming task. The key parameter of interest for a hinged door of length L is C/L, where C is the position along the door where the mean of the swarm will push. The swarm is directed toward

$$\begin{aligned} goal_x &= O_x + C\sin(O_\theta - \theta_{goal}) \\ goal_y &= O_y + C\cos(O_\theta - \theta_{goal}) \end{aligned} \tag{5}$$

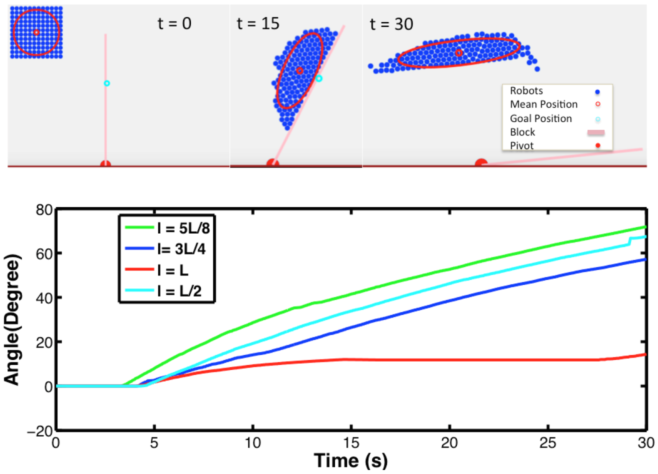

Fig. 2. Simulation results from a swarm applying force to a hinged door. The swarm mean is steered toward a point C units along the object from the pivot point. Simulation used 144 robots of diameter XX with a standard deviation of less than 1.5 m and an object length of 6 m.

Fig. 2 illustrates how different values of C result in different rates of turning. These simulations tested $C = \{1/2, 3/4, 5/8, 1\}L$ The fastest turning rates occurred with $C = 5/8L$.

b) Pure translation of the object: These simulations used a uniform density rectangle as the object. This object was X??X× larger than the robots. When the total force is applied perpendicular to the object and in line with the center of mass, according to Eq. 1 there will be no torque. The following goal position for the mean position of the swarm regulates the object's orientation using $\Delta\theta$ for proportional feedback to determine where to apply force. $\Delta_\theta = goal_\theta - O_\theta$ is the difference between the goal angle and the current object angle. K_τ is a constant and (O_x, O_y) is the position of the object's COM.

$$\begin{aligned} goal_x &= O_x \\ goal_y &= K_\tau \Delta_\theta + O_y \end{aligned} \tag{6}$$

Fig. 3 shows how $\Delta\theta$ converges to zero with different initial configurations of the swarm. When the swam is above or below of the object the swarm applies a torque to the object.

c) Orientation of the object: Using the pure torque control discussed in the previous paragraph, the orientation of the object can be controlled by applying force. The rectangular object is not pivoted, so it moves in addition to rotating. The swarm still may split into multiple components. We use the hysteresis variance control from [1] to gather the swarm when its variance grows too large. The following control law chooses a goal position to regulate the orientation of the object. In the following equation, O_θ is the orientation of the object's major axis. The object COM is at (O_x, O_y).

Let θ_O = the orientation of the object's major axis, measured from the world x-axis.

$$\begin{aligned} goal_x &= O_x + K_{orient}(\theta_O - \theta_{desired})\cos(\theta_O) \\ goal_y &= O_y + K_{orient}(\theta_O - \theta_{desired})\sin(\theta_O) \end{aligned} \tag{7}$$

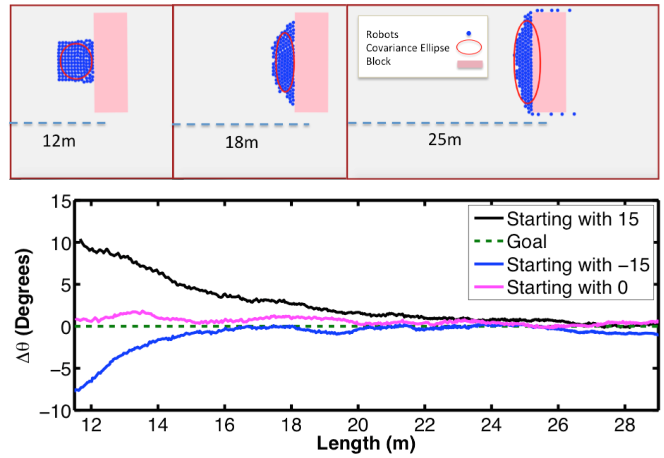

Fig. 3. In this task, the swarm pushed the object in the $+x$ direction while trying to regulate the orientation to $goal_\theta = 0°$. The swarm can push the object without changing its orientation only if it pushes along a line intersecting the COM of the object. A feedback control law regulates the object's orientation.

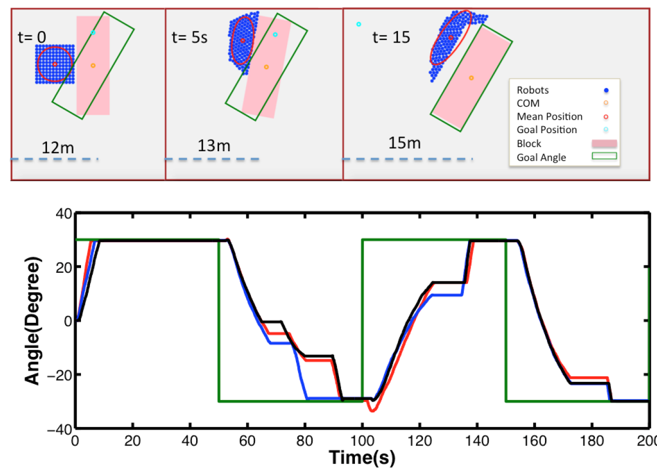

Fig. 4. Plot demonstrating orientation control of a rectangular object. The green line is the goal orientation. When the plot traces are constant the swarm is no longer pushing the object and instead is being regathered in a corner of the workspace until the variance is below a desired threshold.

Here K˙orient is a positive gain on the control input.

Fig. 4 illustrates this controller with different starting positions. When the plot traces are constant the swarm is no longer pushing the object and instead is being regathered in a corner of the workspace.

d) Object pose control: TODO

write this section. We need a simulation

V. EXPERIMENT

Our experiments are on centimeter-scale hardware systems called *kilobots*. These allows us to emulate a variety of dynamics, while enabling a high degree of control over robot function, the environment, and data collection. The kilobot, from [15], [16] is a low-cost robot designed for testing collective algorithms with large numbers of robots. It is available as an open source platform or commercially from [17]. Each robot is approximately 3 cm in diameter, 3 cm tall, and uses two vibration motors to move on a flat surface at speeds up to 1 cm/s. Each robot has one ambient light sensor that is used to implement *phototaxis*,

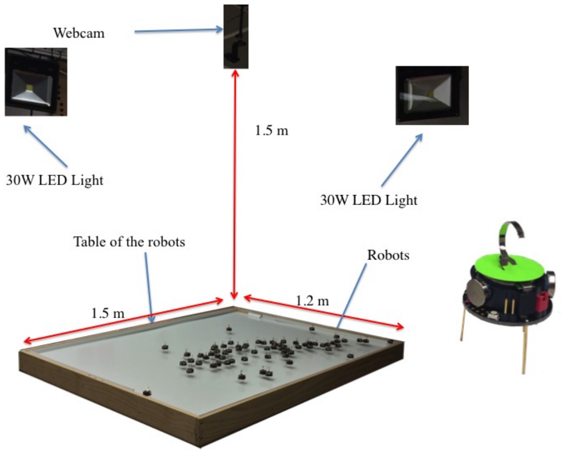

Fig. 5. Hardware platform: table with 1.5×1.2 m workspace, surrounded by eight remotely triggered 30W LED floodlights, with an overhead machine vision system.

moving towards a light source. In these experiments as shown in Fig. 5, we used n=97 kilobots, a glass-covered 1.5 m×1.2 m whiteboard as the workspace, and eight 30W LED floodlights arranged 1.5 m above the plane of the table at the $\{N, NE, E, SE, S, SW, W, NW\}$ vertices of a 6 m square centered on the workspace. The lights were controlled using an Arduino Uno board connected to an 8-relay shield. Above the table, an overhead machine vision system tracks the position of the swarm.

VI. Conclusion and Future Work

This paper presented techniques for controlling the orientation of an object by manipulating it using a swarm of simple robots with global inputs. The paper provided algorithms for precise orientation control, as well as demonstrations of orientation control.

Future efforts should be directed toward optimizing torque control, applying the techniques to hardware robots, pose control for multiple part assembly, and manipulation in a crowded workspace. The control laws in this paper used only the mean and variance of the swarm. The control techniques may be optimized using high-order moments, or by stochastic modeling of the collisions between swarm members and the object.

References

[1] S. Shahrokhi and A. T. Becker, "Stochastic swarm control with global inputs," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2015, p. tbd.

[2] K. Lynch, "Locally controllable manipulation by stable pushing," *IEEE Trans. Robot. Autom.*, vol. 15, no. 2, pp. 318–327, Apr. 1999.

[3] K. M. Lynch, "Nonprehensile robotic manipulation: Controllability and planning," Ph.D. dissertation, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213, Mar. 1996.

[4] S. Akella, W. H. Huang, K. M. Lynch, and M. T. Mason, "Parts feeding on a conveyor with a one joint robot," *Algorithmica*, vol. 26, no. 3, pp. 313–344, 2000.

[5] J. D. Bernheisel and K. M. Lynch, "Stable transport of assemblies by pushing," *IEEE Transactions on Robotics*, vol. 22, no. 4, pp. 740–750, Aug. 2006.

[6] A. Sudsang, F. Rothganger, and J. Ponce, "Motion planning for disc-shaped robots pushing a polygonal object in the plane," *IEEE Trans. Robot. Autom.*, vol. 18, no. 4, pp. 550–562, Aug. 2002.

[7] J. Fink, N. Michael, and V. Kumar, "Composition of vector fields for multi-robot manipulation via caging," in *Robotics Science and Systems*, Atlanta, GA, Jun. 2007.

[8] M. D. Armani, S. V. Chaudhary, R. Probst, and B. Shapiro, "Using feedback control of microflows to independently steer multiple particles," *Journal of Microelectromechanical systems*, vol. 15, no. 4, Aug. 2006.

[9] A. Becker, R. Sandheinrich, and T. Bretl, "Automated manipulation of spherical objects in three dimensions using a gimbaled air jet," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, oct. 2009, pp. 781 –786. [Online]. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5354427

[10] M. Egerstedt and X. Hu, "Formation constrained multi-agent control," *IEEE Trans. Robotics Automat.*, vol. 17, pp. 947–951, 2001.

[11] M. A. Hsieh, V. Kumar, and L. Chaimowicz, "Decentralized controllers for shape generation with robotic swarms," *Robotica*, vol. 26, no. 05, pp. 691–701, 2008.

[12] S. Martel, "Magnetotactic bacteria for the manipulation and transport of micro-and nanometer-sized objects," *Micro-and Nanomanipulation Tools*, 2015.

[13] X. Yan, Q. Zhou, J. Yu, T. Xu, Y. Deng, T. Tang, Q. Feng, L. Bian, Y. Zhang, A. Ferreira, and L. Zhang, "Magnetite nanostructured porous hollow helical microswimmers for targeted delivery," *Advanced Functional Materials*, vol. 25, no. 33, pp. 5333–5342, 2015. [Online]. Available: http://dx.doi.org/10.1002/adfm.201502248

[14] A. Becker, G. Habibi, J. Werfel, M. Rubenstein, and J. McLurkin, "Massive uniform manipulation: Controlling large populations of simple robots with a common input signal," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Nov. 2013, pp. 520–527.

[15] M. Rubenstein, C. Ahler, and R. Nagpal, "Kilobot: A low cost scalable robot system for collective behaviors," in *IEEE Int. Conf. Rob. Aut.*, May 2012, pp. 3293–3298.

[16] M. Rubenstein, A. Cornejo, and R. Nagpal, "Programmable self-assembly in a thousand-robot swarm," *Science*, vol. 345, no. 6198, pp. 795–799, 2014.

[17] K-Team, "Kilobot, www.k-team.com/mobile-robotics-products/kilobot," 2015.