Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Leader-Following Formation Navigation for Multiple Robots with Collision Avoidance ⋆

Kazunori Sakurama * **Kazushi Nakano** *

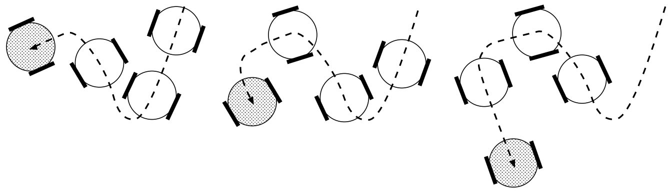
* *The University of Electro-Communications, 1-5-1 Chofugaoka, Chofu-shi, Tokyo 182-8585 Japan (e-mail: sakurama@ee.uec.ac.jp,nakano@ee.uec.ac.jp)*

Abstract: This paper deals with a collision avoidance problem in leader-following formation navigation for multiple mobile robots. Because followers should move along a leader's trajectory, we first try to avoid collisions by adjusting their velocities on the trajectory. This strategy causes delays of the followers from the leader, which is often problematic because we cannot predict how late the followers will be from the leader. Moreover, there are situations that the robots cannot avoid collisions only with this strategy. If the leader goes straight and turns back toward the followers suddenly, the followers have to move back along the trajectory where the leader just moved. In this case, the followers would be better off moving around to the back of the leader and recover their delays even if they go away from the specified trajectory. From these viewpoints, this paper proposes a collision avoidance method for leader-following formation navigation taking into account both the tracking errors and the delays of the followers from the leader. This method adjusts the velocity of each follower, as well as modifies the shape of the trajectory if a delay becomes too large. The effectiveness of the proposed method is demonstrated by a simulation with three mobile robots.

1. INTRODUCTION

In recent years, a formation control has been studied with great interest for practical application, e. g. formation flight of aircrafts (Chen et al. [2003], Semsar and Khorasani [2006]) and formation navigation of mobile robots and marine vehicles (Ikeda et al. [2006], Borhaug et al. [2006]), and has been reviewed in some papers (Scharf et al. [2003], Chen and Wang [2003]). Although decentralization of functions in robots is important, a leader-follower formation, where one leader commands followers, is still important when the leader plays a specific role, and has been adopted in many papers (Chen et al. [2003], Mariottini et al. [2005]). For example, in order to carry relief supplies by multiple robots in a disaster area, one human operates only one robot (leader) and guides other robots (followers) as shown in Fig. 1, where the shaded robot is the leader and the others are the followers. The robots move as shown in the these figures from (1) to (3), where the followers move along the leader's trajectory. We call this formation a *leader-following formation navigation* (LFFN), which is one of the most important issues in formation control. Due to operation of a human, the leader passes along a safe route to avoid obstacles in a disaster area, and moves in a reasonable motion, e. g. slows down for curves. Thus, the followers should pass along the leader's trajectory in the same motion after some delays.

In spite of the importance and applicability of the LFFN, it is not recognized well that the LFFN causes a serious problem which does not happen in previous formation navigation. Because the leader's trajectory varies the shape of



(1) (2) (3)

Fig. 1. Leader-following formation navigation

the formation pattern continuously at every moment as shown in Fig. 1, even after completing the formation, the robots might collide with each other due to quick turns and slowdowns of the leader as shown in Fig. 2 (1). Then, the followers have to avoid collisions even if they destroy the formation. However, it is difficult to realize collision avoidance with keeping formation as much as possible. It should be noted that most papers for formation navigation deal with fixed formation patterns, e. g. line and diamond as shown in Fig. 2 (2). In these cases after completing formation, robots are located apart from each other, and never collide with each other. Although Shao et al. [2005] and Li and Chen [2005] have discussed how to change a formation pattern into another one in finite fixed patterns, the formation pattern does not change continuously in their papers.

In order to achieve collision avoidance for the LFFN, we naturally come up with the following two strategies for each follower:

(a) Change the shape of its trajectory.
(b) Alter its delay time from the leader with velocity adjustments.

⋆ This work was supported in part by Saneyoshi Scholarship Foundation.

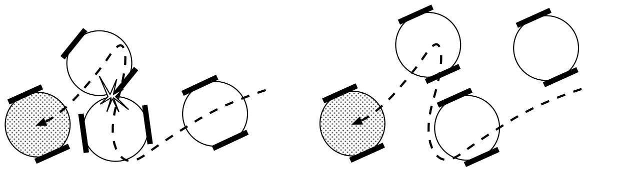


Fig. 2. LFFN and navigation with a fixed formation pattern

The authors have proposed an online method which can be applied to this problem from the viewpoint of (a) (Sakurama and Nakano [2006, 2007]). This method modifies the shape of the leader's trajectory for each robot using a mapping. Although this method realizes collision avoidance, we prefer (b) to (a) for the LFFN because the followers ought to be on the leader's trajectory to move around the safe areas where the leader passed. Although Akella and Hutchinson [2002] have proposed a method for collision avoidance from the viewpoint of (b) by scheduling start times of robots, it is offline, that is the leader's trajectory has to be given in advance with the whole time of usage for scheduling.

This paper deals with a collision avoidance problem in the LFFN. Because followers should move along the leader's trajectory, we first try to avoid collisions based on strategy (b) by adjusting their velocities on the trajectory. This strategy causes delays of the followers from the leader, which is often problematic because we cannot predict how late the followers will be. Moreover, there are situations that the robots cannot avoid collisions only by strategy (b). If the leader goes straight and suddenly turns back toward the followers, they then have to move back along the trajectory where the leader just moved. In this case, the followers would be better off moving around to the back of the leader and recover their delays even if they go away from the leader's trajectory, which is realized by modifying their trajectories based on strategy (a). From these viewpoints, this paper proposes a collision avoidance method for the LFFN based on both strategies (a) and (b), taking into account both the tracking errors and the delays of the followers from the leader. This method adjusts the velocity for each follower based on (b), as well as modifies the shape of the trajectory based on (a) when the delay becomes too large. The effectiveness of the proposed method is demonstrated by a simulation with three mobile robots.

2. PROBLEM SETTING AND STRATEGIES FOR COLLISION AVOIDANCE

Consider the situation shown in Fig. 1, where a leader guides followers along the leader's trajectory after some delays. Because the leader is operated by a human, any future information about the trajectory is unavailable. Let m be the dimension of the space, which is not necessarily 2. There are n robots in the space, whose shapes are circles with radii r_i. We call the leader Robot 1 and the followers Robot $2, 3, \cdots, n$ in order from the leader. Let $\phi : \mathbb{R} \to \mathbb{R}^m$ be a function describing the shape of the leader's trajectory. The leader passes the trajectory ϕ with a trajectory parameter $\sigma(t)$ which is strictly monotonically increasing. In short, the leader moves as $q_1(t) = \phi(\sigma(t))$,

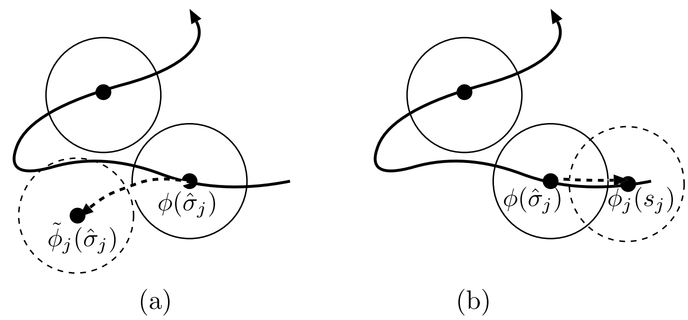


Fig. 3. Collision avoidance methods (a) and (b)

where q_i represents the coordinate position of Robot i. Set the delay time of Robot i as $\tau_i (< \tau_{i+1})$, where the leader's delay is $\tau_1 = 0$. Then, the follower moves as $q_i(t) = \phi(\sigma(t-\tau_i))$. Note that because of the lack of future information about the trajectory ϕ and parameter σ, only $\sigma(t),\ t \in [0, t_c]$ and $\phi(\sigma),\ \sigma \in [\sigma(0), \sigma(t_c)]$ are available for a current time t_c. Assume that the leader moves smoothly enough for ϕ and σ to be of class $\mathcal{C}^p$ for a natural number p. Note that there are infinite combinations of ϕ and σ to describe the trajectory $q_i(t)$ as $\phi(\sigma(t-\tau_i))$. For example, we can assign the trajectory parameter as the time $\sigma(t) = t$ or the arc-length parameter $\sigma(t) = \int_0^t \|\dot{q}_i(s)\| \mathrm{d}s$.

Now, we are naturally afraid that the robots collide with each other when they track the trajectories $\phi(\sigma(t-\tau_i))$. In order for the robots to avoid collisions the trajectories have to satisfy the L inequalities

$$\|\phi(\sigma(t-\tau_i)) - \phi(\sigma(t-\tau_j))\| > r_{ij},\ (i,j) \in \mathcal{L},\ \forall t, \quad (1)$$

where $r_{ij} = r_i + r_j$, $\mathcal{L} = \{(i,j) : i < j,\ i,j \in \mathcal{N}\}$ representing all the pairs of the robots, $\mathcal{N} := \{1, 2, \cdots, n\}$, $L = {}_n\mathrm{C}_2 (= |\mathcal{L}|)$, and $\|\cdot\|$ is the Euclidean norm in the space $\mathbb{R}^m$. Since whether (1) holds or not depends on ϕ and σ, the leader has to move carefully enough for the followers to satisfy (1). However, they are often roughly designed by the operator of the leader, e. g. quick turns and slowdowns. In order to reduce burdens on the operator and to keep safety for unexpected accidents in a disaster area, we need to take care of the case that (1) does not hold. As mentioned in Section 1, there are two strategies (a) and (b) to satisfy (1) for collision avoidance. With above notations, (a) and (b) are described as follows for Robot i:

(a) Modify the shape of the trajectory, that is change ϕ into a new trajectory $\tilde{\phi}_i$.
(b) Modify the trajectory parameter, that is change $\sigma(t-\tau_i)$ into a new parameter $s_i(t)$.

Fig. 3 illustrates these methods, where $\hat{\sigma}_i = \sigma(t-\tau_i)$. With (a), Robot j avoids collision by changing its trajectory. On the other hand, with (b), it does by going back along the leader's trajectory.

A method based on (a) has been proposed by the authors using a $\mathcal{C}^p$ mapping $\Phi_i : \mathbb{R}^{n\times m} \to \mathbb{R}^m$ (Sakurama and Nakano [2007]). The new trajectories $\tilde{\phi}_i$ modified by this method as

$$\tilde{\phi}_i(t) = \Phi_i(\phi(\sigma(t-\tau_1)), \phi(\sigma(t-\tau_2)), \cdots, \phi(\sigma(t-\tau_n)))$$

always satisfy the inequalities for collision avoidance

$$\|\tilde{\phi}_i(t) - \tilde{\phi}_j(t)\| > r_{ij},\ (i,j) \in \mathcal{L},\ \forall t. \quad (2)$$

However, as noted in Section 1, (b) is better than (a) for the objective of the LFFN. In order to avoid collisions based on (b), the new parameters $s_i(t)$ have to be designed to satisfy

$$\|\phi(s_i(t)) - \phi(s_j(t))\| > r_{ij},\ (i,j) \in \mathcal{L},\ \forall t. \tag{3}$$

To realize (3) is often problematic because we cannot predict how late the followers will be and large delays hinder formation navigation. Thus, the delays of the new parameters should be bounded. For this purpose, we demand the inequalities

$$|s_i(t) - \sigma(t - \tau_i)| \le \delta_i,\ \forall i \in \mathcal{N},\ \forall t, \tag{4}$$

with some positive constants δ_i. Assume that the parameter s_i is not late at the initial time, and that is smooth enough that is:

$$s_i(0) = \sigma(-\tau_i),\ s_i \in \mathcal{C}^p. \tag{5}$$

The question is whether the collision avoidance (3) and the tracking delay (4) can be satisfied at once only with (b), that is there exists a parameter $s_i(t)$ satisfying both of them. For example, in the case that the leader turns back toward the followers, they have to go back along the trajectory which the leader just passed. Then, the delays of the parameters become large unboundedly, and (4) could not hold. This paper proposes a collision avoidance method for the LFFN changing the strategies based on (b) to (a) to recover the delays by modifying the trajectory when the delays become too large. The new trajectories $\tilde{\phi}_i(t)$ are generated by the mapping Φ_i as

$$\tilde{\phi}_i(t) = \Phi_i(\phi(s_1(t)), \phi(s_2(t)), \cdots, \phi(s_n(t))), \tag{6}$$

then they always satisfy the inequalities (2) for collision avoidance. The followers surely move without collisions although the modified trajectories might be different from the leader's trajectory.

The mapping Φ_i proposed by Sakurama and Nakano [2007] is valid in (6) if the trajectories do not coincide at every time, that is satisfy

$$\phi(s_i(t)) \ne \phi(s_j(t)),\ \forall (i,j) \in \mathcal{L},\ \forall t. \tag{7}$$

For (7), we demand that the parameters s_i decrease in order of i, that is

$$s_i(t) > s_{i+1}(t),\ \forall t \ge 0,\ \forall i \in \mathcal{N}_n, \tag{8}$$

where $\mathcal{N}_i = \mathcal{N} \backslash \{i\}$. As far as the parameters $s_i(t)$ satisfy (8), the existing method guarantee the collision avoidance (3).

Remark 1. Note that if the trajectory ϕ has a crossing point where $\phi(s_a) = \phi(s_b)$, $s_a \ne s_b$ for some s_a and s_b, (8) does not necessarily guarantee (7). However, (8) is enough to guarantee (7) in practice, because $s_i(t)$ and $s_j(t)$ are rarely equal to s_a and s_b, respectively, at once by accident.

The following sections discuss how to design s_i satisfying (4) and (8) certainly, and satisfying (3) as far as they can. We give some notations as follows:

$$s = [s_1, s_2, \cdots, s_n]^\top,\ \bar{s}_i = s_i - s_{i+1}$$
$$\phi_i(s) = \phi(s_i),\ \bar{\phi}_{ij} = \phi_i - \phi_j$$
$$\hat{\sigma}_i(t) = \sigma(t - \tau_i),\ \hat{\sigma} = [\hat{\sigma}_1, \cdots, \hat{\sigma}_n]^\top,\ \bar{\sigma}_{ij} = \hat{\sigma}_i - \hat{\sigma}_j$$
$$\mathcal{N} = \{1, 2, \cdots, n\},\ \mathcal{N}_i = \mathcal{N} \backslash \{i\},\ \mathcal{N}_{i,j} = \mathcal{N} \backslash \{i, j\}$$

3. DESIGN OF TRAJECTORY PARAMETERS

In this section, we design parameters $s_i(t)$ considering (3) and (4). For (3), the function

$$F(s) = \sum_{(i,j)\in\mathcal{L}} f_{ij}(\|\bar{\phi}_{ij}(s)\|) \tag{9}$$

evaluates the distance between the robots, where f_{ij} is a monotonically decreasing $\mathcal{C}^p$ function. The smaller $f_{ij}(\|\bar{\phi}_{ij}(s)\|)$ is, the larger the distance between $\phi_i(s)$ and $\phi_j(s)$ is. Thus, (3) tends to hold when $F(s)$ is small. In order to take care of the tracking delay (4) in addition to (3), we consider the function

$$V(s) = \frac{1}{2}\sum_{i\in\mathcal{N}} (s_i - \hat{\sigma}_i)^2 + F(s). \tag{10}$$

Because a small $V(s)$ leads to both (3) and (4), s_i should be designed to decrease this function. Note that we can change the follower's trajectory parameters s_i, $i \in \mathcal{N}_1$, but cannot the leader's one s_1, which is always σ. Assume that $f_{ij} = f_{ji}$ and f'_{ij} is a non-positive monotonically increasing function, where $f'_{ij} = \mathrm{d}f_{ij}(x)/\mathrm{d}x$.

The time derivative of $V(s)$ in (10) is given as

$$\frac{\mathrm{d}V}{\mathrm{d}t} = \sum_{i\in\mathcal{N}} \{(\dot{s}_i - \dot{\hat{\sigma}}_i)(s_i - \hat{\sigma}_i + \Gamma_i(s)) + \Gamma_i(s)\dot{\hat{\sigma}}_i\}, \tag{11}$$

where the scaler function $\Gamma_i(s)$ is

$$\Gamma_i(s) = \sum_{j\in\mathcal{N}_i} \gamma_{ij}(s) \tag{12}$$

$$\gamma_{ij}(s) = f'_{ij}(\|\bar{\phi}_{ij}(s)\|)\frac{\bar{\phi}_{ij}(s)^\top}{\|\bar{\phi}_{ij}(s)\|}\phi'_i(s). \tag{13}$$

Note that the negative definiteness of (11) cannot be guaranteed by any $\dot{s}_i$ because of the second term $\Gamma_i(s)\dot{\hat{\sigma}}_i$ in the summation. However, the first term is negative with

$$\dot{s}_i = \dot{\hat{\sigma}}_i - k(s_i - \hat{\sigma}_i) - k\Gamma_i(s),\ k > 0 \tag{14}$$

for a positive constant $k > 0$. The first and second terms force s_i move toward $\hat{\sigma}_i$, and reduce the delay (4). The third term is the function working to avoid collisions by controlling s_i. $\Gamma_i(s)$ in (12) is the summation of the $n-1$ functions $\gamma_{ij}(s)$, which consists of $f'_{ij}(\|\bar{\phi}_{ij}\|)$ and $\bar{\phi}_{ij}^\top\phi'_i/\|\bar{\phi}_{ij}\|$. The former adjusts s_i according to the distance between ϕ_i and ϕ_j. The latter does according to the angular relation between $\bar{\phi}_{ij}$ and ϕ'_i as shown in Fig. 4. The solid curve represents the leader's trajectory ϕ, and $\phi_i = \phi(s_i)$ and $\phi_j = \phi(s_j)$ are the positions of Robots i and j, respectively, on the trajectory. $\phi'_i = \phi'(s_i)$ is the tangent vector of ϕ at the point s_i, and $\bar{\phi}_{ij} = \phi(s_i) - \phi(s_j)$ is the relative vector. In (A), the angle between ϕ'_i and $\bar{\phi}_{ij}$ is small, thus $\bar{\phi}_{ij}^\top\phi'_i$ become large, and then s_i increases quickly for ϕ_i to escape from ϕ_j. In (B), the angle is a little more than $\pi/2$[rad], thus $\bar{\phi}_{ij}^\top\phi'_i$ is negative and its absolute value is small. Then, s_i decreases just a little because in this situation to adjust s_i is not so effective to avoid collisions.

Unfortunately, even if we use the control law (14), $V(s)$ does not necessarily decrease. However, if all the delays are small enough, the parameter control (14) makes $F(s)$ in (9) decease fast, which leads to (3). The following theorem states this fact.

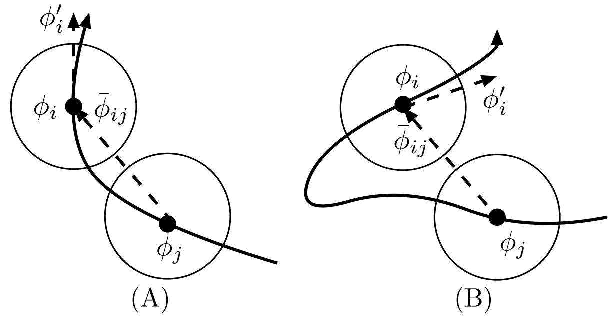


Fig. 4. How s_i moves to avoid collisions

Theorem 1. Assume that f_{ij} and ϕ are of class $\mathcal{C}^p$ and that their first and second derivative and $\dot{\sigma}$ are bounded. Then, if $\|s - \hat{\sigma}\|$ is small enough and the inequalities

$$\|s - \hat{\sigma}\| \le \epsilon_1 \|\Gamma(s)\|, \ \|s - \hat{\sigma}\| \le \epsilon_2 \|\Gamma(s)\|^2 \tag{15}$$

hold for small positive constant ϵ_1 and ϵ_2 and $\Gamma(s) \neq 0$, then

$$\dot{F}(s) < \dot{F}(\hat{\sigma}) \tag{16}$$

holds, where $\Gamma = [0, \Gamma_2, \cdots, \Gamma_n]^\top$.

Proof. $\dot{F}(s) - \dot{F}(\hat{\sigma})$ is calculated as

$$\begin{aligned}\dot{F}(s) - \dot{F}(\hat{\sigma}) &= \sum_{i \in \mathcal{N}_1} \{-k\Gamma_i(s)(s_i - \hat{\sigma}_i + \Gamma_i(s)) \\ &\quad + (\Gamma_i(s) - \Gamma_i(\hat{\sigma}))\dot{\hat{\sigma}}_i)\} \\ &= -k\|\Gamma\|^2 - (k\Gamma^\top - \dot{\hat{\sigma}}^\top \Gamma'(s))(s - \hat{\sigma}) \\ &\quad + O(\|s - \hat{\sigma}\|^2) \sum_{i \in \mathcal{N}_1} \dot{\hat{\sigma}}_i \\ &\le \|\Gamma\|^2 \{-k + \epsilon_1 k + \|\dot{\hat{\sigma}}\|(\epsilon_2 \|\Gamma'(s)\| \\ &\quad + (n-1) O(\|s - \hat{\sigma}\|))\},\end{aligned} \tag{17}$$

where $\Gamma'(s) = \partial\Gamma(s)/\partial s$ and the inequality comes from (15). The right hand side of (17) is positive if ϵ_1, ϵ_2 and $\|s - \hat{\sigma}\|$ are small enough compared with a constant k because all the functions are bounded. Then, $\dot{F}(s) - \dot{F}(\hat{\sigma})$ is negative under the assumption $\Gamma \neq 0$. Thus, (16) holds, which completes the proof. □

Remark 2. Theorem 1 says that in the case that (15) holds, that is the delay $\|s - \hat{\sigma}\|$ is small enough compared with $\|\Gamma(s)\|$, $F(s)$ in (9) deceases faster with the control law (14) than $F(\hat{\sigma})$ without it, and that (3) tends to hold. The proof shows that the larger the control parameter k is, the faster $F(s)$ decreases. Unfortunately, we have no views after the parameters become large by the controller.

Next, consider the parameter delay (4). (14) is rewritten as

$$\frac{\mathrm{d}(s_i - \hat{\sigma}_i)}{\mathrm{d}t} = -k(s_i - \hat{\sigma}_i) - k\Gamma_i(s),$$

which is regarded as the stable system of $s_i - \hat{\sigma}_i$ with the disturbance $-k\Gamma_i(s)$. Thus, the following inequality holds from (12), (13) and Lemma 9.2 in Khalil [2002]:

$$\begin{aligned}|s_i - \hat{\sigma}_i| &\le \sup_{s \in \mathbb{R}^n} |\Gamma_i(s)| \le \sum_{j \in \mathcal{N}_i} \sup_{s_i, s_j \in \mathbb{R}} |\gamma_{ij}(s)| \\ &\le \sup_{u \in \mathbb{R}} \|\phi'(u)\| \sum_{j \in \mathcal{N}_i} |f'_{ij}(0)|.\end{aligned} \tag{18}$$

From the above discussion, the following is given.

Theorem 2. The parameters s_i driven by (14) satisfy (4) for δ_i given by the right hand side of (18).

In order to make the estimation δ_i of the delay small, the curvature $\|\phi'(u)\|$ and the absolute value $|f'_{ij}(0)|$ of the parameter would be better off being small.

4. NONCOINCIDENT CONDITION OF THE TRAJECTORY PARAMETERS

This section discusses the noncoincident condition (8) for s_i controlled by (14), with which the collision avoidance method using the mapping (6) is available. The derivative of $\bar{s}_i$ are calculated with (14) as

$$\dot{\bar{s}}_1 = -k\bar{s}_1 + \dot{\bar{\sigma}}_{12} + k\bar{\sigma}_{12} + k\gamma_{21} + k \sum_{j \in \mathcal{N}_{1,2}} \gamma_{2j} \tag{19}$$

$$\begin{aligned}\dot{\bar{s}}_i &= -k\bar{s}_i + \dot{\bar{\sigma}}_{i,i+1} + k\bar{\sigma}_{i,i+1} + k(\gamma_{i+1,i} - \gamma_{i,i+1}) \\ &\quad + k \sum_{j \in \mathcal{N}_{i,i+1}} (\gamma_{i+1,j} - \gamma_{ij}), \ i \in \mathcal{N}_{1,n}\end{aligned} \tag{20}$$

Under the assumption that (8) holds at the initial time, (8) holds after that time if $\bar{s}_i = 0 \Rightarrow \dot{\bar{s}}_i > 0$, that is

$$\lim_{s_{i+1} \to s_i - 0} \dot{\bar{s}}_i > 0. \tag{21}$$

From (19), the following inequality holds:

$$\begin{aligned}\lim_{s_2 \to s_1 - 0} \dot{\bar{s}}_1 &\ge \inf_{t \ge 0} (\dot{\bar{\sigma}}_{12} + k\bar{\sigma}_{12}) \\ &\quad + k \inf_{s_1 \in \mathbb{R}} \{\|\phi_1'\|(-f'_{12}(0) + \sum_{j \in \mathcal{N}_{1,2}} f'_{2j}(0))\}\end{aligned} \tag{22}$$

If the right hand side of (22) is positive, then (21) holds. The following theorem is given by similar discussions for $i \in \mathcal{N}_{1,n}$:

Theorem 3. If the right hand side of (22) and

$$\begin{aligned}&\inf_{t \ge 0} (\dot{\bar{\sigma}}_{i,i+1} + k\bar{\sigma}_{i,i+1}) + k \inf_{s_i \in \mathbb{R}} [-2f'_{i,i+1}(0)\|\phi_i'\| \\ &+ \sum_{j \in \mathcal{N}_{i,i+1}} \inf_{s_j \in \mathbb{R}} \left\{ (f'_{i+1,j}(\|\bar{\phi}_{ij}\|) - f'_{ij}(\|\bar{\phi}_{ij}\|)) \frac{\bar{\phi}_{ij}^\top}{\|\bar{\phi}_{ij}\|} \phi_i' \right\} \Bigg]\end{aligned} \tag{23}$$

for $i \in \mathcal{N}_{1,n}$ are positive, then (8) holds with the parameter controller (14).

The proof of this theorem is omitted because the lack of space. Although (22) and (23) are difficult to calculate in advance, they present a policy for designing the delays $\hat{\sigma}_i$ and the parameters f_{ij}. For example, $f'_{i,i+1}$ should be small, and $\bar{\sigma}_{i,i+1}$ and $\dot{\bar{\sigma}}_{i,i+1}$ should be large to satisfy (8). In order to simplify the conditions (22) and (23), we choose the parameters f_{ij} as f which is uniform with respect to i and j, and is monotonically decreasing. Then, (22) and (23) are reduced to

$$\inf_{t \ge 0} (\dot{\bar{\sigma}}_{12} + k\bar{\sigma}_{12}) + k \inf_{u \in \mathbb{R}} ((n-3) f'(0) \|\phi'(u)\|) \tag{24}$$

$$\inf_{t \ge 0} (\dot{\bar{\sigma}}_{i,i+1} + k\bar{\sigma}_{i,i+1}) - 2kf'(0) \inf_{u \in \mathbb{R}} \|\phi'(u)\|. \tag{25}$$

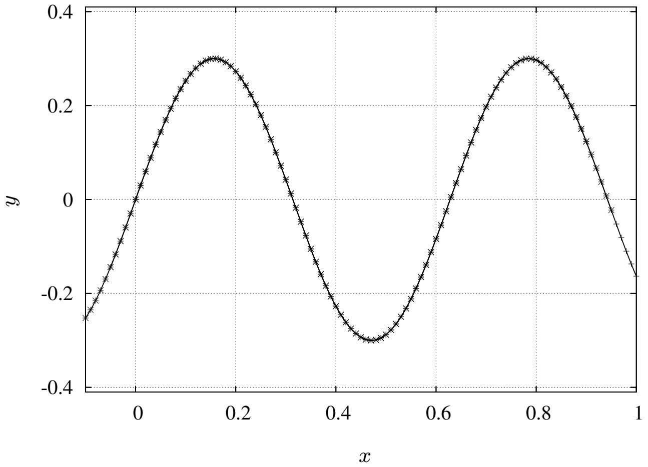


Fig. 5. Original trajectory designed by the leader

5. NUMERICAL EXAMPLE

This section demonstrates the effectiveness of the proposed method by a simulation with mobile robots in a plane. Let the number n of the robots be 3, whose radii are $r_i = 0.05$. The dimension of the space m is 2. The trajectory parameter and the leader's trajectory are given as $\sigma(t) = t$ and $\phi(\sigma) = [0.1\sigma,\ 0.3\sin\sigma]^\top$, respectively. Let the delays of the two followers be $\tau_2 = 0.5$ and $\tau_3 = 1$. Fig. 5 shows the trajectory curve $\phi(\sigma)$ in the x-y plane, where $*$ is marked at every 0.1 second. Note that the marks are dense around the curves, where the leader slows down. The distances $\|\phi_i(\hat{\sigma}) - \phi_j(\hat{\sigma})\|$ between the robots on the trajectory are depicted in Fig. 6 by the dashed-dotted lines, which shows that the distances between Robots 1 and 2, and between 2 and 3 are less than $r_{ij} = 0.1$ at some time. Thus, the robots collide when they move along this trajectory with the assigned delays τ_i.

First, we use only the existing method proposed in Sakurama and Nakano [2007], which modifies the trajectory $\phi(\hat{\sigma}_i(t))$ as $\tilde{\phi}_i(t)$ using the mapping (6) for $s_i(t) = \hat{\sigma}_i(t)$. The modified trajectories $\tilde{\phi}_i(t),\ i = 1, 2, 3$ are depicted by the solid, broken and dashed-dotted lines in Fig. 7, respectively. The distances $\|\tilde{\phi}_i - \tilde{\phi}_j\|$ between the robots on the modified trajectories are depicted in Fig. 6 by the broken lines, which shows that the distances between all pairs are more than $r_{ij} = 0.1$. Thus, (2) holds and the robots on the trajectories avoid collisions. However, as shown in Fig. 7, the followers are considerably away from the leader's trajectory on some sections.

Next, we use the proposed method, which controls the trajectory parameters $s_i(t)$ with (14) before modifying the trajectory ϕ as $\tilde{\phi}_i$ using (6). Let the design parameters be $k = 0.3$ and

$$f_{ij}(x) = \begin{cases} -30(x-0.5)^3, & x \le 0.5 \\ 0, & x > 0.5 \end{cases}$$

for all $(i, j) \in \mathcal{L}$. It follows that (24) and (25) are positive, and then Theorem 3 guarantees (8). The delay (4) is estimated as $\delta_i = 2.5$ by (18). The modified trajectories for Robots $i = 1, 2, 3$ are depicted by the solid, broken and dashed-dotted lines in Fig. 8, respectively. Comparing with Fig. 7, the followers move closer to the leader's trajectory with the proposed method than ones with the existing one. This fact is illustrated by Fig. 9, where the solid and dashed lines show $F(s)$ and $F(\hat{\sigma})$ given in (9), respectively. The smaller F is, the larger the distances between the robots are. In this figure, $F(s)$ is smaller than $F(\hat{\sigma})$, which implies that the controlled parameter s makes the robots move away from each other along the leader's trajectory as noted in Theorem 1. The distances $\|\tilde{\phi}_i - \tilde{\phi}_j\|$ between the robots on the trajectory are depicted in Fig. 6 by the solid lines, which shows that the distances between all pairs are more than $r_{ij} = 0.1$. Thus, (2) holds and the robots on the trajectories avoid collisions. The parameter delays $s_i(t) - \hat{\sigma}_i(t),\ i = 2, 3$ are depicted by the solid and broken lines in Fig. 10, respectively. The absolute value of the delays are less than $\delta_i = 2.5$ estimated by (18).

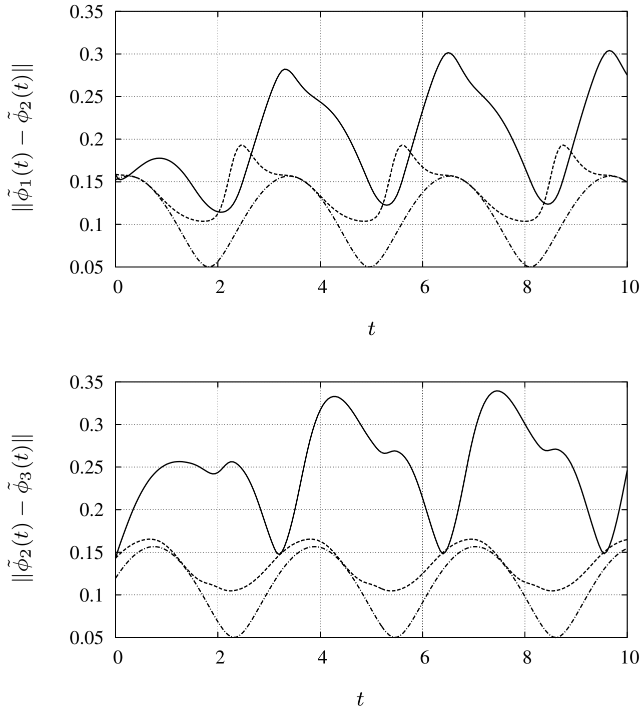


Fig. 6. Distances between the robots

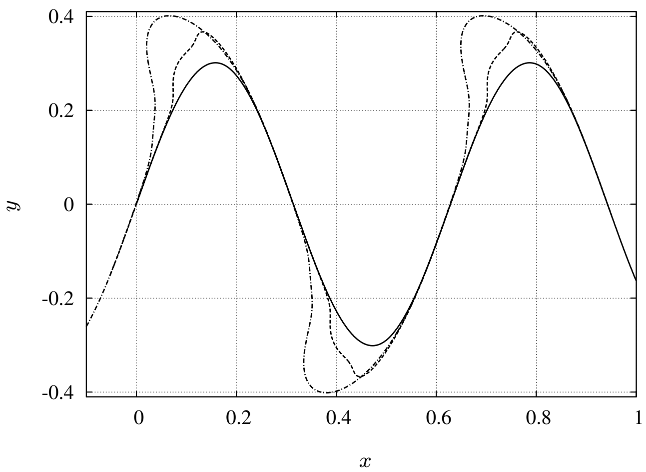


Fig. 7. Modified trajectories with the existing method

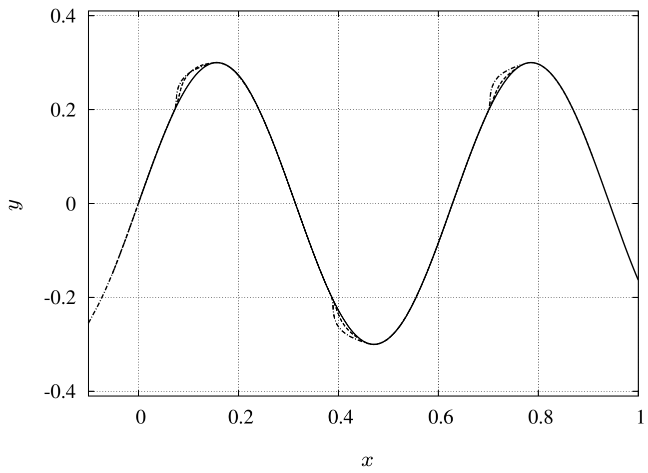


Fig. 8. Modified trajectories with the proposed method

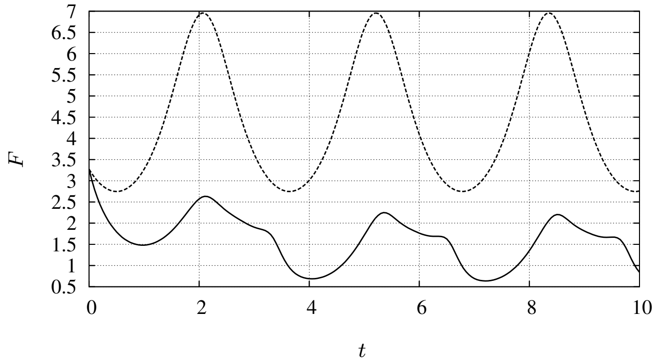


Fig. 9. Estimation of the distances of the robots

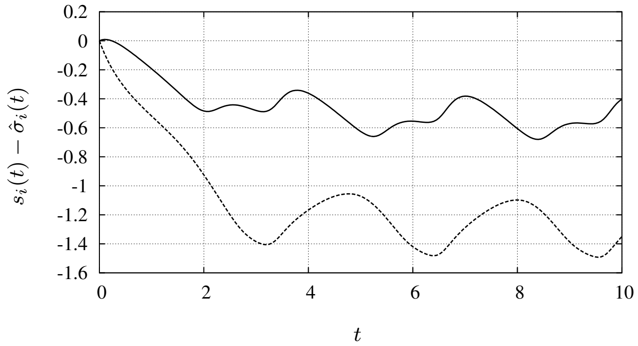


Fig. 10. Delays of the trajectory parameters

This figure shows that the proposed method reduces the tracking errors at the expense of the tracking delays. Note that there exist some design parameters such that the followers can completely move on the leader's trajectory although the delays increase. These results show that the proposed method takes into account both the tracking errors and the delays of the followers. It should be noted that the reference trajectories ϕ are not necessarily given in advance as this simulation because the proposed method is online.

6. CONCLUSION

This paper has dealt with a collision avoidance problem in the LFFN for multiple mobile robots. The proposed method takes into account both tracking errors and delays of the followers from the leader. This method adjusts the velocity for each follower, as well as modifies the shape of the trajectory when a delay becomes too large. The effectiveness of the proposed method is demonstrated by a simulation with three mobile robots.

REFERENCES

S. Akella and S. Hutchinson. Coordinating the motions of multiple robots with specified trajectories. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, page 624–631, 2002.

E. Borhaug, R. Ghabcheloo, A. Pavlov, K.Y. Pettersen, and A. Pascoal. Formation control of underactuated marine vehicles with communication constraints. In *In Proc. of the 7th IFAC Conference on Maneouvring and Control of Marine Craft*, 2006.

X. Chen, A. Serrani, and H. Özbay. Control of leader-folloer formations of terrestrial uavs. In *Proc. of 42th IEEE Conf. on Decision and Control*, 2003.

Y. Chen and Z. Wang. Formation control: a review and a new consideration. In *In Proceeding in Int. Conf. on Intelligent Robots and Systems*, 2003.

T. Ikeda, J. Jongusuk, T. Ikeda, and T. Mita. Formation control of multiple nonholonomic mobile robots. *Electrical Engineering in Japan*, 157(3):814–819, 2006.

H. K. Khalil. *Nonlinear Systems*. Prentice Hall, New Jersey, third edition, 2002.

Y. Li and X. Chen. Leader-formation navigation using dynamic formation pattern. In *Proc. in the 2005 IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics*, page 1494–1499, 2005.

G.L. Mariottini, G. Pappas, D. Prattichizzo, and K. Daniilidis. Vision-based localization of leader-follower formations. In *Proc. of 43th IEEE Conf. on Decision and Control, and the European Control Conference 2005*, 2005.

K. Sakurama and K. Nakano. Online modification of reference trajectories for multiple robots with collision avoidance. In *Proc. of 45th IEEE Conf. on Decision and Control*, 2006.

K. Sakurama and K. Nakano. Online modification of reference trajectories for multiple robots with collision avoidance — a method using relative angles between robots —. In *Proc. of 46th IEEE Conf. on Decision and Control*, 2007.

D.P. Scharf, Fred Y. Hadaegh, and S.R. Ploen. A survey of spacecraft formation flying guidance and control (part i): Guidance. In *Proc. of American Control Conference*, 2003.

E. Semsar and K. Khorasani. Adaptive formation control of uavs in the presence of unknown vortex forces and leader commands. In *Proc. of American Control Conference*, 2006.

J. Shao, J. Yu G. Xie, and L. Wang. Leader-following formation control of multiple mobile robots. In *In Proc. of the 2005 IEEE Int. Symp. of Intelligent Control*, page 808–813, 2005.