Distributed Planar Manipulation in Fluidic Environments

Guillaume Sartoretti[1], Samuel Shaw[2] and M. Ani Hsieh[3]

Abstract— **We present a distributed control mechanism allowing a swarm of non-holonomic autonomous surface vehicles (ASVs) to synchronously arrange around a rectangular floating object in a grasping formation; the swarm is then able to collaboratively transport the object to a desired final position and orientation. We analytically consider the problem of synchronizing the ASVs' arrival at the object, with regard to their initial random positions. We further analytically construct a set of acceptable trajectories, allowing the transport of the grasped object to its final desired position. Numerical simulations illustrate the performance of the presented control mechanism. We present experimental results, to demonstrate the feasibility and relevance of our strategy.**

I. INTRODUCTION

As robotic swarm sizes can vary over time, we are interested in developing scalable control mechanisms, that can adapt to any swarm size, and that are resistant to single robot failures. In some cases, scalability can be obtained via the use of a distributed control framework, where the global swarm's behavior is distributed among all agents. The robustness of the resulting distributed swarm strategy can then be increased by adding redundancy in the number of agents performing each role.

In this paper, we address the problem of transporting an object in a fluidic environment, using a swarm of autonomous non-holonomic robots. We build on [1] to devise a distributed Braitenberg-inspired control mechanism, allowing the autonomous surface vehicles (ASVs) to move along smooth trajectories only as they approach, grasp and manipulate a rectangular object.

Previous works have succesfully achieved decentralized manipulations of large objects by swarms of ground robots. In [2], the decentralized manipulation of an L-shaped object held under caging is obtained by first having the team surround the object of interest and then transporting it by collectively pushing it along the desired direction. Manipulation is achieved through careful composition of artificial potential fields. More recently, a decentralized control mechanism allowing the manipulation of a general-shaped object by a swarm of gripping robots evolving on the ground is presented in [3]. However, when considering a swarm of ASVs, new challenges arises, due to the decreased friction and the non-negligible inertial effects. Closely related to the present work, in [4], the authors devise a decentralized control mechanism allowing a swarm of tugboats to rotate a larger object, by collaboratively using their individual torques. Using tugboats attached to a larger vessel, the authors successfully devise and implement a control mechanism enabling the vessel's manipulation in [5], [6].

The main contribution of this work is the construction of a set of acceptable trajectories, allowing a swarm of ASVs to transport an object from any given initial state (position, given orientation) toward any desired final state, under CC. The presented distributed control mechanism then allows the swarm of ASVs to collaboratively grasp the object under CC, and to collectively transport it toward its final desired state.

The outline of this paper is as follows: in Section II, we briefly presents the considered problem. In Section III, we define the two main phases of our control mechanism, and present numerical results. Section IV is devoted to our experimental results, while Section V discusses of the future perspectives.

II. PROBLEM STATEMENT

We consider a swarm of N autonomous robotic ASVs evolving on a still body of water, initially positioned at random. A large vessel, represented by a rectangular object on the water surface, is initially positioned at a position that we define as the origin $\mathbf{O} = (0, 0)$ of the swarm's global frame of reference. Without the loss of generality, we will assume that the vessel is positioned so that its length lays on the X-axis. The goal for the robotic swarm is to first regroup on one side of the vessel, facing the vessel (namely, on the side where $y > 0$), and then to push the vessel toward a final position $\{x_f, y_f\}$, with a final orientation h_f. We separate the control mechanism into two phases, namely the approach and the push phases.

Each ASV is modeled as a differential autonomous robot, evolving on a two-dimensional plane. Each robot is then assigned a specific target point T_i $(1 \leq i \leq N)$, which it aims for during the approach phase. The target points $T_2, ..., T_{N-1}$ are uniformly distributed along the central axis of the vessel, while T_1 and T_N are placed at each end of the vessel. This allows the swarm, once positioned near their target points, to hold the vessel under conditional closure (CC) (see Fig.1) [7]. A robot starting on the bottom half-plane $\{y < 0\}$ must first move around the vessel. Once on the other half-plane $\{y > 0\}$, the ASV must then gradually change its course to approach the vessel perpendicularly to its side.

We gratefully acknowledge the support of ONR Award Nos. N000141211019 and N000141310731 and National Science Foundation (NSF) grant IIS-1253917. Guillaume Sartoretti is funded by the Swiss National Science Foundation.

[1] Guillaume Sartoretti is with the Laboratory of Microengineering for Manufacturing, Ecole Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland guillaume.sartoretti@epfl.ch

[2] Samuel Shaw is with The Haverford School, PA 19041, USA samushaw@haverford.org

[3] M. Ani Hsieh is with the Scalable Autonomous Systems Laboratory, Mechanical Engineering, Drexel University, Philadelphia, PA 19104, USA mhsieh1@drexel.edu

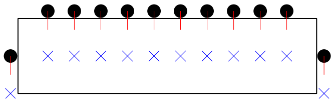


Fig. 1: Example of conditional closure (CC), where the pushing ASVs (black dots) are able to translate and rotate the vessel, with constraints on the translation direction and the rotation pivot. The target points are depicted as blue crosses, and the ASVs' heading vectors as red segments).

Once each ASV in the swarm has reached the vicinity of its target, the vessel is held under CC. In this position, the robots can only translate the vessel towards the half-plane $\{y \geq 0\}$. Furthermore, any rotation applied to the vessel under CC cannot have its pivot point within the vessel. Therefore, the rotation radius (distance between the barycentric position of the vessel, and the pivot point) must be at least $\frac{S_x}{2}$ (i.e., half the length of the vessel). These constraints define a set of allowed translations and rotations, which the swarm can apply to the vessel under CC. The objective then is to devise a distributed control strategy to enable the swarm to maintain CC while manipulating the vessel to the desired position $\{x_f, y_f\}$, with desired orientation h_f, given its initial position/orientation $x_0 = y_0 = h_0 = 0$.

III. PLANAR MANIPULATIONS UNDER CONDITIONAL FORCED CAGING

A. Distributed Approach and Grasp

We first define a general Braitenberg control mechanism [8] allowing an ASV to move at a constant speed, keeping a given point $T = \begin{pmatrix} Tx \\ Ty \end{pmatrix}$ at a given angle ϕ from its heading. The angle ϕ defines the desired difference between the angular position of a robot, and its heading h. For example, when $\phi = \frac{\pi}{2}$, the ASV simply revolves around the point T (i.e., T is the instantaneous center of rotation for the ASV, remaining at an angle of $\frac{\pi}{2}$ from the ASV's heading). Fig. 2 illustrates the situation at a given time and depicts the various relevant quantities.

With these quantities, both motor speeds are updated as follows:

$$\begin{cases} V_{L,i}(t+1) & = S_{1,i} + \beta \cdot \cos\big[h_i(t) - \alpha_i(t) \\ & \quad +(\frac{\pi}{2} - \phi_i(t))\big], \\ V_{R,i}(t+1) & = S_{1,i} - \beta \cdot \cos\big[h_i(t) - \alpha_i(t) \\ & \quad +(\frac{\pi}{2} - \phi_i(t))\big], \end{cases} \tag{1}$$

with $\alpha = \arctan\left(\frac{y_i - Ty_i}{x_i - Tx_i}\right)$, (x_i, y_i) the position of the robot when its motor speeds are updated and $S_{1,i}$ its (constant) approach speed. The control mechanism defined in Eq.(1) allows a robot to rotate around the origin point in a clockwise direction for $\phi = \frac{\pi}{2}$, and in a counter-clockwise direction for $\phi = 3\frac{\pi}{2}$. We want to allow robots starting in the half-plane $\{x > 0\}$ to rotate in the counter-clockwise direction, to reach their destination faster. To achieve this, we let $D \in \{0, 1\}$ denote each robot's direction of rotation (clockwise (0) or counter-clockwise (1)). An ASV starting in the half-plane $\{x < T_i\}$ is assigned $D = 0$, and an ASV starting at initial position $x_{0,i} \geq T_i$ receives $D = 1$. A correction is then made to the control mechanism of Eq.(1) to encompass this rotation direction, by changing $\phi_i(t) \rightarrow \phi_i(t) + D\pi$.

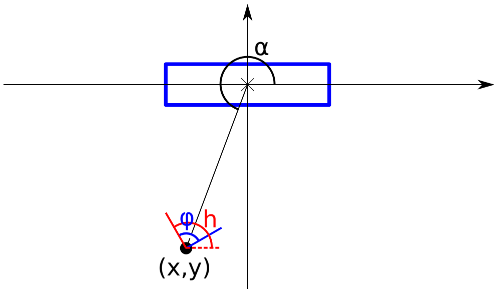


Fig. 2: Illustration of the system and the relevant quantities needed for the Braitenberg control mechanism. Here, $T = \mathbf{O} = (0, 0)$. The vessel (blue rectanlge) and an ASV (black dot) located at position (x, y). The heading h of the ASV is illustrated in red, and its angular position α in black. A desired angle ϕ, defining the desired difference between α and h, is depicted in blue.

To obtain a spiral-shaped motion from the initial position of the robot until its perpendicular approach to the vessel's side, we now let ϕ depend on the angular position of each robot. We want $\phi \approx \frac{\pi}{2}$ when the robot is in the lower half-plane $\{y < 0\}$ (i.e., pure rotation around the origin point). We then want ϕ to gradually decrease toward $\phi = 0$, as α approaches $\frac{\pi}{2}$ (i.e., pure attraction to the origin point). We let ϕ exponentially decrease, as α approaches $\frac{\pi}{2}$:

$$\phi_\sigma(\alpha) = \frac{\pi}{2}\left(1 - \exp\left(-\frac{arcmin(\alpha, \frac{\pi}{2})}{\sigma}\right)\right), \tag{2}$$

where $\sigma^2 \in \mathbb{R}^+$ controls the rate of exponential decay. We write $arcmin(\theta_1, \theta_2)$ as the minimal angle difference between θ_1 and θ_2:

$$arcmin(\theta_1, \theta_2) := \min \begin{pmatrix} mod(\theta_1 - \theta_2, 2\pi) \\ mod(\theta_2 - \theta_1, 2\pi) \end{pmatrix}.$$

We finally incorporate $\phi_\sigma(\alpha)$ into Eq.(1) to obtain the final control mechanism for our robotic swarm. Note how $\phi_\sigma(\alpha)$ must also be corrected with regard to D, to obtain suitable dynamics for both turning directions:

$$\begin{cases} V_{L,i}(t+1) & = S_{1,i} + \beta \cdot \cos\big[h_i(t) - \alpha_i(t) + \frac{\pi}{2} \\ & \quad +D\pi - (2D-1)\phi_\sigma(\alpha_i(t))\big], \\ V_{R,i}(t+1) & = S_{1,i} - \beta \cdot \cos\big[h_i(t) - \alpha_i(t) + \frac{\pi}{2} \\ & \quad +D\pi - (2D-1)\phi_\sigma(\alpha_i(t))\big], \end{cases} \tag{3}$$

1) Computation of the Approaching Speeds: Using the control mechanism from Eq.(3), a robot starting at any given initial position moves along a spiral-shaped trajectory toward its given target point T. We now want all robots to converge to their targets T_i within a desired given time T_{goal}. The correct synchronization of the arrival times allows the robots to start pushing the vessel all at once.

We need to estimate the length of the path a robot will move along during its approach, given its initial position (x_0, y_0). To do this, we approximate the trajectories created by the Braitenberg control mechanism of Eq.(3) as exponential spirals. We express a spiral as a function of the radius of each point of the spiral with respect to its angular position. To simplify the expression, we rotate the frame of reference, and measure the angular position θ of the spiral's points starting from the angle $\frac{\pi}{2}$. In our case, our spirals are defined for any angular position $\theta \in [0;\pi]$, where, because of symmetries, robots cannot move more than half a circle around their target point. Using the relation for $\phi_\sigma(\alpha)$, we now approximate our spiral as:

$$r(\theta) := r_\pi\left(1 - \exp\left(-\frac{\theta}{\sigma}\right)\right). \tag{4}$$

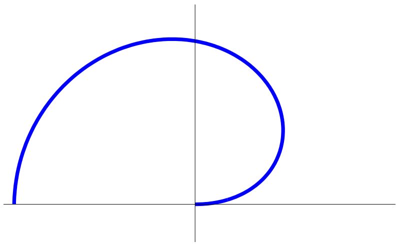


Fig. 3: Typical shape of the spiral of Eq.(4), with $r_\pi = 1$. Here, we make the substitution $\theta = arcmin(\alpha, \frac{\pi}{2})$, meaning that both axes are simply switched.

The typical shape of the spiral, with $r_\pi = 1$, is depicted in Fig. 3. In our case, we need to let each robot compute r_π, from its initial position (r_0, θ_0), where $\theta_0 = arcmin(\alpha_0, \frac{\pi}{2})$. This is achieved by putting r_0 and θ_0 in Eq.(4), and by extracting r_π. We finally find:

$$r_\pi = \frac{r_0}{1 - \exp\left(-\frac{\theta_0}{\sigma}\right)}. \tag{5}$$

With the expression of the spiral for the trajectories' shapes, we can now obtain the trajectories lengths from the initial position of each robot. The initial angular and radial positions of each robot are first computed with respect to their target point:

$$\begin{cases} r_0 = \sqrt{(x_0 - T_x)^2 + (y_0 - T_y)^2} \\ \theta_0 = arcmin(\arctan\frac{y_0 - T_y}{x_0 - T_x}, \frac{\pi}{2}). \end{cases}$$

We then compute the length of the spiral following Eq.(4), from the initial position to the target point, as:

$$L_i(r_0, \theta_0) = \int_0^{\theta_0} \sqrt{r(\theta)^2 + \left(\frac{d}{d\theta}r(\theta)\right)^2}\, d\theta,$$

which yields a lengthy but exact result, given by Eq.(6).

With this analytical expression for the trajectory length of an ASV starting at any initial position, we are able to adapt the individual speeds $S_{1,i}$ of each ASV. We let the approach speed of each ASV depend on its trajectory length $L_i(r_0, \theta_0)$ in order to let the ASVs meet their target point in the vicinity of a desired time T_{goal}:

$$S_{1,i} = S_{1,i}(r_0, \theta_0) = \frac{L_i(r_0, \theta_0)}{T_{\text{goal}}}. \tag{7}$$

2) Link with Brownian Swarms: Eq.(3), defining our Braitenberg control mechanism, can be translated into a multi-agent swarm dynamic, expressed in terms of differential equations. We identify the ASVs as Brownian agents, evolving on the plane $\mathbb{R}^2$, and following the dynamic expressed in polar coordinates:

$$\begin{cases} dr_i(t) = -S_{1,i} \cdot \exp\left(-\frac{arcmin(\alpha_i(t), \frac{\pi}{2})}{\sigma}\right), \\ d\alpha_i(t) = (2D-1)\frac{S_{1,i}}{r_i(t)}\left(1 - \exp\left(-\frac{arcmin(\alpha_i(t), \frac{\pi}{2})}{\sigma}\right)\right). \end{cases} \tag{8}$$

Noise can be added to this dynamic, in order to encompass the actuators and sensors' errors, by adding independent White Gaussian Noise sources on either or both state variables. This effectively allows us to express the swarm's dynamic as a set of N Stochastic Differential Equations (SDE). Conventional tools of dynamical systems theory can be used on the resulting nonlinear SDEs to study the probability distribution of the swarm's position during the approach.

Note that, similarly to the original Braitenberg control dynamics, the norm of the instantaneous speed of each agent remains constant through time:

$$v_i(t) = r(t)\frac{d}{dt}\alpha(t) + \frac{d}{dt}r(t) = S_{1,i} \qquad \forall t.$$

Fig. 4 illustrates the initial position and final position of a swarm of $N = 10$ robots following Eq.(8) without noise, up to time $T = 10$. Notice how the trajectories of the agents are very close to the spiral shapes created from Eq.(4).

B. Computation of the Manipulation Trajectories

Once the ASVs have achieved CC around the vessel, the pushing phase starts. During this second phase, the ASVs continue to follow Eq.(3), but adapt their forward speeds to translate/rotate the vessel. To manipulate the vessel by controlling the individual ASVs' forward speeds, we restrict the vessel's motion to circular arcs only, until it has achieved the desired final position and orientation. By construction, we can show that from its initial position, two well chosen circular arcs always allow the vessel to end up at the desired final position given a desired orientation. We prove this by

$$L_i(r_0,\theta_0) = r_0\left(\frac{1}{e^{\theta_0/\sigma}-1}+1\right)\left(\left(1-\frac{\sigma^2\log\left(\sqrt{\sigma^2+1}+1\right)}{\sqrt{\sigma^2+1}}\right)-e^{-\frac{\theta_0}{\sigma}}\sqrt{\sigma^2\left(e^{\theta_0/\sigma}-1\right)^2+1}+\frac{\sigma^2\log\left(e^{-\frac{\theta_0}{\sigma}}\left(\sigma^2\left(1-e^{\theta_0/\sigma}\right)+\sqrt{\sigma^2+1}\sqrt{\sigma^2\left(e^{\theta_0/\sigma}-1\right)^2+1}+1\right)\right)}{\sqrt{\sigma^2+1}}-\sigma\sinh^{-1}\left(\sigma-\sigma e^{\theta_0/\sigma}\right)\right). \quad (6)$$

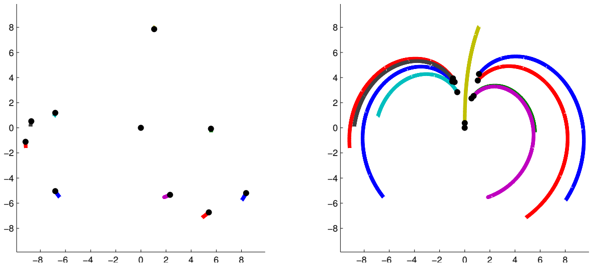


Fig. 4: Initial (left) and final (right) positions, and trajectories, of a swarm of $N = 10$ agents following Eq.(8) without noise. Here, $\sigma^2 = \frac{1}{2}$. Notice the trajectories following the spiral-shape of Eq.(4).

computing the centers of the two circles the vessel will be transported along. We let the vessel first travel an arc length α_1 around the center $\{C_{1,x}, C_{1,y}\}$ of a first circle C_1. It then travels an arc length α_2 around the center $\{C_{2,x}, C_{2,y}$ of a second circle C_2.

We first note that $C_{1,y} = 0$, since the vector $\{C_{1,x} - x_0, C_{1,y} - y_0\}$ must be perpendicular to the initial normal vector of the vessel. The normal vector of the vessel is defined as a unitary vector, perpendicular to the vessel's orientation (i.e., initially the normal vector of the vessel is $\{1, 0\}$). This ensures that the vessel's normal vector is tangent to C_1, since its orientation faces the center of C_1. We then express the constraints within our system:

1) We know that after the first arc α_1, the orientation of the vessel is α_1. After the second arc, the final orientation must therefore verify:

$$h_f = \alpha_1 + \alpha_2. \quad (9)$$

2) Using the rotation matrix $M_\alpha = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin/\alpha) & \cos(\alpha) \end{pmatrix}$, we can compute the successive positions of the vessel after traveling along each of the two arcs. After the first arc, the vessel ends up at the position $\{x_1, y_1\}$, with

$$\begin{aligned} \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} &:= M_{\alpha_1} \cdot \begin{pmatrix} x_0 - C_{1,x} \\ y_0 - C_{1,y} \end{pmatrix} + \begin{pmatrix} C_{1,x} \\ C_{1,y} \end{pmatrix} \\ &= \begin{pmatrix} C_{1,x} \cdot (1 - \cos(\alpha_1)) \\ -C_{1,x} \cdot \sin(\alpha_1) \end{pmatrix}. \end{aligned} \quad (10)$$

From this intermediate position, the vessel then follows an arc length α_2 around the center of C_2. The final position must equal $\{x_f, y_f\}$, which gives us a second relation:

$$M_{\alpha_2} \cdot \begin{pmatrix} x_1 - C_{2,x} \\ y_1 - C_{2,y} \end{pmatrix} + \begin{pmatrix} C_{2,x} \\ C_{2,y} \end{pmatrix} = \begin{pmatrix} x_f \\ y_f \end{pmatrix}. \quad (11)$$

3) Finally, the vector $\{x_1 - C_{2,x}, y_1 - C_{2,y}\}$ must be perpendicular to the vessel's normal vector after the first arc. This condition, similar to the one raising $C_{1,y} = 0$, ensures that at $\{x_1, y_1\}$, the ASV is simultaneously on both circles C_1 and C_2.

$$\begin{pmatrix} x_1 - C_{2,x} \\ y_1 - C_{2,y} \end{pmatrix} \cdot \left(M_{\alpha_1} \cdot \overbrace{\begin{pmatrix} 1 \\ 0 \end{pmatrix}}^{\text{initial normal vector}} \right) = $$

$$(C_{2,x} - C_{1,x}) \cdot \sin(\alpha_1) - C_{2,y} \cdot \cos(\alpha_1) = 0. \quad (12)$$

With these 4 equations, and 5 variables $\{\alpha_1, \alpha_2, C_{1,x}, C_{2,x}, C_{2,y}\}$, we are left with one degree of freedom. We choose to fix α_1, and to compute the values of the other variables accordingly. After solving Eqs(9)-(12), with α_1 fixed, we end up with:

$$\begin{cases} \alpha_2 &= h_f - \alpha_1 \\ C_{1,x} &= -\frac{1}{2}\csc\left(\frac{\alpha_1}{2}\right)\csc\left(\frac{\alpha_1+\alpha_2}{2}\right)\left(x_f\cos\left(\alpha_1 + \frac{\alpha_2}{2}\right) + y_f \sin\left(\alpha_1 + \frac{\alpha_2}{2}\right)\right) \\ C_{2,x} &= \frac{1}{2}\csc\left(\frac{\alpha_2}{2}\right)\csc\left(\frac{\alpha_1+\alpha_2}{2}\right)\left(x_f\cos\left(\frac{\alpha_1}{2}\right) + \sin\left(\frac{\alpha_1}{2}\right)\left(y_f\cos(\alpha_1+\alpha_2) - x_f\sin(\alpha_1+\alpha_2)\right)\right) \\ C_{2,y} &= \cos\left(\frac{\alpha_1}{2}\right)\csc\left(\frac{\alpha_2}{2}\right)\left(x_f\cos\left(\frac{\alpha_1+\alpha_2}{2}\right) + y_f\sin\left(\frac{\alpha_1+\alpha_2}{2}\right)\right) \end{cases} \quad (13)$$

Furthermore, we want to ensure that the initial push is made in the correct direction. Indeed, we could initially find $\alpha_1 > 0$ and $C_{1,x} > 0$. In this case, we would need to initially push the vessel towards $\{y > 0\}$, which is not allowed. To correct this move, we would simply rewrite $\alpha_1 = \alpha_1 - 2\pi$, meaning that the robots will push the vessel in the correct direction, but for a longer arc. Notice that this does not change the final position $\{x_1, y_1\}$ after the first arc. We must therefore check for these cases, and adapt α_1 accordingly:

$$\tilde{\alpha}_1 := \begin{cases} \alpha_1 - 2\pi & \text{if } \operatorname{sign}\alpha_1 = \operatorname{sign}C_{1,x}, \\ \alpha_1 & \text{otherwise.} \end{cases} \quad (14)$$

Similarly, at the interface $\{x_1, y_1\}$ between both circles, we need to ensure that the trajectory is continuous. We also need to correct α_2 if its value would mean the movement must be instantly reverted when switching to the second arc. To obtain the condition, we set the frame of reference at the position $\{x_1, y_1\}$, with the orientation α_1:

$$\begin{pmatrix} \tilde{C}_{2,x} \\ \tilde{C}_{2,y} \end{pmatrix} = M_{-\alpha_1} \cdot \begin{pmatrix} C_{2,x} - x_1 \\ C_{2,y} - y_1 \end{pmatrix}.$$

We then check the same condition as in Eq.(15) (same sign between α_2 and $\tilde{C}_{2,x}$), to correct α_2:

$$\tilde{\alpha}_2 := \begin{cases} \alpha_2 - 2\pi & \text{if } \mathrm{sign}\alpha_2 = \mathrm{sign}\tilde{C}_{2,x}, \\ \alpha_2 & \text{otherwise.} \end{cases} \quad (15)$$

Fig. 5 shows an incorrect trajectory corrected using this method, enabling us to obtain a smooth trajectory following only allowed translations/rotations under CC.

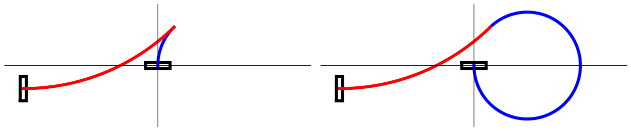


Fig. 5: Example trajectory where α_1 must be corrected (left), and the corrected trajectory using Eq.(15) (right). The initial and final states of the vessel are shown in black.

1) Computation of the Pushing Speeds: Once we find the centers of the circles C_1 and C_2 for an acceptable trajectory, we let the speed of each robot in the swarm depend on its distance to the center of the circle the vessel is traveling around. Namely, if we let $P(t) = \{P_x(t), P_y(t)\}$ be the instantaneous position of the barycenter of the caged vessel at time t, we let the speed of the robot i at time t be:

$$V_i(t) = \frac{||X_i(t) - C||}{||P(t) - C||} \cdot \overline{S_2}, \quad (16)$$

with C the center of the circle the vessel is currently travel along, and $\overline{S_2}$ the desired average speed of the vessel along the trajectory. Furthermore, we let the robots switch from $C = C_1$ to $C = C_2$ when $P(t) \approx \{x_1, y_1\}$ (Eq.(10)).

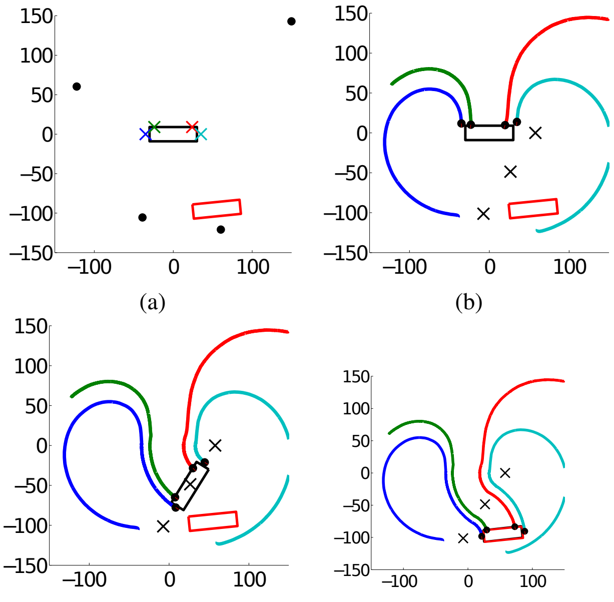
2) Optimal Trajectory in Fluidic Environment: For mobile robots evolving on the ground, a well-known result states that the shortest trajectory under constraint of maximal curvature, and with selected initial and final tangents, is a Dubins path [9], [10]. Such a path always starts with a rotation with maximal curvature (i.e., $\frac{S_x}{2}$ in our case), followed by a forward motion, and finally by a second rotation with maximal curvature. This means that, on the ground at least, our circular trajectories are always sub-optimal (i.e., equal or longer than the corresponding Dubins path).

But since our ASVs evolve in a fluidic environment, a Dubins path can prove to be difficult to apply, due to the necessary rotations with maximal curvature. To apply a rotation with maximal curvature $\frac{S_x}{2}$ to our vessel, we would need to use one of the end ASVs (i.e., the ASVs caging the plank along its main axis) as the rotational pivot. This maneuver is easily managed on the ground, but proves to be harder on water, where the force exerted by the vessel on the pivot during the rotation will most likely move the pivot ASV. In this case, trajectories favoring rotation with smaller curvature can increase the transport precision under CC.

C. Numerical Results

Fig. 6 shows selected consecutive states in a run with $N = 10$ ASVs. In the simulation, we assume that the sum of the ASVs' weights is equal to the individual weight of the vessel. This helps synchronize the pushing, by preventing early arriving ASVs from being able to rotate the vessel too fast on their own. The push phase can therefore only start when the whole swarm is here to push the vessel together. The approach speeds of the ASVs are computed to synchronize their arrival around $T_{\text{goal}} = 30\,s$. The arc length around the first circle is selected to be $\alpha_1 = 1$, and the trajectory is computed in order to position the ASV at $\{x_f, y_f, h_f\} = \{55, -95, 0.1\}$.

To better reflect reality, we add noise to the exact angular position of the agents at each timestep, and add individual noise on their (x, y) position when it is updated in Eq.(3). The noise sources are all White Gaussian Noise sources, with an individual variance of 0.05. The control mechanism of Eq.(3) is updated at $50\,Hz$. Notice that the arrival of the ASVs to the vessel is very well synchronized.



(a)

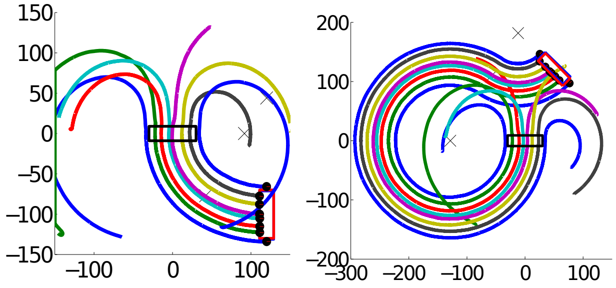
(b)

(c)

(d)

Fig. 6: Selected successive states of the system during a numerical simulation of the approach and push phases. Here, a swarm of $N = 4$ ASVs (black dots) push the large vessel (black rectangle) towards the final position $\{x_f, y_f, h_f\} = \{55, -95, 0.1\}$ (red rectangle).

Fig. 7 show the complete trajectories followed by a swarm of $N = 8$ ASVs to transport the vessel to its final state, from its initial state $\{x_0, y_0, h_0\} = \{0, 0, 0\}$. The considered trajectories allow the plank to be transported to any position, even on the half-plane $\{y > 0\}$, as illustrated in Fig. 7b.



(a) (b)

Fig. 7: Complete trajectories of a swarm of $N = 8$ ASVs (back dots), moving a vessel to its final position (red rectangle). The vessel is initially positioned at $\{x_0, y_0, h_0\} = \{0, 0, 0\}$. The two circles' centers, and the intermediate switching point, are depicted as black crosses.

IV. RESULTS

A. Experimental Setup

In this section we evaluate the proposed control mechanism employing our indoor laboratory experimental testbed : multirobot Coherent Structure Testbed (mCoSTe). The mCoSTe is an indoor laboratory experimental testbed that consists of a $3m \times 3m \times 1m$ flow tank and a fleet of micro-autonomous surface vehicles (mASVs). The mASVs are differential drive surface vehicles, approximately 12 cm long, equipped with a micro-controller board, XBee radio module, and an inertial measurement unit (IMU). Localization for the mASVs and the vessel being manipulated is provided by an external motion capture system.

The considered control mechanism was successfully implemented on a swarm of $N = 4$ mASVs, evolving on still water. The boats are tracked in real time by a collection of overhead cameras, and their positions, headings, speeds and heading derivatives are read at a frequency of $20\,Hz$. The role of the vessel is played by a floating plank of $40\,cm \times 15\,cm$, with fins at both ends of its length, limiting its movement along its main axis.

Since the boat propellers' rotation speeds cannot be linearly translated into a forward speed (as opposed to wheeled mobile robots), Eq.(2) is used to control the boats' headings during the approach phase. The boats' headings and forward speeds are updated at a frequency of $10\,Hz$, using the gathered tracking data. A couple of controllers acting on the boats' headings (PD controller) and forward speeds (P controller) are also updated at $10\,Hz$ to ensure a proper application of the underlying dynamic.

During the approach, the optimal approach speed of each boat is updated at each iteration ($dt = 0.1\,s$), to synchronize the arrival of all the boats. The targets are re-attributed at each iteration, to ensure a minimal probability of collision between the boats. The targets are attributed to the nearest boats to minimize path crossings between boats. The push phase only starts when all the boats are arranged in CC around the plank. This stop phase is needed, since a single early arriving boat can move the light plank easily, which would have a harmful effect on the approach of the other boats.

Once the CC is obtained, the desired final state of the plank $\{x_f, y_f, h_f\}$ is computed in the current frame of reference of the plank. The resulting trajectory is computed to transport the plank toward the desired position under CC. During the push phase, the 2 pushing boats control the plank's orientation along the computed trajectory, ensuring the plank faces the current circle center. The end boats, caging the plank along its x axis, remain in the plank's vicinity to correct its possible deviations along its main axis.

Once in range of the final position, all boats stop their motors. The plank continues its movement, driven by its inertia. Since the plank is held under CC, a stopping maneuver cannot be made by the boats to ensure the plank is stopped precisely on target.

B. Experimental Results

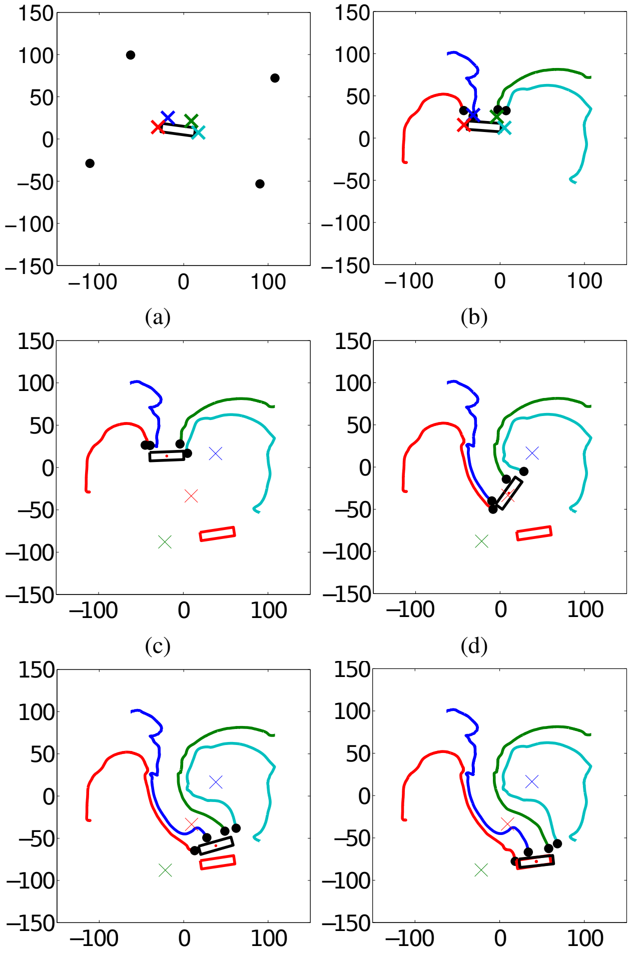
Fig. 8 presents successive states in an experimental run with $N = 4$ boats. During this experimental run, the following values are set for the parameters of the control mechanism:

$$\begin{cases} \sigma = 1 & T_{\text{goal}} = 80\,s \\ \alpha_1 = 1 & \{x_f, y_f, h_f\} = \{55, -95, 0.1\}. \end{cases}$$

The initial position of the boats (black dots) and the plank (black rectangle) is depicted in Fig. 8a, along with the targets along the plank's side (colored crosses). In Fig. 8b, the boats have just arrived to their targets, and have stopped to synchronize the beginning of the push phase. At this moment, the final position and orientation of the plank is computed from its current state (red rectangle), and the corresponding trajectory is computed. The two circle centers, along with the intermediate switching point, are computed (black crosses) (Fig. 8c). The boats then start moving the plank under CC along the computed trajectory, and switch once the plank has just passed the intermediate point (Fig. 8d). As soon as the final position is in range ($20\,cm$) of the plank, the boats stop their motors, and let the plank derive to its final position (Fig. 8e). The plank, still moving under its inertia, derives toward its final position (Fig. 8f).

Fig. 9 presents still frames from the video recording the run depicted in Fig. 8.

A series of experimental runs was performed to test the repeatability of the considered control mechanism. The minimal distance from the plank's center to the desired final position $\{x_f, y_f\}$ was computed for each run, along with the angle error at that precise instant. The corresponding measured errors are illustrated in Fig. 10, along with their respective mean over the 12 experiments. The mean distance error and its standard deviation read $4.88\,cm \pm 1.8\,cm$, while the mean orientation error and its standard deviation read $-0.26\,rad \pm 0.44\,rad$.



(a) (b)

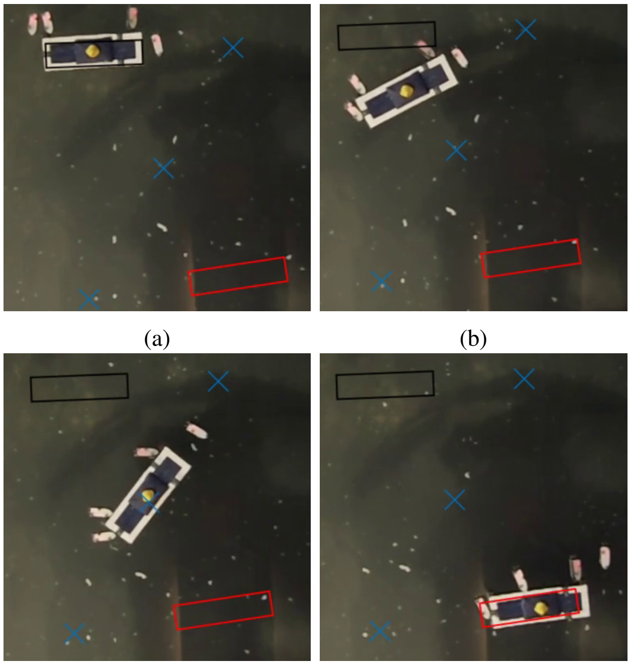
(c) (d)

(e) (f)

Fig. 8: Key successive states during an experimental run of the presented control mechanism with $N = 4$ boats. The trajectories (colored lines) of the boats (black dots), are depicted, along with the plank's current position (black rectangle) and its final desired state (red rectangle). During the approach phase, the boats' targets along the plank's side are plotted as colored crosses. During the push phases, the circle centers and intermediate switching points are illustrated as black crosses.

V. CONCLUSIONS AND PERSPECTIVES

The control mechanism presented in this paper is relevant to solve the problem of manipulating an object in a fluidic environment. It allows simple non-holonomic ASVs to distributively arrange and grasp an object under CC. Despite the fact that this grasping method reduces the set of acceptable translations and rotations, the object can always be transported to any position on the Euclidean plane. Numerical and experimental results are very good, as we test the manipulation of a rectangular object, representing a large vessel for instance.



(a) (b)

(c) (d)

Fig. 9: Key successive states during an experimental run of the presented control mechanism with $N = 4$ boats. Initial position of the boat (black rectangle); final desired state (red rectangle). The circles centers, and the intermediate switching point, are depicted as blue crosses. Caging position obtained by the boats after the approach phase (9a). Start of the pushing phase (9b), switch between both circles (9c), and stopping point of the boats (9d).

The computed trajectories followed by the object toward its final position are known to be sub-optimal, compared to the corresponding Dubins path, but appear to be more stable on the surface of a viscous fluid. Future works will include the comparison between object manipulation using both trajectories in terms of time consumption, stability during manipulation, and precision of final placement.

REFERENCES

[1] G. Sartoretti, M.-O. Hongler, M. Elias de Oliveira, and F. Mondada, "Decentralized self-selection of swarm trajectories: From dynamical system theory to robotic implementation," *Swarm Intelligence*, vol. vol. 8, no. no. 4, pp. 329–351, 2014.

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

[3] G. Habibi, Z. Kingston, 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), 2015 IEEE International Conference on*, 2015, pp. 1282–1288.

[4] M. Feemster, J. Esposito, and J. Nicholson, "Manipulation of large object by swarms of autonomous marine vehicles, part i: Rotational motions," in *System Theory, 2006. SSST '06. Proceeding of the Thirty-Eighth Southeastern Symposium on*, March 2006, pp. 205–209.

[5] J. Esposito, "Distributed grasp synthesis for swarm manipulation with applications to autonomous tugboats," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, May 2008, pp. 1489–1494.

[6] J. Esposito, M. Feemster, and E. Smith, "Cooperative manipulation on the water using a swarm of autonomous tugboats," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, May 2008, pp. 1501–1506.

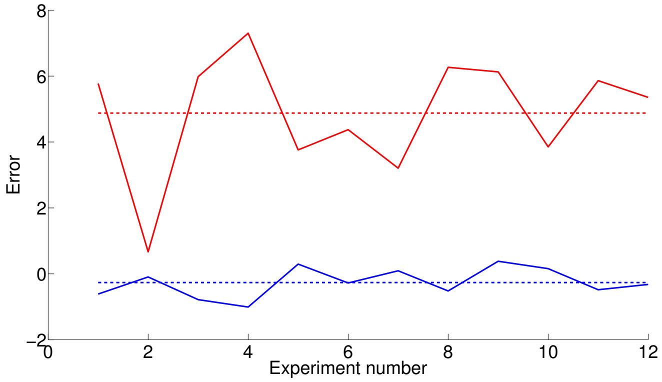


Fig. 10: Minimal error on the position (red) and corresponding error on the orientation (blue) during the 12 experimental trials. The respective mean errors are depicted as dotted lines.

[7] G. A. S. Pereira, V. Kumar, and M. F. M. Campos, “Decentralized algorithms for multirobot manipulation via caging,” *International Journal of Robotics Research*, vol. 23, pp. 783–795, 2002.

[8] V. Braitenberg, *Vehicles: Experiments in Synthetic Psychology*. Cambridge, MA: MIT Press, 1984.

[9] L. E. Dubins, “On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents,” *American Journal of Mathematics*, vol. 79, no. 3, pp. pp. 497–516, 1957.

[10] X.-N. Bui, J.-D. Boissonnat, P. Soueres, and J.-P. Laumond, “Shortest path synthesis for dubins non-holonomic robot,” in *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, May 1994, pp. 2–7 vol.1.