Kinematic Multi-Robot Manipulation with no Communication Using Force Feedback

Zijian Wang[1] and Mac Schwager[2]

***Abstract*— This paper proposes a novel decentralized algorithm that coordinates the forces of a group of robots during a cooperative manipulation task. The highlight of our approach is that no communication is needed between any two robots. Our underlying intuition is that every follower robot can measure the direction of the movement of the object and then applies its force along that direction to reinforce the movement. We prove that using our algorithm, all followers' forces will synchronize to the direction of the force applied by one leader robot, who guides the robotic fleet to its destination. We first verify our algorithm by simulation in a physics engine, where 20 robots transport a chair collectively. We then validate our algorithm in hardware experiments by building four low-cost robots, equipped with force and velocity sensors, to transport a cardboard box in a laboratory environment. In addition, our algorithm allows the leader to be a human, and we also demonstrate the human-swarm cooperation in our manipulation experiments.**

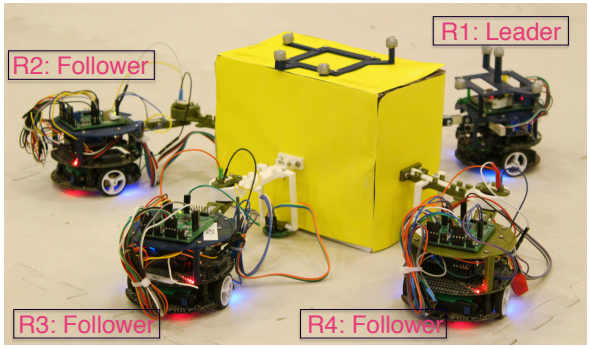


Fig. 1. Four robots manipulate a yellow cardboard box together. Only the leader is tracked by the OptiTrack system and knows the destination. There is no communication among all the robots and the follower robots do all the sensing and computation onboard.

I. Introduction

If an object is too heavy for a single robot, can we move it using a group of robots? Even more challenging, can the robots collaborate to move the object without communication? The answer is obviously yes to the first question, but maybe obscure for the second one. In this paper, we propose an algorithm that uses no communication to address this multi-robot manipulation problem. Before giving technical details, let us start with an intuitive example to explain our method. Imagine that your friend and you want to move a heavy table together. Your friend holds the table in the front while you grab the rear side. Your eyes will be covered by a piece of cloth so that you cannot see, and you are not allowed to talk to your friend. Now if your friend starts to move forward, your hand will feel a pulling force, and you will sense that the table is moving forward. Then although you cannot communicate with your friend, you can get indirect information from your hand that your friend wants to move the table forward, so you can push forward to contribute your force. Likewise, for any motion your friend initiates (forward, back, turn left or right), your hand will give you local force and motion information, which you can use to determine how to apply your own force. In this way, your friend and you can manipulate the table together, without any direct communication.

In this paper, we propose an algorithm that lets the robots behave as in the previous example. We have two types of robots: (1) many follower robots that do not know their own global position, the desired path or destination of the object, nor any information about their peer robots; (2) one leader robot who knows the destination but has no information about the follower robots. The followers collaborate with the other robots using the local measurement of the velocity of the object, and then apply their forces along the direction of the object's velocity. The local measurement works as an indirect means of communication for force coordination. A leader robot, who changes its force according to its relative position to the destination, can steer the forces of the whole robotic group. We prove that using our algorithm, the followers' forces will converge to the direction of the leader's force. This enables the leader robot to further guide the object to its destination, or along a pre-assigned trajectory. We validate our algorithm in both simulation in a physics engine and hardware experiments with four custom-built robots. Additionally, in our setting the leader can also be a human operator. We show this case in two experiments: (1) the leader robot is tele-operated by a human using a joystick; (2) a human directly manipulates the object together with the robots. Our approach is truly scalable, and efficient by canceling the need for communication, which can be complicated, unreliable, and power hungry, especially in a large network.

Previously, we proposed a first-order dynamical controller [1], where both velocity and acceleration are measured to achieve a consensus on the forces of all robots without communication. Our previous approach is suitable for many small robots to manipulate a comparatively massive object, where inertial and viscous force are dominating. In this paper, we instead propose a kinematic controller based

*This work was supported in part by NSF grant CNS-1330036.

[1]Z. Wang is with the Department of Mechanical Engineering, Boston University, Boston, MA 02215, USA, zjwang@bu.edu

[2]M. Schwager is with the Department of Aeronautics and Astronautics, Stanford University, Stanford, CA 94305, USA, schwager@stanford.edu

only on velocity measurement to deal with the case where kinetic friction is more significant, and the sizes of both the object and the robots are comparable. We provide a new convergence analysis, as well as new custom-built robots, and new hardware experiments to verify our method.

There is a large body of relevant work in the field of multi-robot manipulation. A minimalist approach was presented in [2] to move furniture and the tradeoff between different amount of sensing, communication and computation was discussed. A popular solution for multi-robot manipulation is called caging, where robots move in a formation that keeps the object from escaping outside of the formation [3], [4], [5]. Forces from the robotic group can be coordinated given the desired acceleration and a communication network [6]. Recently, many interesting experiments are conducted using real robots, including massive uniform manipulation with 100 Kilobot robots using one global control input [7], [8], a kinematic manipulation scheme based on decentralized centroid estimation [9], [10], four omni-directional robots that can lift and move a passenger vehicle [11], and three mobile robotic arms that can carry a deformable bed sheet by running local convex optimization [12]. There are also attempts to deal with multi-robot manipulation using no communication, such as moving a tall object towards a destination based on visual occlusion [13], and a similar idea of using inter-robot force sensing as an implicit way for coordination [14]. Passive robots also do not need communication when cooperating with a human to handle an object [15]. None of the work above is decentralized, uses no communication, and guarantees the efficiency of force alignment at the same time, which can be achieved by our approach.

Our work also coincides with some discoveries from biological studies of ants in cooperative food transport behaviors. In [16], it was pointed out that ants adjust their forces according to their local detection of small-scale vibration or deformation of the object instead of directly communicating with other ants. Researchers measured the forces from real ants [17] and it was found out that the forces align better and better as time goes on. The leader-following mechanism is also found in [18], where a small informed portion of ants actively lead and steer the whole group to the nest.

The rest of the paper is organized as follows. Section II formulates the problem. Section III presents our controller design and proof of force convergence. We introduce the design of our physical robots in Section IV. Our approach is verified by both simulation in Section V and hardware experiments in Section VI. Finally, conclusion and future work are in Section VII.

II. Problem Formulation

We consider a manipulation task in a horizontal 2D region $Q \subset \mathbb{R}^2$, where the ground has friction with coefficient μ. The acceleration of gravity is denoted by g. The target object sits on the ground and its mass is M. We assume that the object is too heavy for a single robot to move. We therefore employ a fleet of N robots, $\{R_1, R_2, \cdots, R_N\}$ where R_1 is the leader robot and $\{R_2, \cdots R_N\}$ are the follower robots. Each of these robots is able to grasp the object at some attachment points and apply a force to the object, denoted by $\{F_1, F_2, \cdots, F_N\}$. Once the total force from the robots overcomes the friction, the object can slide on the ground, and we denote the velocity of the object at the center of the mass as vector $\mathbf{v}$, expressed in the world reference frame. Then according to Newton's second law, the dynamics of the object is given by

$$M\dot{\mathbf{v}} = \sum_{i=1}^{N} \mathbf{F}_i - \mu M g \frac{\mathbf{v}}{\|\mathbf{v}\|}, \tag{1}$$

or its discrete approximation by Euler method:

$$M\frac{\mathbf{v}_{t+1} - \mathbf{v}_t}{\Delta t} = \sum_{i=1}^{N} \mathbf{F}_i(t) - \mu M g \frac{\mathbf{v}_t}{\|\mathbf{v}_t\|}. \tag{2}$$

The only force we consider from the environment is kinetic friction, given by the last term on the right in (1) and (2), whose magnitude is constant, and whose direction opposes the velocity of the object. We find that this friction strongly dominates in real-world experiments at the scales of mass and velocity considered in this paper. Unfortunately, it is a nonlinear effect, making traditional analysis difficult.

Each robot R_i can measure the velocity of the object in its local reference frame, denoted as $\mathbf{v}^i$. However, there is no global reference frame and global positioning information, and the follower robots also do not know where the destination is. Only the leader robot knows the destination and its relative position to the destination. Based on this information, the leader robot can then compute the desired velocity $\mathbf{v}_d$ for the object using a path planning algorithm. The objective of this paper can be formulated as follows.

***Objective** 1:* Given all the sensing and actuation capabilities described above, design: (i) a force feedback controller for the follower robots such that the followers' forces $\mathbf{F}_i$ will converge to the direction of the leader's force $\mathbf{F}_1$; (ii) a force feedback controller for the leader robot such that the velocity of the object $\mathbf{v}$ can converge to the desired velocity $\mathbf{v}_d$. No communication is allowed between any two robots.

We also need an additional assumption to accomplish the objective, which is stated below.

***Assumption** 1:* All the robots know the value of μ, M, N, g.

Note that in this paper, we do not control the rotation of the object, which is left for future work.

III. Controller Design

In this section, we will first present our controller design for both the follower and leader robots. Then we will mathematically prove that our controllers guarantee the robots' forces align and the object's velocity converges to the desired velocity.

A. Follower's Controller

The followers assume that each robot in the group contributes equal force to the object so that the magnitude of their forces should be $\mu Mg/N$. As we explained in the intuitive example in the introduction, the direction of the

followers' forces should simply be the moving direction of the object such that the followers can help maintain the object's motion. Bringing together the magnitude and direction, the controller for the follower robots is formally written below as

$$\mathbf{F}_i^i = \frac{\mu M g}{N} \frac{\mathbf{v}^i}{\|\mathbf{v}^i\|},\ i = \{2, 3, \cdots, N\}. \tag{3}$$

Note that the superscript i represents the local reference frame since there is no global reference frame information for the robots.

B. Leader's Controller

The leader robot is responsible for steering the direction the object moves by injecting its destination information. We define the controller for the leader robot as

$$\mathbf{F}_1^1 = f_d \frac{\mathbf{v}_d^1}{\|\mathbf{v}_d^1\|}, \tag{4}$$

where $\mathbf{v}_d^1$ is the desired velocity of the object expressed in the leader's local reference frame and f_d is a scalar determined by

$$f_d = K_p \max\{\|\mathbf{v}_d^1\| - \|\mathbf{v}^1\|, 0\}. \tag{5}$$

Again, $\mathbf{v}^1$ is the object's velocity in the leader's frame. The underlying principle in determining $\mathbf{F}_1$ is to drive the object towards the desired velocity $\mathbf{v}_d^1$. From equation (4) we can see that $\mathbf{F}_1$ has the same direction as $\mathbf{v}_d^1$ while the magnitude is governed by a proportional controller that tends to reduce the difference between the magnitude of $\mathbf{v}_d^1$ and $\mathbf{v}^1$. The $\max$ function is used to ensure that $\mathbf{F}_1$ does not point opposite to $\mathbf{v}_d^1$. Notice that the follower and leader controllers are nonlinear, as are the dynamics of the object.

C. Convergence of Controller

Next we present our main theorem, where we prove that using the controller (3), (4) will align all followers' forces with the leader's force. The object's velocity $\mathbf{v}$ will also converge to the desired velocity $\mathbf{v}_d$. From the controller(3), (4) itself, it can be seen that our approach requires no communication among robots. The proof of the main theorem relies on the lemma below.

Lemma 1: Given a constant vector $\mathbf{w}$, update vector $\mathbf{v}$ using the following discrete formula,

$$\mathbf{v}_{t+1} = \alpha_t \mathbf{w} + \beta_t \mathbf{v}_t,$$

where $\{\alpha_t | \alpha_t \geqslant 0, \alpha_1^2 + \alpha_2^2 + \cdots + \alpha_t^2 \neq 0\}$ is a series of non-negative constants, and $\{\beta_t | 0 < \beta_t < 1\}$ is a series of constants between 0 and 1. Then the direction of $\mathbf{v}_t$ will converge to the direction of $\mathbf{w}$ as $t \to +\infty$, i.e.,

$$\mathbf{v}_t \to \gamma \mathbf{w},$$

where $\gamma > 0$ is a scalar determined by $\{\alpha_t\}$ and $\{\beta_t\}$.

Proof: Starting from $t = 1$, we have

$$\mathbf{v}_2 = \alpha_1 \mathbf{w} + \beta_1 \mathbf{v}_1,$$

$$\mathbf{v}_3 = \alpha_2 \mathbf{w} + \beta_2 \mathbf{v}_2 = (\alpha_2 + \beta_2 \alpha_1)\mathbf{w} + \beta_1 \beta_2 \mathbf{v}_1,$$

$$\mathbf{v}_4 = \alpha_3 \mathbf{w} + \beta_3 \mathbf{v}_3 = (\alpha_3 + \beta_3 \alpha_2 + \beta_3 \beta_2 \alpha_1)\mathbf{w} + \beta_1 \beta_2 \beta_3 \mathbf{v}_1.$$

By induction, we further have

$$\mathbf{v}_t = (\alpha_{t-1} + \beta_{t-1}\alpha_{t-2} + \beta_{t-1}\beta_{t-2}\alpha_{t-3} + \cdots + \beta_{t-1}\beta_{t-2}\cdots\beta_3\beta_2\alpha_1)\mathbf{w} + (\beta_1\beta_2\cdots\beta_{t-2}\beta_{t-1})\mathbf{v}_1.$$

As $t \to +\infty$, we know $\beta_1\beta_2\cdots\beta_{t-2}\beta_{t-1} \to 0$. Therefore,

$$\mathbf{v}_t \to \gamma \mathbf{w},$$

where

$$\gamma = \alpha_{t-1} + \beta_{t-1}\alpha_{t-2} + \beta_{t-1}\beta_{t-2}\alpha_{t-3} + \cdots + \beta_{t-1}\beta_{t-2}\cdots\beta_3\beta_2\alpha_1.$$

Apparently $\gamma > 0$, so $\mathbf{v}_t$ and $\mathbf{w}$ have the same direction, and this completes the proof. ■

Theorem 1: Given the dynamics of the object in (2) and the desired velocity $\mathbf{v}_d$, using the controller (3), (4), with a time step $0 < \Delta t < N\|\mathbf{v}_t\|/\mu g$, will lead to all followers' forces aligning in the same direction as the leader's force and the object's velocity will converge to $\mathbf{v}_d$.

Proof: We will prove the theorem in the discrete time domain, and assume that all controllers and dynamics are updated in synchronized time steps. Rewriting equation (3), (4) in the world frame, we get

$$\mathbf{F}_i(t) = \frac{\mu M g}{N} \frac{\mathbf{v}_t}{\|\mathbf{v}_t\|},\ i = \{2, 3, \cdots, N\}, \tag{6}$$

$$\mathbf{F}_1(t) = K_p \max\{\|\mathbf{v}_d\| - \|\mathbf{v}_t\|, 0\} \frac{\mathbf{v}_d}{\|\mathbf{v}_d\|}. \tag{7}$$

Note that we remove the superscript i on all the $\mathbf{v}$ because we express all the quantities in the world reference frame throughout this proof for the convenience of analysis. This does not affect the correctness of the proof since all the $\mathbf{v}_t^i$ are actually the same vector, although they may be different in different local reference frames. Plug (6), (7) into (2) and we have

$$\begin{aligned}\mathbf{v}_{t+1} &= \mathbf{v}_t + \frac{\Delta t}{M}\left(\mathbf{F}_1(t) + (N-1)\frac{\mu M g}{N}\frac{\mathbf{v}_t}{\|\mathbf{v}_t\|}\right) - \mu g \Delta t \frac{\mathbf{v}_t}{\|\mathbf{v}_t\|} \\ &= \frac{K_p \max\{\|\mathbf{v}_d\| - \|\mathbf{v}_t\|, 0\}\Delta t}{M\|\mathbf{v}_d\|}\mathbf{v}_d + \left(1 - \frac{\mu g \Delta t}{N\|\mathbf{v}_t\|}\right)\mathbf{v}_t\end{aligned} \tag{8}$$

Then we can use the conclusion in Lemma 1 by letting

$$\alpha_t = \frac{K_p \max\{\|\mathbf{v}_d\| - \|\mathbf{v}_t\|, 0\}\Delta t}{M\|\mathbf{v}_d\|} \geqslant 0,$$

$$0 < \beta_t = 1 - \frac{\mu g \Delta t}{N\|\mathbf{v}_t\|} < 1,$$

which are true when $\Delta t < N\|\mathbf{v}_t\|/\mu g$. Therefore we know that the direction of $\mathbf{v}_t$ will converge to that of $\mathbf{v}_d$. The magnitude of $\mathbf{v}_t$ will also converge to $\mathbf{v}_d$ because the magnitude is decoupled from the direction and is governed by a proportional controller. As a result, the followers' forces will align with the leader's according to (6) when $\mathbf{v}_t \to \mathbf{v}_d$. ■

The evolution of the force alignment is also visualized in Figure 2 by showing an example from time t to $t + 2$. We find that the condition on Δt is easily satisfied in practice with an update rate of 100Hz.

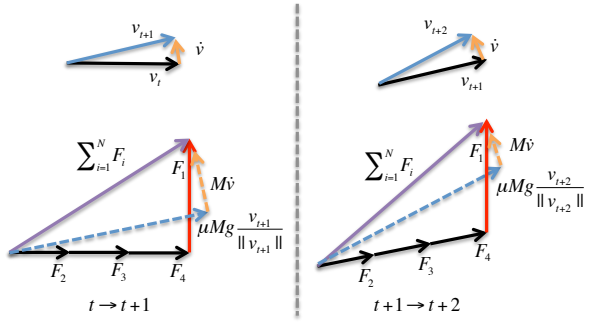


Fig. 2. Geometric explanation of Theorem 1. The evolution from t to $t+2$ with 1 leader and 3 follower robots is shown. It can be seen that if F_1 sticks with its direction all the time, then graudually v and F_i will converge to the direction of F_1.

D. Note on Initial Movement and Stop

Since the controllers (3), (4) are effective only when the object is moving, we need to trigger the initial movement of the stationary object in the first place, and do so without communication. This is done through random trials, where robots apply random forces and there is a chance that the resulting force is larger than the friction [1]. Once the object starts to move, this is sensed by the robots, and the controllers (3), (4) take over. Stopping can be performed by the leader by just applying zero force to the object. According to (1), (3), if $\mathbf{F}_1$ is zero, the sum force from the robots $\sum_{i=1}^{N} F_i$ will be less than the friction, and thus the object will stop.

IV. Robot Design

We build a robot prototype that brings together a series of sensing, computation and actuation capabilities in order to verify our theory experimentally. We insist on keeping our robot as simple and low-cost as possible. Therefore, we build our robot upon a commercially available, affordable robot called the m3pi from Pololu[1], as shown in Figure 3. The m3pi robot uses a differential drive scheme, and is controlled by an mbed-enabled[2] LPC1768 micro-controller running at 96MHz. We integrate more functionality into the m3pi robot, through customized hardware and software components described below.

A. Mechanical Design

The original m3pi robot comes with two layers. The base layer contains batteries, motors and related driving circuits. The second layer hosts the LPC1768 micro-controller. We add an additional third layer to accommodate our force sensors (Figure 5) and customized PCB. Besides the robot, we also build a gripper (Figure 4) that consists of one revolute joint, realized through a bearing and a shaft. The gripper can be mounted to the robot on the tip of the force sensor through screws. The other side of the gripper can be manually attached to the object, also using screws. Mounting holes for holding the optical sensor are available on the object side of the gripper. All the customized mechanical components except screws and nuts are 3D-printed.

[1]www.pololu.com
[2]mbed.org

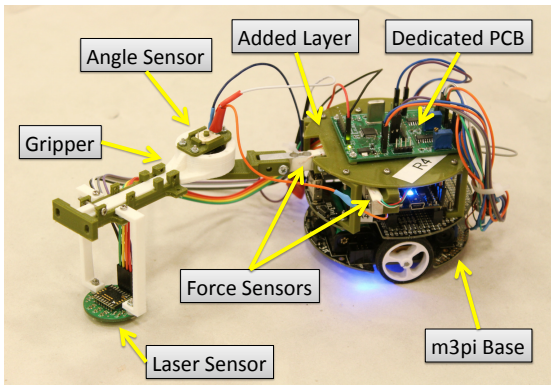


Fig. 3. Our custom-built manipulation robot with key components shown.

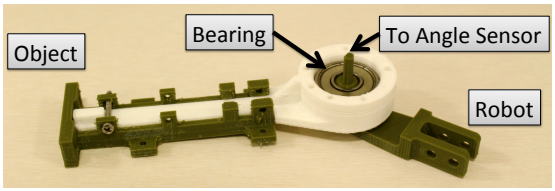


Fig. 4. The one DOF gripper. The long shaft can be connected to the rotary potentiometer to obtain the angle reading.

B. Force Sensing

Each robot is equipped with a force sensing unit (Figure 5), consisting of two load cells that are installed perpendicularly in order to measure the 2D force applied by the robot to the object in the x and y direction. Each load cell, having four strain gauges configured as a Wheatstone bridge, can measure the force in one direction. The differential output voltages are amplified and measured on a dedicated PCB, which has an STM32 micro-controller running at 72MHz. STM32 can then send the A/D data to our main controller LPC1768 via serial port. Before usage, each load cell is carefully calibrated using calibration weight. The capacity of each load cell is -2N to 2N.

C. Velocity Sensing

We measure the velocity of the object using an ADNS9800 optical laser sensor, typically used in the optical mouse. The sensor outputs the delta distance in x and y direction to the LPC1768 via Serial Peripheral Interface (SPI) bus. We then differentiate the distance signal with the time interval to get the average velocity. The laser sensor is installed as close as possible to the object on the gripper in order to get an

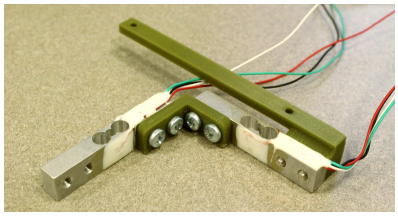


Fig. 5. The force sensing unit, consisting of two perpendicularly connected load cells.

accurate measurement of the object's velocity. Two holders extending from the gripper make sure that the optical sensor touches the ground properly.

D. Angle Sensing and Miscellaneous

Since the robot and the object do not necessarily have the same heading, the angle measurement on the revolute joint of the gripper is needed in order to convert velocity from the object's reference frame to the robot's frame. We place a rotary potentiometer on top of the rovolute joint, and connect it with the shaft via the through-hole on the sensor. The voltage is then sampled by LPC1768 and converted into an angle value.

There is no global localization information for the follower robots. However, we track the position of the leader robot through an OptiTrack[3] system by placing reflective markers on the leader (Figure 1). This matches our assumption that only the leader robot knows the destination. Note that there is an Xbee wireless module on each robot, but the follower robot does not use it for any kind of communication during the experiment. The follower robots only use Xbee to receive a start command before the experiment and upload their sensor measurement histories to a computer for analysis after the experiment.

E. Force Feedback Control

In order to implement the controller (3), (4), robots first need to be able to control how much force they apply along the x and y directions [19]. We can generate the force by controlling the two wheel speeds of our robot. Here we introduce our low-level force feedback controller that maps the force generation to motor control.

Our force feedback controller contains three steps. First of all, since the velocity measured by the laser sensor (attached rigidly to the object) is in the object's frame, we need to convert it into the robot's local frame. This is achieved by

$$\mathbf{v}^i = R(\theta_i)\mathbf{v}, \tag{9}$$

where i specifies the robot, θ_i is the angle of the joint on the gripper and $R(\cdot)$ represents the rotation matrix. Note that we ignore the effect of rotation on the velocity, which is reasonable given that the object does not spin too fast. A more detailed analysis on this issue can be found in our previous paper [20].

The second step is force generation. Briefly speaking, the forces come from the robots' tendencies to go faster or slower than the object. The larger the tendency is, the greater the force will be and vice versa. This tendency can be characterized by the difference between the commanded velocity of the robot and the actual velocity of the object [12]. Thus, we use a linear model to describe this phenomenon, given by

$$\mathbf{v}_c^i - \mathbf{v}^i = K_f(\mathbf{F}_i - \mathbf{f}_i), \tag{10}$$

where $\mathbf{v}_c^i$ represents the commanded velocity of robot i, $\mathbf{F}_i$ is the desired force in (3) and $\mathbf{f}_i$ is the actual force vector applied by robot i measured by force sensors. The parameter K_f is a constant that needs to be tuned experimentally.

The last step is to fulfill the commanded velocity $\mathbf{v}^i$ on the robot. Note that although our differential-driven robot is non-holonomic, an offset point outside the center of the mass of the robot can be holonomic [21]. Since the joint on the gripper is where the force is measured and applied, we choose the pivot of the joint to be our holonomic offset point. Denote the angular velocity of the left and right motor as ω_l and ω_r, the radius of the wheels as r_w, the distance between two wheels as d_w, and the offset distance of the point from the center of the mass as l. Then the velocity of the offset point can be written as

$$\mathbf{v}_p^i = \left[\frac{(\omega_l - \omega_r)r_w l}{d_w}, \frac{(\omega_l + \omega_r)r_w}{2}\right]. \tag{11}$$

Letting $\mathbf{v}_p^i = \mathbf{v}_d^i$, we can solve for the wheel speeds and then output the command to the two motors respectively.

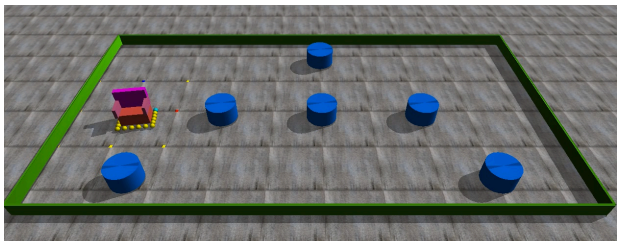


Fig. 6. Simulation setup in ODE. The chair is in red. 20 robots, denoted by spheres, surround the chair at the bottom. The yellow spheres are follower robots while the leader is in light blue, situated at the upper right conner of the chair. The blue cylinders are obstacles that should be avoided by the object. Video link: https://youtu.be/Ldy-e6WUrJs

V. Simulation

A simulation study is conducted to verify our controller (3), (4), as well as Theorem 1. We use an open-source physics engine called Open Dynamics Engine (ODE)[4] in order to ensure the correctness of the rigid-body motion. In the simulation, we deploy 20 robots to manipulate a chair of realistic dimensions, as shown in Figure 6. Each robot is able to apply a force to the chair up to 0.8N in any direction. For simplicity, we do not draw any specific grasping mechanism and abstract the robots as spheres instead. The small chair (0.6m(L)×0.4m(W)×0.8m(H)) weighs 2kg and the friction coefficient is set to be 0.59, therefore the friction is 11.564N (assuming $g = 9.8$ m/s^2). One can see that at least 15 of our robots are needed in order to overcome the friction. The friction coefficient and robot force are the same as in our hardware experiments below. The leader's objective is to guide the object to follow a series of pre-assigned waypoints in the arena. The waypoints are carefully chosen to avoid the obstacles, denoted by blue cylinders. We use a simple strategy to determine the $\mathbf{v}_d$ in (4): let $\mathbf{v}_d$ always point towards the next waypoint with a constant magnitude at 0.3m/s. There may be better waypoint following algorithms, but that is not the focus of this paper. The initial movement

[3]http://www.optitrack.com

[4]http://www.ode.org

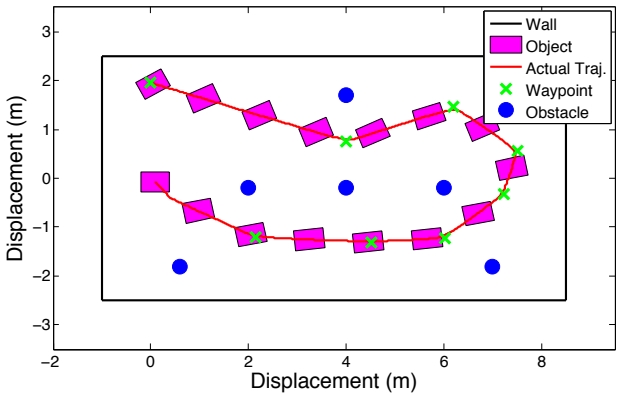


Fig. 7. The trajectory of the object in the simulation. The object is drawn every 4 seconds.

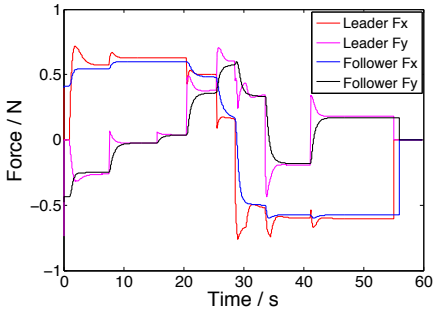


Fig. 8. Forces of the robots during the simulation. The abrupt jumps of the leader's force are due to the switching between old and new waypoints.

of the object is triggered through random trials as described in Section III-D.

Figure 7 shows that the robots successfully follow the waypoints and transport the chair to the destination. Note that the object does spin during the manipulation process, however we do not control the rotation, which is left for future work. Figure 8 shows the forces of both the leader and follower robot during the simulation. Since we assume perfect sensing and actuation, all the follower robots will apply the same force to the object. In the experiments, we will investigate robustness to real-world noise and disturbances. The force convergence, as proven in Theorem 1, is evident in Figure 8, even when the leader's force is changing.

VI. Experiments

The effectiveness of our algorithm is successfully verified through experiments. We build four robots to manipulate a cardboard box, whose weight is 270g. The friction is approximately 1.6N, so μ is equal to 0.59. Each robot is able to apply a maximum force at about 0.8N on its own. Note that the friction coefficient and force limit of the robots are the same as they are in the simulation. The experiment is conducted in a 4m $\times$ 2.5m arena, which is covered by an Optitrack motion capture system. Only the leader and the object are tracked by Optitrack. In all experiments, we skip the initial random force trials as described in Section III-D since this is not the focus of the paper. Instead, we send one kickoff command via XBee assigning robots an initial speed. After that and during the experiments, the follower robots complete all the sensing and computation onboard, without having access to communication and localization information. We conducted three experiments with three different types of leaders (either a robot or human), while the followers are the same for all trials. All the experiments are run multiple times to ensure repeatability. The experiment video can be found online at https://youtu.be/Ldy-e6WUrJs.

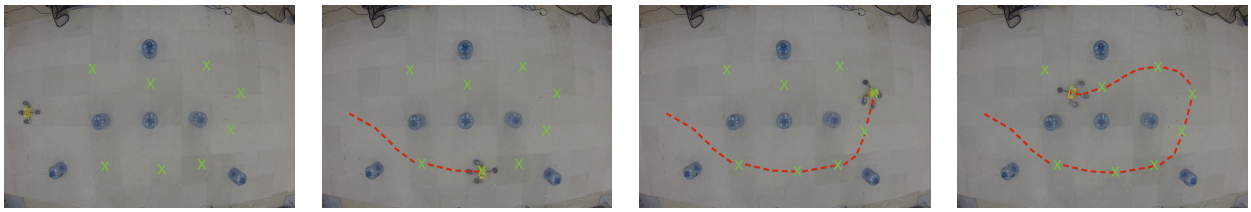
A. Leader is an Autonomous Robot

In this case, the leader robot pulls the front of the object while three follower robots push on the side or back. Note that the configuration of the robots is not unique since our algorithm only cares about the forces rather than the positions of the robots. Three follower robots are also interchangeable because they have the same hardware and run the same program. The leader robot receives commands wirelessly via XBee from a computer, where the controller (4) is implemented. In contrast to the follower robots, which have local sensing and control onboard, the leader robot has external sensing from Optitrack and does not have onboard angle and velocity sensor.

The task setup is the same as the simulation. The experiment demonstrates the effectiveness of the proposed controller (3), (4). The outcome trajectory shows coordinated actions for the groups, such as line tracking, left and right turns, and a U-turn. This indicated that our robotic fleet is able to execute trajectories other than the one demonstrated here. Figure 9 shows the snapshots of the waypoint following process. The force, velocity, and angle measurements of the robots are recorded at 10Hz and then uploaded for analysis after the manipulation is done. Figure 10 plots the force histories of the robots. From Figure 10(a) we can see that the followers' forces are close to 0.4N, which is as desired. In Figure 10(b)–(d), we plot the followers' force records in a rotated local reference frame, where the y axis is the direction of $\mathrm{v}_i(t)$ at every sampling time. Ideally, the follower's force should be located at (0,0.4) according to (3). Due to noise and disturbances in the system, the forces scatter around the y axis. However, the average forces are close to the desired force at (0,0.4) and nearly all the force samples have positive y component, indicating that the follower robots actively help the leader in the process rather than dragging behind.

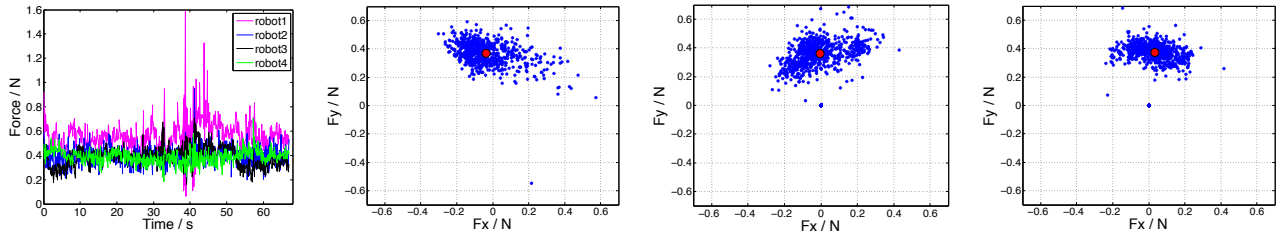
B. Leader is a Robot Tele-operated by Human

We show human-swarm cooperation in the next two manipulation experiments. One advantage of our algorithm is that the follower robots will synchronize to the leader's force, no matter where the force is from. This allows the robotic swarm to interact and cooperate with a human leader. In this experiment, we keep all the settings for the three follower robots, and manually control the leader robot using a remote joystick instead of running the automatic waypoint following program. The joystick, as shown in Figure 11, captures the wheel speeds specified by a human operator in real time and then sends the commands to the leader robot via XBee. For the convenience of comparison, we keep the destination and obstacles the same as before. As shown in the video, the



(a) $t = 0$ (b) $t = 26$s (c) $t = 45$s (d) $t = 62$s

Fig. 9. Snapshots of the waypoint following process. Green "X"s are waypoints and the red dotted line is the actual trajectory. Video link: https://youtu.be/Ldy-e6WUrJs



(a) Force magnitude (b) Robot 2 (c) Robot 3 (d) Robot 4

Fig. 10. Forces measured during *Experiment A*. (a): The magnitude of the forces applied by all the robots. (b)-(d): The force histories applied by robot 2, 3 and 4, respectively in each robot's local frame. The red dot denotes the average force.

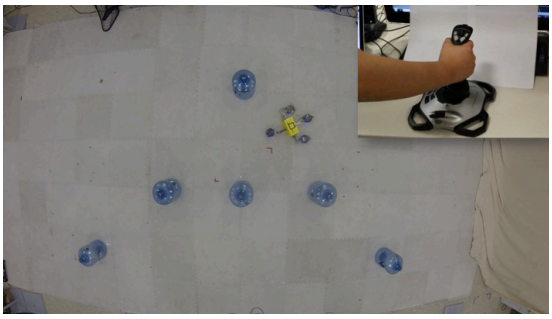


Fig. 11. Leader robot is controlled by a human operator via a remote joystick in *Experiment B*. Video link: https://youtu.be/Ldy-e6WUrJs

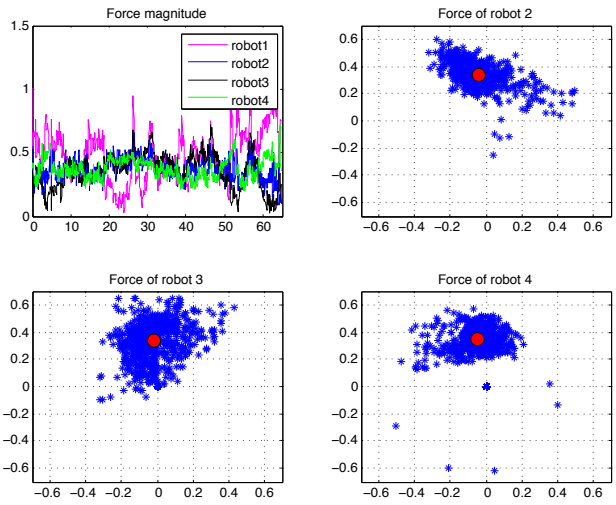


Fig. 12. Force plots of *Experiment B*. The way we plot them is the same as Figure 10.

follower robots can successfully cooperate with the human operator to transport the object. To check the effectiveness of this human-swarm cooperation, we plot the forces of all the robots as before in Figure 12. It can be seen that all the follower robots contribute positive forces most of the time, although there do exist some negative ones compared to Figure 10. The negative forces of the follower robots are caused by the abrupt hard pulling of the leader robot due to the lack of force feedback for the human operator.

C. Leader is a Human

To further interact with the robots more directly, we let a human grasp the object instead of tele-operating a leader robot. As shown in Figure 13, the human leader gently holds the gripper and guides the follower robots. The human leader tries to apply a small force to the object within the same scale as the robots, in order to prevent overwhelming the robots' forces. Although we have not yet developed a device to measure the force from the hand of the human leader, the contribution of the follower robots can be verified through the force plotting again in Figure 14. The forces are more scattered than the previous two experiments, because it is relatively more difficult for the human leader to maintain a constant, small force in this case. However, the followers' mean forces still stay around the ideal $(0, 0.4)$.

VII. Conclusion and Future Work

To conclude, in this paper we have proposed a kinematic controller to coordinate a group of manipulation robots using only onboard velocity sensing and force feedback without communication. The efficiency of this cooperative manipulation is guaranteed by our approach, as proven in our main Theorem 1. Both simulation and hardware exper-

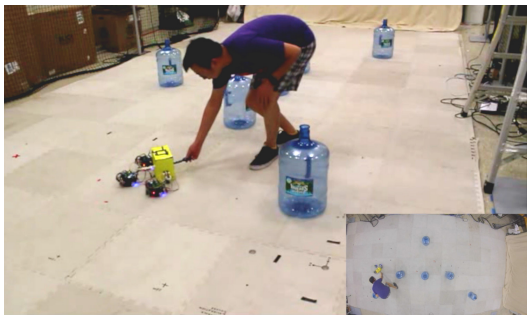


Fig. 13. Three follower robots cooperate with a human leader. Video link: https://youtu.be/Ldy-e6WUrJs

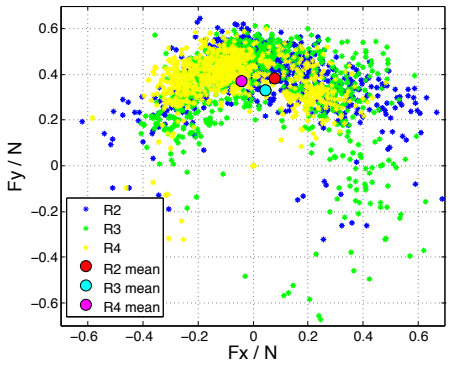


Fig. 14. Forces of three follower robots in *Experiment C*. In the interest of space we have shown them all on the same plot.

iments with four custom-built robots successfully verify the proposed approach. We also show that using our controller, the follower robots can interact with a human leader to move an object together.

We are working to extend the approach in several directions. Adaptive control can be applied to estimate the parameters we require to know beforehand in this work (friction coefficient, object mass, total number of robots), and thus relax Assumption 1. We also plan to study the effect of rotation and how to control the rotation. For the hardware part, we plan to switch to an omni-directional ground robot platform, where 2D force feedback control can be better implemented.

REFERENCES

[1] Z. Wang and M. Schwager, "Multi-robot manipulation without communication," in *International Symposium on Distributed Autonomous Robotic Systems (DARS)*, 2014.

[2] D. Rus, B. Donald, and J. Jennings, "Moving furniture with teams of autonomous robots," in *Proc. of the International Conference on Intelligent Robots and Systems (IROS)*, vol. 1, 1995, pp. 235–242.

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

[4] W. Wan, R. Fukui, M. Shimosaka, T. Sato, and Y. Kuniyoshi, "Cooperative manipulation with least number of robots via robust caging," in *Advanced Intelligent Mechatronics (AIM), 2012 IEEE/ASME International Conference on*, pp. 896–903.

[5] J. Fink, M. A. Hsieh, and V. Kumar, "Multi-robot manipulation via caging in environments with obstacles," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2008, pp. 1471–1476.

[6] J. M. Esposito, "Decentralized cooperative manipulation with a swarm of mobile robots," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, 2009, pp. 5333–5338.

[7] 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)*, 2013, pp. 520–527.

[8] 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 *Proceedings of International Conference on Autonomous Agents and Multi-agent Systems (AAMAS)*, 2013, pp. 47–54.

[9] 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 *Robotics and Automation (ICRA), IEEE International Conference on*, May 2015, pp. 1282–1288.

[10] G. Habibi, Z. Kingston, Z. Wang, M. Schwager, and J. McLurkin, "Pipelined consensus for global state estimation in multi-agent systems," in *Proceedings of International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2015, pp. 1315–1323.

[11] A. Amanatiadis, C. Henschel, B. Birkicht, B. Andel, K. Charalampous, I. Kostavelis, R. May, and A. Gasteratos, "Avert: An autonomous multi-robot system for vehicle extraction and transportation," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, May 2015, pp. 1662–1669.

[12] J. Alonso-Mora, R. Knepper, R. Siegwart, and D. Rus, "Local motion planning for collaborative multi-robot manipulation of deformable objects," in *Robotics and Automation (ICRA), IEEE International Conference on*, 2015, pp. 5495–5502.

[13] J. Chen, M. Gauci, and R. Gross, "A strategy for transporting tall objects with a swarm of miniature mobile robots," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, May 2013, pp. 863–869.

[14] G. Baldassarre, V. Trianni, M. Bonani, F. Mondada, M. Dorigo, and S. Nolfi, "Self-organized coordinated motion in groups of physically connected robots," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 37, no. 1, pp. 224–239, 2007.

[15] Y. Hirata, Y. Ojima, and K. Kosuge, "Variable motion characteristics control of an object by multiple passive mobile robots in cooperation with a human," in *Robotics and Automation (ICRA), IEEE International Conference on*, 2008, pp. 1346–1351.

[16] H. McCreery and M. Breed, "Cooperative transport in ants: a review of proximate mechanisms," *Insectes Sociaux*, vol. 61, no. 2, pp. 99–110, 2014.

[17] 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)*, Zaragoza, Spain, June 2010.

[18] A. Gelblum, I. Pinkoviezky, E. Fonio, A. Ghosh, N. Gov, and O. Feinerman, "Ant groups optimally amplify the effect of transiently informed individuals," *Nature communications*, vol. 6, 2015.

[19] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *Robotics and Automation, IEEE Journal of*, vol. 3, no. 1, pp. 43–53, 1987.

[20] Z. Wang and M. Schwager, "Multi-robot manipulation with no communication using only local measurements," in *Decision and Control (CDC), 2015 IEEE Conference on*, 2015, **To Appear**.

[21] N. Michael and V. Kumar, "Planning and control of ensembles of robots with non-holonomic constraints," *The International Journal of Robotics Research*, vol. 28, no. 8, pp. 962–975, 2009.