2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 978-1-7281-6212-6/20/$31.00 ©2020 IEEE | DOI: 10.1109/IROS45743.2020.9341780

The Application of a Flexible Leader-Follower Control Algorithm to Different Mobile Autonomous Robots

Aleksander S. Simonsen[1] and Else-Line M. Ruud[1]

***Abstract*— In a wide range of applications involving multiple mobile autonomous systems, maneuvering the robots, vehicles or vessels in some sort of formation is a vital component for the overall task performance. Maintaining a specific distance between the platforms or even a relative geometry may greatly enhance sensor performance, provide collision safety, ensure stable vehicle-to-vehicle communication, and is of critical importance when the systems are in some way physically connected. In this paper, we present a flexible leader-follower type formation control algorithm for autonomous robots which is very simple, generic, yet decent in performance. The method applies to any relative geometry between a leader and one or more followers. In addition to testing the algorithm in simulations for a wide range of scenarios, we have performed experiments involving several different autonomous systems, including small Unmanned Aerial Vehicles (UAVs), Autonomous Underwater Vehicles (AUVs) and Unmanned Surface Vehicles (USVs). This includes pairs of USVs physically interconnected by a tow.**

I. INTRODUCTION

Mobile robots in a variety of different domains, including self driving cars, drones, unmanned surface and submerged marine vessels, are promising technologies that can greatly enhance productivity and efficiency in a wide range of industries and applications. In many cases, the desire to use robots instead of humans or manned systems stems from operations in hazardous environments, of long time scales, or tasks too expensive to continuously supervise. In many of these applications, the objectives are more efficiently executed with teams consisting of several robots working together. It is often also the case that the efficiency in solving the task at hand also increases significantly when the robots maintain some form of relative positioning, a formation. In the maritime industry, it is common for multiple manned surface vessels to cooperate in towing large objects such as larger ships, fish farms and oil rigs. In the future, many of these tasks could be performed by teams of Unmanned Surface Vehicles (USVs). In other applications, the task of the mobile robot might be to follow external objects that are not an integral part of the autonomous system. For example, surveillance vessels that patrol a perimeter of interest while maintaining the capability to autonomously follow any suspicious contacts might be useful for border patrols and customs agents. In all these applications, the mobile systems need some form of formation control algorithm.

A variety of different techniques within robotic formation control have been studied, see for instance the survey on multi-agent formation control [16]. In general, three main approaches have been proposed to tackle the robot formation control problem, *behavioral*, *virtual structure*, and *leader-follower* based methods [9]. In behavioral schemes, multiple robot behaviors (e.g., exploration, maintaining formation, and collision avoidance) are simultaneously competing to commandeer the agent, and the resulting movement is achieved by weighting the individual contributions. Behavioral schemes have been studied extensively, where examples include formations of Unmanned Ground Vehicles controlled by *motor schemas* [2], and complex formation maneuvers with groups of small mobile robots [13]. In *virtual structures*, the robot formation behaves like a rigid body and the movement of the collective is assigned to the lattice-like structure rather than to individual agents. Some work on this topic includes [14] where virtual structure is introduced in order to achieve high precision formation control. *Leader-follower* schemes are characterized by different roles among the participating agents: *leaders* maneuvering with no regard for the rest of the formation and *followers* maintaining a desired distance and orientation to the leader systems. Several studies have been performed on leader-follower methods, e.g., formations of nonholonomic robots [9], Autonomous

Fig. 1. Pictures a) and b) show the Unmanned Surface Vehicle (USV) "SeaKit" and a "Hugin" Autonomous Underwater Vehicle (AUV), respectively. The USV was used to follow the AUV from the surface. The lower-left panel, c), shows the leader Unmanned Aerial Vehicle (UAV) from the perspective of the follower in a test involving small quadcopter drones. Finally, picture d) shows two USVs, "Odin" and "Frigg" maintaining a side-by-side formation while towing a mine sweep.

[1] Norwegian Defence Research Establishment (FFI), Kjeller, Norway aleksander.simonsen@ffi.no

Underwater Vehicles (AUVs) [10], and surface vessels [5].

In this paper, we demonstrate a leader-follower formation controller that can be applied to a wide range of autonomous robotic systems in different domains. No knowledge about the leader's internal state and no cooperation from the part of the leader is assumed, thus making the following agent capable of maintaining formation also in the case where the leader is not capable of assisting in maintaining formation, like for a manual system, or when the leader is simply not willing to cooperate. Our scheme combines elements from line-of-sight (LOS) guidance applied in the case of underactuated marine vessels [3], pure pursuit guidance and leader-follower formation control strategies. As opposed to many formation control approaches, we have assumed no prior knowledge about the dynamic model of the mobile robot. We do however assume that a working low-level control system exists which can translate velocity set-points to the appropriate actuator outputs but that the inner workings of this system is unknown to the formation controller itself. Not modeling the relationship between the robot actuators and the expected physical response directly in the formation control loop will in most cases have a detrimental effect on the performance. However, the algorithm will be more generic and integrable with different (and possibly proprietary) autopilots and control systems for a variety of robots in different domains.

The formation controller has been demonstrated on several real-world systems (see Fig. 1). First, it was adopted in the K-MATE autonomy controller used on the USV system "SEA-KIT" in order to follow a submerged "Hugin" AUV from the surface enabling direct communication and surface-assisted positioning for the AUV while conducting a deep seafloor survey [19]. Secondly, it has been applied in the control autonomy of a test system in which pairs of USVs collectively are used to tow an interconnecting mine sweep. In addition, initial experiments have been done in which the same formation control method is applied to teams of small Unmanned Aerial Vehicles (UAVs) where simple collision avoidance is also included in the controller.

The paper is organized as follows: Section II introduces our system architecture and explains the formation control algorithm. Section III describes our applications as well as the main findings from both simulation experiments and real-world testing before we conclude and summarize in Section IV.

II. THEORY AND METHOD

The formation control algorithm in a leader-follower scheme centers around the control policy of the *follower*, which is assumed to be an autonomous agent in control of a mobile robot. The *leader* is expected to maneuver independently of the follower and not to make any effort towards maintaining the formation. The leader may unexpectedly change speed and/or direction at any time. We assume that the autonomous control architecture is able to estimate the position vector of the follower, $\boldsymbol{r}_f(t)$, and of the leader, $\boldsymbol{r}_l(t)$, at the time t. Another prerequisite for our approach is

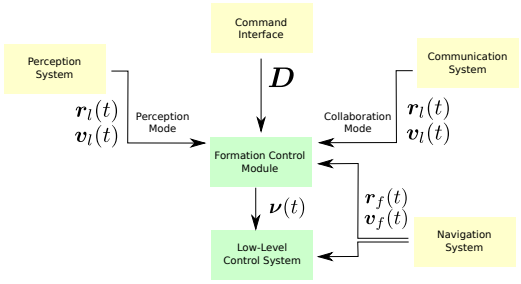

Fig. 2. Illustration of the control architecture employed. Green blocks indicate control components whereas the yellow blocks represent sensor and communication modules. The quantities $\boldsymbol{r}(t)$, $\boldsymbol{v}(t)$ and $\boldsymbol{\nu}(t)$ represent position, velocity, and velocity reference, respectively, and leader and follower variables are marked with subscripts of l and$f$, respectively. The formation geometry is defined by $\boldsymbol{D}$.

direct velocity measurements. Optionally, if the positions can be obtained at a high enough rate, the respective velocities defined by $\boldsymbol{v}_f(t) = \frac{\partial \boldsymbol{r}_f(t)}{\partial t}$ and $\boldsymbol{v}_l(t) = \frac{\partial \boldsymbol{r}_l(t)}{\partial t}$ may be used instead. Moreover, the autonomous agent is assumed to have access to a low-level control system that maps desired velocity set-points, $\boldsymbol{\nu}(t)$ to the actuator outputs. Figure 2 shows the various components connected to the formation control module. Here, the navigation system provides the variables associated with the follower, $\boldsymbol{r}_f(t)$ and $\boldsymbol{v}_f(t)$, to both the control system and the formation (guidance) controller. The external state of the leader, $\boldsymbol{r}_l(t)$ and $\boldsymbol{v}_l(t)$, can either be communicated between the robots through deliberate cooperation, or the follower can estimate it by itself using a perception system.

The goal of the follower is to asymptotically reach a desired position relative to the leader and to maintain that state. This location, which is always defined relative to the leader, is called the *equilibrium* and is denoted with the symbol $\mathcal{E}$. The displacement between the position of the leader and $\mathcal{E}$ is described by the vector $\boldsymbol{D}$, which we define by a distance D from the leader at an angle Ψ around the leaders $z-$axis and an angle Φ around the $y-$axis in the reference frame of the leaders velocity vector,

$$\boldsymbol{D} = DR(\hat{\boldsymbol{v}}_l^y(t), \Phi)R(\hat{\boldsymbol{v}}_l^z(t), \Psi)\hat{\boldsymbol{v}}_l(t), \tag{1}$$

where the notation $\hat{\boldsymbol{a}} = \boldsymbol{a}/a$ is used to indicate the unit vector of $\boldsymbol{a}$, with $a = \sqrt{\sum_i a_i^2}$ being the length. The angle-axis rotation matrix is given by

$$R(\boldsymbol{u}, \theta) = \cos(\theta)\boldsymbol{I} + \sin(\theta)[\boldsymbol{u}]_\times + (1 - \cos(\theta))[\boldsymbol{u} \otimes \boldsymbol{u}], \tag{2}$$

where $\boldsymbol{I}$ is the 3×3 identity matrix, $[\boldsymbol{a}]_\times$ denotes the cross product matrix of $\boldsymbol{a}$, and $\boldsymbol{a} \otimes \boldsymbol{b}$ is the outer product between the vectors $\boldsymbol{a}$ and $\boldsymbol{b}$. Figure 3 illustrates the main concept in a two dimensional case. Here, the formation is defined by the follow distance D and a desired rotation, Ψ, in the horizontal plane (bearing) from the leader to the follower. It is helpful to introduce the following vectors (see Fig. 3) in the formation geometry

$$\boldsymbol{d}(t) = \boldsymbol{r}_l(t) - \boldsymbol{r}_f(t), \tag{3}$$

$$\boldsymbol{l}(t) = \boldsymbol{d}(t) + \boldsymbol{D}. \tag{4}$$

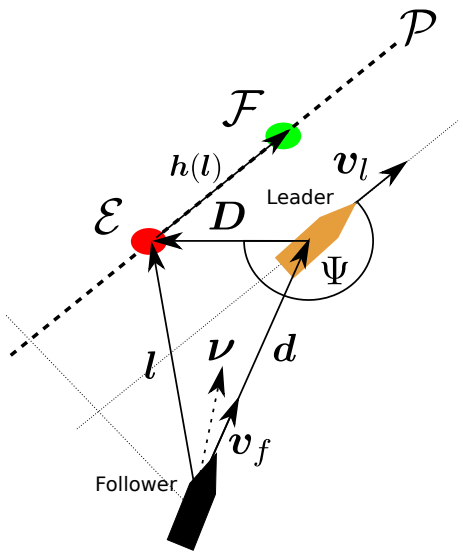

Fig. 3. Illustration of the formation control algorithm applied to the case of two Unmanned Surface Vehicles (USVs). The *leader* is illustrated in orange whereas the *follower* is depicted in black. The red circle indicates the *Equilibrium*, which is the reference position for the follower relative to the leader. The green circle defines the *Follow point*, which correspond to the steering direction of the follower.

The control objectives are formulated by the following asymptotic conditions,

$$\lim_{t\to\infty} d(t) = D, \tag{5}$$

$$\lim_{t\to\infty} \hat{\boldsymbol{d}}(t) \cdot R(\hat{\boldsymbol{v}}_l^y(t), \Phi) R(\hat{\boldsymbol{v}}_l^z(t), \Psi) \cdot \hat{\boldsymbol{v}}_l(t) = -1, \tag{6}$$

$$\lim_{t\to\infty} v_f - v_l = 0, \tag{7}$$

$$\lim_{t\to\infty} \hat{\boldsymbol{v}}_f(t) \cdot \hat{\boldsymbol{v}}_l(t) = 1. \tag{8}$$

Equations (5) and (6) demand that the relative displacement between the follower and the leader satisfies the desired formation geometry defined by $\mathbf{D}$. The alignment of the velocity vectors of the robots are specified in (7) and (8).

In our approach, the controller is divided into two parts, a direction controller for generating $\hat{\boldsymbol{\nu}}(t)$ and a speed controller that calculates $\nu(t)$,

$$\boldsymbol{\nu}(t) = \nu(t)\hat{\boldsymbol{\nu}}(t). \tag{9}$$

Our direction controller is inspired by a LOS scheme for straight line path following as described in [4]. The principle behind LOS guidance is to steer towards a point $\mathcal{F}$ which is displaced by a look-ahead distance h along some path $\mathcal{P}$. In our approach, the follower estimates a virtual straight path parallel to the leaders velocity vector through $\mathcal{E}$ (see Fig. 3). Unlike many other approaches, our look-ahead distance depends on $\boldsymbol{l}$, i.e., $h \equiv h(\boldsymbol{l})$, and increases rapidly as the follower reaches $\mathcal{E}$, i.e., following the path $\mathcal{P}$ is only performed in the later stages of the formation assembly. The look-ahead distance is calculated as follows:

$$h(t) = k_0 + \begin{cases} -\boldsymbol{l}(t) \cdot \hat{\boldsymbol{v}}_l(t) + \frac{1}{1+l(t)} k_l, & \text{if } \boldsymbol{l}(t) \cdot \hat{\boldsymbol{v}}_l(t) \leq 0 \\ \frac{1}{1+l(t)} k_l & \text{otherwise,} \end{cases} \tag{10}$$

where the parameters k_0 and k_l are used to parameterize the sensitivity of the direction controller. The former term in (10), $-\boldsymbol{l}(t) \cdot \hat{\boldsymbol{v}}_l(t)$, is included to ensure that $\mathcal{F}$ always lies in front of the follower with respect to the leader, also in the case when the follower itself is in front of $\mathcal{E}$, i.e., $\boldsymbol{l}(t) \cdot \hat{\boldsymbol{v}}_l(t) < 0$. Our direction controller is then evaluated as follows:

$$\hat{\boldsymbol{\nu}}(t) := \hat{\boldsymbol{f}}(t), \tag{11}$$

where the direction $\hat{\boldsymbol{f}}(t)$ is simply the unit vector pointing towards $\mathcal{F}$, i.e.,

$$\boldsymbol{f}(t) = \boldsymbol{l}(t) + h(t)\hat{\boldsymbol{v}}_l(t). \tag{12}$$

For the speed controller, we apply a saturation function around the distance from $\mathcal{E}$ along the path $\mathcal{P}$, i.e., $\boldsymbol{l}(t) \cdot \hat{\boldsymbol{v}}_l(t)$, in order to regulate the speed reference. This is similar to the work of [7], which studies a surge controller using a $\tan^{-1}$ function on the along-track error that is globally asymptotically stable in the case of formations of underactuated marine vessels. The speed regulator is defined as follows,

$$\nu(t) = \max\left\{0, v_l(t)\left[1 + \frac{2k_p}{\pi}\tan^{-1}\left(\frac{\boldsymbol{l}(t)\cdot\hat{\boldsymbol{v}}_l(t)}{k_s}\right)\right]\right\}, \tag{13}$$

where k_p and k_s are fixed parameters used to tune the responsiveness.

A few remarks are made regarding the various parameters in the formation controller. The parameter k_p is used to increase the importance of the error-correction term with respect to the feed-forward term in (13). Increasing k_s can be used to dampen the responsiveness of the speed controller, which usually needs some tuning for systems of different acceleration capabilities. Generally, larger k_p and smaller k_s yield more aggressive regulation of speed. For the direction controller, setting $k_l = 0$ reduces (10) to ordinary LOS guidance in which the follower uses a fixed look-ahead distance. Additionally, setting $k_0 = 0$ makes $h(t) = 0$ thus making $\mathcal{F} = \mathcal{E}$, which results in a highly responsive, less stable, pure-pursuit guidance controller. Generally, larger k_0 and k_l yield a more sluggish, although more stable, direction controller.

III. RESULTS AND DISCUSSION

In this paper, we will use simulations with one leader and one follower USV to explore the workings and performance of the controller. We then demonstrate the application of the controller on a real-life scenario in which two USVs maintain a specific relative geometry while towing a mine sweep. In addition, results from UAV-UAV experiments are presented in order to justify that the controller is applicable to a wide range of different robots.

A. Simulation

We have performed several simulations in which we investigate the impact of the various controller gains. In addition, updating the formation goals during formation-keeping is performed in order to explore the robustness of the controller.

Our simulator uses the same dynamic model as described in [3] for USVs.

Figure 4 illustrates how the various gains can be altered to induce different behavior in the following dynamics. The leader vessel maintains a constant speed and course (about 30 degrees) and the follower track is plotted for 4 different controller configurations. In general, for a well tuned system, we characterize three distinct phases, which we call the *approach*, *assembly* and *formation* phases (indicated above the plots in Fig. 4). A small k_0 is desired in order to achieve a pure-pursuit-like approach. As has been noted earlier, the nature of pure pursuit guidance tends to guide the follower into a tail-chase scenario [18] which is preferable here due to the leaders inherent lack of concern for the follower. However, as evident from Fig. 4, pure-pursuit-like configurations tend to yield unstable formations, especially for robots with a significant time delay between controller output and response. A large look-ahead distance $k_0 >> 0$ on the other hand is very good for stability, but causes the follower to approach much closer to the leader at an angle as well as increasing the assembly time. If k_0 is kept small, and a non-zero k_l is selected (orange track in Fig. 4), the controller produces a fast tail-chase (pure-pursuit) trajectory that ends in a stable (LOS) formation. This is explained by the gradual transition from a small look-ahead distance to a large one during the assembly.

Figure 5 shows some of the results from a simulated test with stepwise changing reference values. The controller gains were chosen as $k_0 = 50$, $k_l = 20$, $k_p = 1$ and $k_s = 20$. The leader follows a path consisting of four straight lines, along which different changes in follower reference values were performed. We tested different reference changes along each leg; on the first leg, Ψ was kept fixed and D was altered, on the second, Ψ was varied and D kept fixed, while both Ψ and D were altered simultaneously in the third leg. From Fig. 5 we can see that the formation controller quickly adjusts the follower's speed in order to achieve the requested formations, as is clear from the speed curves, ensuring a fast transition between formation reference changes. The reference changes result in smooth transitions without oscillations on both bearing and distance to the leader, which is as expected considering the relatively large look-ahead gains (k_0 and k_l). Once formation is reached, it is stable until the next change in reference. At times, we can see a small offset between the desired and actual distance between the vessels after formation convergence, which might have been reduced with a more aggressive direction controller or through the use of integral effects. In this scenario, however, it is more important to keep a stable formation.

Because collision avoidance is not actively imposed by the controller, a risk of collision might arise in certain cases, such as in sharp turns or with reckless bearing reference changes. One such situation can be seen in the top right part of the position plot in Fig. 5, where the following vessel cuts the corner and speeds up to be able to keep up with the leader. We believe that this problem is solvable by running a collision avoidance component in parallel with the following

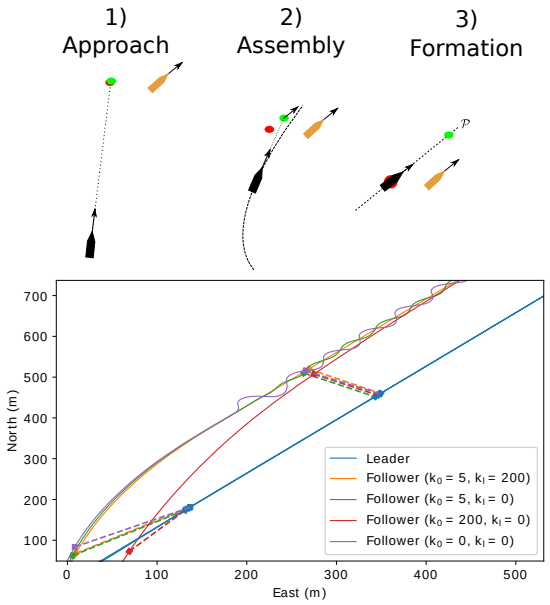

Fig. 4. Position track for the leader (blue) and the follower for 4 different controller configurations for k_0 and k_l. The leader vessel maintains a constant speed and course (about 30 degrees). These configurations include a pure-pursuit like (purple) and two fixed look-ahead controllers depicted in red and green. A hybrid configuration with non-zero k_0 and k_l is depicted in orange. The lines connecting the different trajectories indicate the temporal alignment between the leader and the follower. Above the plot, from left to right, the *approach*, *assembly* and *formation* stages are illustrated. The red and green circles in the illustration depict the *Equilibrium* and *Follow-point*, respectively.

algorithm, and using some method to combine or prioritize outputs, such as *motor schema* [1] or *subsumption* [6]. These methods allow for easy reuse of control modules and do not necessitate expanding the following algorithm itself. An even simpler solution, although not always applicable, could be to plan the leader path using wide turns only.

B. Experiments

The results presented here are from experiments involving USV-USV teaming and UAV-UAV teaming.

1) USV - USV Teaming: Our main use case for USV-USV teaming is in testing future mine counter measures concepts involving mine sweeping using cooperating USVs [15]. In this scenario, the goal is to pull an interconnecting closed-loop mine sweep between two USVs. We use our highly maneuverable, waterjet propelled USV test platforms for these tests. Because the USVs are physically connected by the mine sweep cables during operation, it is necessary that the USVs keep a more or less fixed formation at all times (see bottom right picture in Fig. 1).

In these experiments, the USVs follow a straight line with a fixed distance between the vessels, keeping formation on a line perpendicular to the course vector. We performed two sets of tests: without and with cables attached. The data from the first tests, where we tested a few different distance references, are shown in Fig 6. The formation parameters were the same as in the simulator tests, that is, $k_0 = 50$, $k_l = 20$, $k_p = 1$ and $k_s = 20$. The bearing reference was

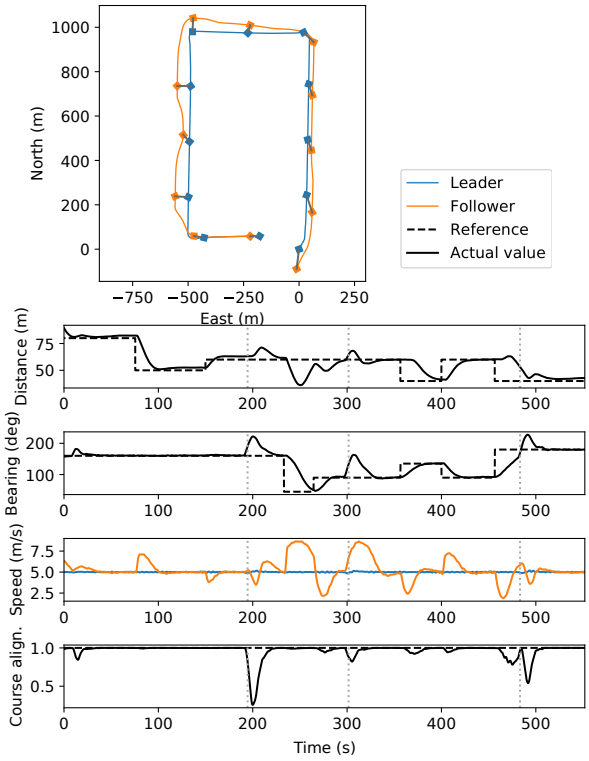

Fig. 5. Data from simulation of Unmanned Surface Vehicles (USVs) in formation. The plots show the position of the USVs, the distance and bearing references (D and Ψ), as well as the speeds and course alignment of the USVs, from top to bottom. The gray lines connecting the paths in the upper panel illustrate the time synchronization of the data where the connected squares in each track correspond to the same instances in time. The transition between the four legs in the leaders path is visualized in the four bottom plots with vertical, dotted lines.

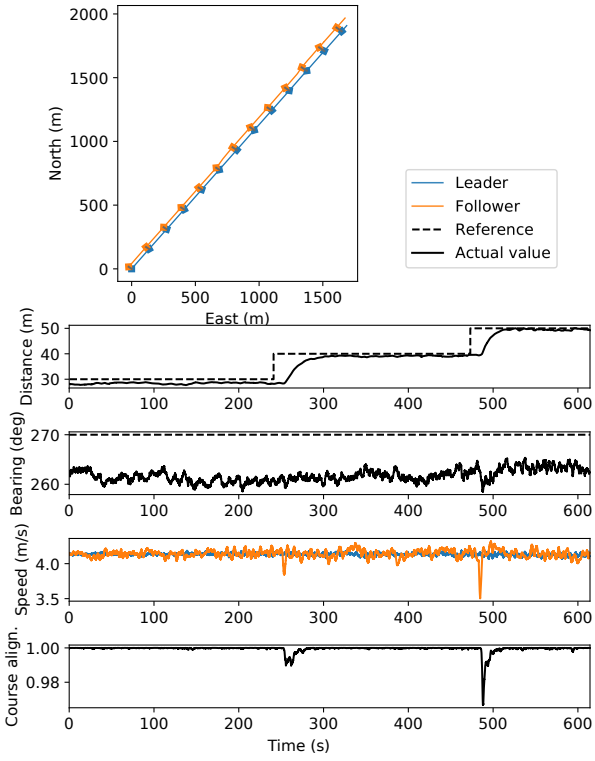

Fig. 6. Results from real-world experiments with two Unmanned Surface Vehicles (USVs) in formation. The plots show the position of the USVs, the distance and bearing references (D and Ψ), as well as the speeds and course alignment of the USVs, from top to bottom.

kept at $\Psi = 270°$ for the whole run, keeping the follower directly to the port side of the leader. Figure 6 shows the same tendencies as predicted by the simulations; all reference changes in D result in smooth and steady transitions. The follower speed is steady for the whole run, except for slight changes on changing D, as expected. This, along with the steady course alignment and bearing from the leader, shows that the algorithm works well for keeping formation along straight lines, which is the typical operation format when towing a mine sweep.

The sea was very calm during the tests, and we were thus not able to test the robustness of the controller in higher sea states. However, because the control system and the formation controller are decoupled, we would expect the performance to be more limited by the robustness of the former than the latter.

2) UAV - UAV Teaming: Lastly, we demonstrate the same algorithm on small rotor-wing UAVs, which have been modified to support third-party integration with custom autonomy controllers and inter-UAV communication [11], [12]. These experiments serve two purposes, 1) they demonstrate the algorithm in the case of a completely different robot domain and 2), the UAVs are used as a test case for adding active collision avoidance to the controller. In this regard, we have applied Artificial Potential Fields (APFs), a simple and reactive collision avoidance layer that is well studied in a variety of forms within the robotic community, examples include [8], [17]. The modification in our case consists in adding an avoidance component, $\boldsymbol{\nu}_{oa}$, to the velocity reference, where

$$\boldsymbol{\nu}_{oa} = -\frac{C}{d^2}\hat{\boldsymbol{d}}. \tag{14}$$

Here, the follower acts as if being pushed away by a repulsive charge at the position of the leader, and C is a constant used to parameterize the repulsion strength.

Figure 7 shows the positions of the leader and follower UAVs in two separate experiments. The lead UAV is tasked to follow a northbound path at a fixed speed ($v_l = 5$ m/s). The follower starts off by following the leader directly to the right ($90°$). When a stable formation is reached, a reckless bearing reference update to the opposite side of the leader ($300°$) is given. With the collision avoidance modification applied (left case), the artificial potential field, which are given a strength $C = 200$, prevents the follower from intercepting too close to the leader, whereas for the original method (right panel), the controller produces a high-risk maneuver that can be seen in the sharp dip in distance which peaks only $5m$ from the leader. In the case without active collision avoidance, the UAVs were artificially separated in altitude,

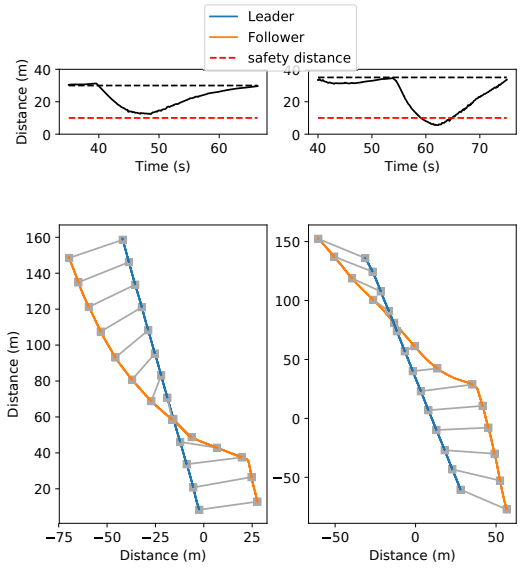

Fig. 7. Results from applying the formation controller to Unmanned-Aerial-Vehicles (UAVs). The position track of the leader and follower aircraft are depicted in blue and orange, respectively. The gray lines connecting the paths illustrate the time synchronization of the data where the connected squares in each track correspond to the same instances in time. The left and right panels depict the case with and without active collision avoidance during a reckless formation update.

i.e., the controller was fed an altitude offset in its inputs to prevent collision.

IV. CONCLUSIONS

We have presented a leader-follower formation control algorithm that has been tested in multiple applications involving real-world mobile robots. The controller is inspired by work addressing formation control for marine applications [4], [7] but has here been developed further and generalized to three dimensions. Using only information about the leader position and its time-derivative (velocity) the algorithm works for both cooperative and non-cooperative formations. The controller performs well in multiple maritime applications where it has been tested on two real world USVs, including scenarios in which USVs keep a side-by-side formation while pulling a tow together. To demonstrate the universality of our approach, the same algorithm has been applied to small UAV systems. In order to increase the robustness with respect to safety, as some weaknesses have been identified in sharp turns and for some mid-formation configuration changes, we have performed initial experiments where simple collision avoidance has been included. Future work includes more testing with extensions to the method, such as more sophisticated collision avoidance, better handling of sharp turns, as well as testing the controller in a richer set of configurations and conditions. Also, the scheme is applicable to scenarios involving more than two robots in different configurations, for instance N robots following one leader or other scenarios involving multiple leaders and followers.

REFERENCES

[1] R. C. Arkin. Motor schema based navigation for a mobile robot: an approach to programming by behavior. pages 264–271, 1987.

[2] T. Balch and R. C. Arkin. Behavior-based formation control for multirobot teams. *IEEE Transactions on Robotics and Automation*, 14(6):926–939, 1998.

[3] E. Børhaug, A. Pavlov, E. Panteley, and K. Y. Pettersen. Straight line path following for formations of underactuated marine surface vessels. *IEEE Transactions on Control Systems Technology*, 19(3):493–506, May 2011.

[4] M. Breivik and T. I. Fossen. Path following for marine surface vessels. *Ocean '04 - MTS/IEEE Techno-Ocean '04: Bridges across the Oceans - Conference Proceedings*, 4(7491):2282–2289, 2004.

[5] M. Breivik, V. E. Hovstein, and T. I. Fossen. Ship formation control: A guided leader-follower approach. *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 17(1 PART 1):16008–16014, 2008.

[6] R. Brooks. A robust layered control system for a mobile robot. *IEEE Journal on Robotics and Automation*, 2(1):14–23, 1986.

[7] M. Burger, A. Pavlov, E. Børhaug, , and K. Y. Pettersen. Straight line path following for formations of underactuated surface vessels under influence of constant ocean currents. *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 7(PART 1):521–526, 2009.

[8] S. J. Chung, A. A. Paranjape, P. Dames, S. Shen, and V. Kumar. A Survey on Aerial Swarm Robotics. *IEEE Transactions on Robotics*, 34(4):837–855, 2018.

[9] L. Consolini, F. Morbidi, D. Prattichizzo, and M. Tosques. Leader-follower formation control of nonholonomic mobile robots with input constraints. *Automatica*, 44(5):1343–1349, 2008.

[10] R. Cui, S. S. Ge, B. Voon Ee How, and Y. Sang Choo. Leader-follower formation control of underactuated autonomous underwater vehicles. *Ocean Engineering*, 37(17-18):1491–1502, 2010.

[11] S. Engebraten, K. Glette, and O. Yakimenko. Field-Testing of High-Level Decentralized Controllers for a Multi-Function Drone Swarm. *IEEE International Conference on Control and Automation, ICCA*, 2018-June:379–386, 2018.

[12] S. Engebraten, K. Glette, and O. Yakimenko. Networking-Enabling Enhancement for a Swarm of COTS Drones. *IEEE International Conference on Control and Automation, ICCA*, 2018-June:562–569, 2018.

[13] J. R. T. Lawton, R. W. Beard, and B. J. Young. A decentralized approach to formation maneuvers. *IEEE Transactions on Robotics and Automation*, 19(6):933–941, 2003.

[14] M. A. Lewis and K. H. Tan. High Precision Formation Control of Mobile Robots Using Virtual Structures. *Autonomous Robots*, 4(4):387–403, 1997.

[15] Ø. Midtgaard and M. Nakjem. Unmanned systems for stand-off underwater mine hunting. In *Proceedings of the Undersea Defence Technology Conference (UDT) Europe 2016*, 2016.

[16] K. K. Oh, M. C. Park, and H. S. Ahn. A survey of multi-agent formation control. *Automatica*, 53:424–440, 2015.

[17] M. T. Wolf and J. W. Burdick. Artificial potential functions for highway driving with collision avoidance. In *2008 IEEE International Conference on Robotics and Automation*, pages 3731–3736, May 2008.

[18] T. Yamasaki and S. N. Balakrishnan. Sliding mode based pure pursuit guidance for UAV rendezvous and chase with a cooperative aircraft. *Proceedings of the 2010 American Control Conference, ACC 2010*, pages 5544–5549, 2010.

[19] Y. Zarayskaya, C. Wallace, R. Wigley, K. Zwolak, E. Bazhenova, A. Bohan, M. Elsaied, J. Roperez, M. Sumiyoshi, S. Sattiabaruth, et al. Gebco-nf alumni team technology solution for shell ocean discovery xprize final round. In *OCEANS 2019-Marseille*, pages 1–10. IEEE, 2019.