2012 IEEE International Conference on Control Applications (CCA)
Part of 2012 IEEE Multi-Conference on Systems and Control
October 3-5, 2012. Dubrovnik, Croatia

Navigation Function of Mobile Robots for Formation Driving

Masahiro YAMADA, Yuh YAMASHITA, and Daisuke TSUBAKINO

Abstract— **This paper investigates a vehicle formation control based on the following of a leader by nonholonomic mobile robots with two driving wheels. In this research, we propose a method for creating continuous and non-differentiable navigation functions, from which we derive a formation control law for each robot. The non-differentiability of the proposed navigation function removes the singularity encountered by conventional methods. The stability condition of a formation based on leader-following has been demonstrated by Tanner et al. as "Leader-to-Formation Stability" (LFS) [1]. In this paper, we assess the LFS property by regarding the navigation function as a LFS Lyapunov function.**

I. INTRODUCTION

Recently, the use of transport systems, such as automated highway systems [2] and control groups of unmanned vehicles [3], has increased, and their efficiency has attracted a great deal of interest. One of the keys to this efficiency is to make the agents in the transport systems move in formation. Advantages of the formation control include the realization of mass transportation by a few drivers, a decrease in air resistance, and an improvement in road-usage capacity by shortening the gaps between vehicles. In a formation control, each agent moves to its desired position and avoids collisions with other agents.

In this paper, we investigate autonomous driving by the creation of a desired formation based on a leader-following control. Conventional methods of formation creation often have singular points where the control inputs diverge or the movement of the vehicle stagnates. For example, the ℓ-ℓ controller proposed by Desai et al. [4] has singularities on the straight line that connects two reference agents. In order to solve such a problem, we adopt the idea of a navigation function, which is often used in obstacle avoidance problems [5]. Each follower agent has a navigation function, and moves to its desired position according to the positions of other, hierarchically superior robots. Therefore, the navigation function method achieves the creation of a formation in a distributed cooperative control manner. However, the motion of the robot becomes sluggish on the saddle points and local maxima of a differentiable navigation function. In this paper, we propose a design method for a control law realizing a desired formation via continuous and non-differentiable navigation functions, which have no undesired equilibrium points or singularities.

It is very important that the formations created by the proposed method are stable, in order that they are sustainable. We assume that each robot has no knowledge of the other robots' input values, and, in particular, the input to the leader robot, which can be considered as the disturbance in the formation control system. The concept of the stability of the formation based on leader-following has been introduced as "Leader-to-Formation Stability" (LFS) by Tanner et al. [1]. In this paper, we assess the stability of the formation in the sense of LFS.

II. PRELIMINARIES

A. Formation definition

In this paper, we call the foremost mobile robot the leader, and denote the i-th follower robot as follower i. The state of these robots is distinguished by indices, i.e., x_0 denotes the state of the leader, and x_i refers to the state of follower i.

We consider the formation to consist of nonholonomic mobile robots with two driving wheels, as illustrated in Fig. 1. For each mobile robot, we consider the following kinematic model:

$$\dot{X}_i = v_i \cos\theta_i,\ \dot{Y}_i = v_i \sin\theta_i,\ \dot{\theta}_i = \omega_i,$$

where (X_i, Y_i) is the center position of robot i's wheel axle, θ_i is its heading angle, and v_i, ω_i are the translational and rotational velocity control inputs, respectively. We set the state x_i and input u_i of robot i as follows:

$$x_i = (X_i, Y_i, \theta_i)^T,\ u_i = (v_i, \omega_i)^T.$$

The output of each mobile robot is defined as the coordinate of its foremost point

$$y_i = (X_i + d\cos\theta_i,\ Y_i + d\sin\theta_i)^T,$$

where d is the distance between the reference point and the center of the wheel axle. By adopting the offset d, we can use an input-to-output linearization technique in Sections Ⅲ and Ⅳ.

B. Problem settings

The control objectives in this research are the asymptotic creation of a formation in a predetermined pattern, and collision avoidance among the robots. We assume that each robot can observe the position and orientation, but not the velocity, of other robots. There is a hierarchy among the leader and the followers. Each follower adjusts its own position, such that it can form the desired pattern, according to the positions of

This work was supported by JSPS, Grant-in-Aid for Scientific Research(B), 22360167.

M. Yamada, Y. Yamashita and D. Tsubakino are with Graduate School of Information Science and Technology, Hokkaido University, Sapporo, Hokkaido 060-0814 JAPAN m-yamada@stl.ssi.ist.hokudai.ac.jp, yuhyama@ssi.ist.hokudai.ac.jp, tsubakino@ssi.ist.hokudai.ac.jp

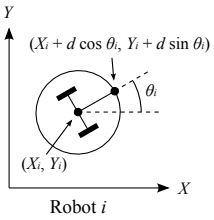

Fig. 1. Robot model

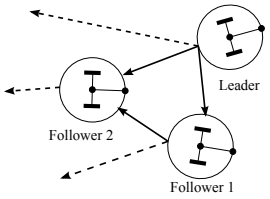

Fig. 2. Information flow

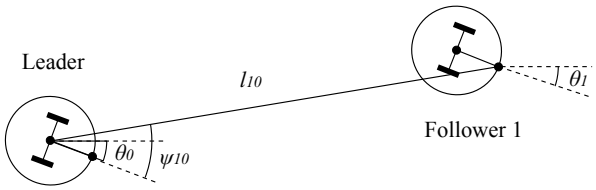

Fig. 3. Specification for the leader–follower 1 relationship

hierarchically superior robots. The responsibility for collision avoidance is placed on the inferior robots. The hierarchical structure is expressed as a directed acyclic graph (DAG) in Fig. 2. The DAG also indicates the information flow. In this paper, we suppose that the leader of the overall formation is unique, so there exists a root in the DAG. In general, the rank order among robots cannot be uniquely determined from the DAG structure alone. However, in this paper, we dictate the order of the robots as: leader, follower 1,..., follower n, which is consistent with the DAG structure.

C. Leader-to-formation stability

The concept of stability in a formation was introduced as Leader-to-Formation Stability (LFS) by Tanner et al. [1]. LFS provides nonlinear gain estimates that quantify the transient effects of the initial error and the steady-state effects of leader inputs on the amplitude of the formation error.

We define the formation error of robot i as $\tilde{x}_i = \xi_i(x_0, \ldots, x_i)$, and the overall formation error as $\tilde{x} = (\tilde{x}_0^T, \ldots, \tilde{x}_n^T)^T$. The error dynamics of the overall formation with the input u_0 to the leader are expressed as

$$\dot{\tilde{x}} = f(t, \tilde{x}, u_0).$$

Though Tanner et al. have introduced the LFS of the formation for the case of multiple-leaders, we define the LFS for the single leader case as follows.

Definition 1 (LFS): The formation is called LFS if there exists a class KL function β and a class K function γ such that, for any initial formation error $\widetilde{x}(0)$ and for any bounded disturbances from the formation leader u_0, the formation error satisfies

$$\|\widetilde{x}(t)\| \leq \beta(\|\widetilde{x}(0)\|, t) + \gamma(\sup_{\tau \in [0,t]} \|u_0(\tau)\|). \tag{1}$$

The functions $\beta(r,t)$ and $\gamma(r)$ are called the transient and asymptotic LFS gains for the formation, respectively.

LFS is based on the notion of Input-to-State Stability (ISS). From the necessary condition of ISS [6], Tanner et al. derived the following theorem:

Theorem 1 (Necessary condition of LFS): A formation is LFS if there is a radially unbounded, positive-definite function V: $R^n \to R^+$ such that

$$\dot{V} \leq -a(\|\widetilde{x}\|) \ + \ b(\|u_0\|), \tag{2}$$

where $a \in K_\infty, b \in K_\infty$.

We will assess the LFS properties of subsystems of the overall formation by checking (2) for the navigation functions described later. It is shown that, if each leader–follower pair is LFS, the overall formation is also LFS.

III. CONTROL OF FOLLOWER 1

A. Construction of navigation function

In this section, we propose a control method for follower 1. To control follower 1, we do not have to consider the position of follower 2, because only the leader occupies a higher level in the hierarchy. We consider the coordinate system illustrated in Fig. 3 in order to realize the intended formation and avoid a collision with the leader. The variables l_{10}, ψ_{10} in Fig. 3 are defined as

$$l_{10} = \sqrt{(X_1 + d\cos\theta_1 - X_0)^2 + (Y_1 + d\sin\theta_1 - Y_0)^2}$$
$$\psi_{10} = \arg(X_1 + d\cos\theta_1 - X_0, Y_1 + d\sin\theta_1 - Y_0) - \theta_0,$$

where $\arg(X,Y)$ is a multi-valued function denoting the angle of the point (X,Y), i.e.,

$$\arg(X,Y) = \begin{cases} \{\text{atan}(Y/X) + 2i\pi | i \in \mathbb{I}\}, & (X > 0) \\ \{\pi/2 + 2i\pi | i \in \mathbb{I}\}, & (X = 0, Y > 0) \\ \{\text{atan}(Y/X) + (2i+1)\pi | i \in \mathbb{I}\}, & (X < 0) \\ \{-\pi/2 + 2i\pi | i \in \mathbb{I}\}, & (X = 0, Y < 0) \\ \mathbb{R}, & (X = Y = 0). \end{cases}$$

Therefore, ψ_{10} is a set-valued function.

We design the following navigation function V_1 for follower 1:

$$\begin{aligned} V_1(l_{10}, \psi_{10}) = \min_{\psi \in \psi_{10}} k_1 \Bigg[& \frac{1}{2}(l_{10} - l_{10des})^2 \\ & + k_a \Bigg\{ (a - l_{10des}) \log\left(\frac{l_{10} - a}{l_{10des} - a} \right) \\ & + (l_{10} - l_{10des}) \Bigg\} \Bigg] + \frac{1}{2} k_2 (\psi_{10} - \psi_{10des})^2, \end{aligned} \tag{3}$$

where

$$l_{10des} = \sqrt{x_{1des}^2 + y_{1des}^2}, \ \psi_{10des} = \arg(x_{1des}, y_{1des}).$$

Fig. 4 shows the proposed navigation function in the (X_1, Y_1) coordinates. The navigation function is positive-definite, and its origin is chosen as the objective point. The area around the leader is not included in the domain of the navigation function. Collision avoidance is ensured because the navigation function diverges to infinity near the prohibited area.

Moreover, the navigation function does not have a point where its gradient equals zero, except at the origin, due to its non-differentiability. The possibility of a robot moving sluggishly at an undesired point, which is often a problem

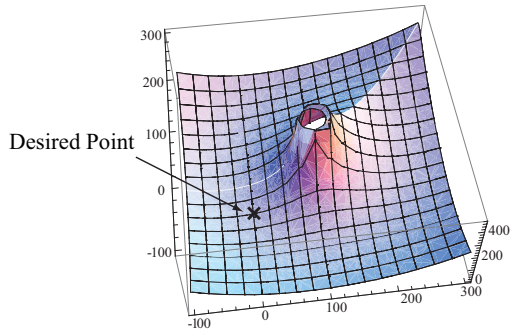

Fig. 4. Navigation function of follower 1

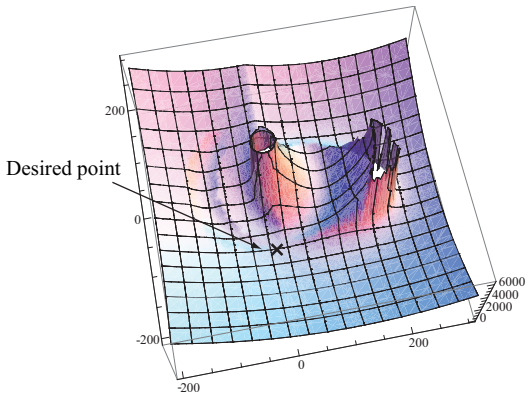

Fig. 5. Navigation function for follower 2

in the conventional navigation function method, is removed by the proposed method.

Once a navigation function is obtained, we can derive the control law from it. The dynamics of follower 1's position relative to the leader in Fig. 3 can be written as

$$\begin{bmatrix} \dot{l}_{10} \\ \dot{\psi}_{10} \end{bmatrix} = F_{10} \begin{bmatrix} v_1 \\ \omega_1 \end{bmatrix} - \begin{bmatrix} \cos\psi_{10} & 0 \\ -\sin\frac{\psi_{10}}{l_{10}} & 1 \end{bmatrix} \begin{bmatrix} v_0 \\ \omega_0 \end{bmatrix}, \tag{4}$$

where F_{10} is given by

$$F_{10} = \begin{bmatrix} \cos(\psi_{10} - \theta_1 + \theta_0) & d\sin(\psi_{10} - \theta_1 + \theta_0) \\ -\frac{\sin(\psi_{10} - \theta_1 + \theta_0)}{l_{10}} & \frac{d\cos(\psi_{10} - \theta_1 + \theta_0)}{l_{10}} \end{bmatrix}.$$

Using input-to-output linearization, we can choose an input for follower 1 as

$$\begin{bmatrix} v_1 \\ \omega_1 \end{bmatrix} = F_{10}^{-1} \begin{bmatrix} -k_1 \widetilde{l}_{10}(1 + \frac{k_a}{l_{10} - a}) \\ -k_2 \widetilde{\psi}_{10} \end{bmatrix}, \tag{5}$$

where $\widetilde{l}_{10} = l_{10} - l_{10des}$ and $\widetilde{\psi}_{10} = \psi_{10} - \psi_{10des}$.

B. LFS property in formation error dynamics

By substituting the input (5) into the dynamics of the leader–follower 1 relative position (4), we obtain

$$\begin{bmatrix} \dot{l}_{10} \\ \dot{\psi}_{10} \end{bmatrix} = - \begin{bmatrix} k_1 \widetilde{l}_{10}(1 + \frac{k_a}{l_{10} - a}) \\ k_2 \widetilde{\psi}_{10} \end{bmatrix} - \begin{bmatrix} \cos\psi_{10} & 0 \\ -\frac{\sin\psi_{10}}{l_{10}} & 1 \end{bmatrix} \begin{bmatrix} v_0 \\ \omega_0 \end{bmatrix}.$$

Thus, the time derivative of the positive-definite function (3) is given by

$$\dot{V}_1 = \frac{\partial V_1}{\partial l_{10}} \dot{l}_{10} + \frac{\partial V_1}{\partial \psi_{10}} \dot{\psi}_{10} = -\widetilde{z}_1^T Q_1 \widetilde{z}_1 + \widetilde{z}_1^T W_{10} w_0,$$

$$Q_1 = \begin{bmatrix} k_1^2 (1 + \frac{k_a}{l_{10} - a})^2 & 0 \\ 0 & k_2^2 \end{bmatrix}$$

$$W_{10} = \begin{bmatrix} -k_1 (1 + \frac{k_a}{l_{10} - a}) \cos\psi_{10} & 0 \\ k_2 \frac{\sin\psi_{10}}{l_{10}} & -k_2 \end{bmatrix},$$

where $\widetilde{z}_1 = (\widetilde{l}_{10}, \widetilde{\psi}_{10})^T$ and $w_0 = (v_0, \omega_0)^T$. This leads to

$$\dot{V}_1 \le -\frac{1}{2} \widetilde{z}_1^T Q_1 \widetilde{z}_1 + \frac{1}{2} w_0^T W_{10}^T Q^{-1} W_{10} w_0,$$

which satisfies the LFS condition (2). Therefore, the leader–follower 1 relative dynamics have the LFS property.

IV. CONTROL OF FOLLOWER 2

A. Construction of navigation function

The control objectives of follower 2 are to create the intended formation and avoid colliding with the leader and follower 1. The formation error of follower 2 is affected by the inputs of the leader and follower 1, and the formation error between the leader and follower 1. We now design a navigation function for follower 2. In order to avoid a collision, the navigation function must have two holes—prohibited areas—in its domain, and the value of the function becomes infinite at the edges of these holes. We synthesize the navigation function from two local navigation functions by taking the minimal value of these two functions at each point, as in Fig. 5. The navigation function is defined as

$$V_2 = \min(V_{2in}, V_{2out}), \tag{6}$$

where V_{2out} is a local navigation function defined outside the ellipse that fences the leader and follower 1, and V_{2in} is another local navigation function that has a donut-shape region around follower 1 as its domain. We call V_{2out} an "outside function" and V_{2in} an "inside function."

1) Outside function: We consider a coordinate system such as in Fig. 6 so that follower 2 can move to the desired point and avoid a collision with the leader or follower 1. In Fig. 6, l_{2out} and ψ_{21} are expressed as

$$\begin{aligned} l_{2out} = {} & \{(X_2 + d\cos\theta_2 - X_0)^2 + (Y_2 + d\sin\theta_2 - Y_0)^2\}^{\frac{1}{2}} \\ & + \{(X_2 + d\cos\theta_2 - X_1 - d\cos\theta_1)^2 \\ & + (Y_2 + d\sin\theta_2 - Y_1 - d\sin\theta_1)^2\}^{\frac{1}{2}} \\ \psi_{21} = {} & \arg(X_2 + d\cos\theta_2 - X_1 - d\cos\theta_1, Y_2 + d\sin\theta_2 \\ & - Y_1 - d\sin\theta_1) - \arg(X_1 + d\cos\theta_1 - X_0, \\ & Y_1 + d\sin\theta_1 - Y_0). \end{aligned}$$

The outside local navigation function V_{2out} is designed as

$$\begin{aligned} V_{2out}(l_{2out}, \psi_{21}) = \min_{\psi \in \psi_{21}} k_1 \Bigg[& \frac{1}{2}(l_{2out} - l_{2outdes})^2 + \\ & k_a \Bigg\{ (a - l_{2outdes}) \log\left(\frac{l_{2out} - a}{l_{2outdes} - a} \right) \\ & + (l_{2out} - l_{2outdes}) \Bigg\} \Bigg] + \frac{1}{2} k_2 (\psi_{21} - \psi_{21des})^2, \end{aligned} \tag{7}$$

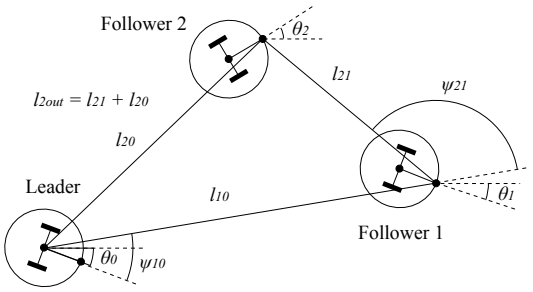

Fig. 6. Specification for the robots' relationship in the outside function

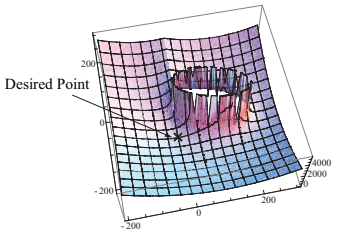

Fig. 7. Outside function for follower 2

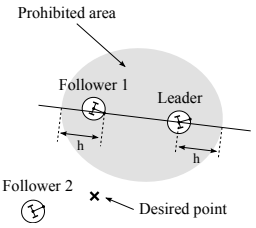

Fig. 8. Domain of the outside function

where

$$
\begin{aligned}
l_{2outdes} &= \sqrt{(X_{2des} - X_{1des})^2 + (Y_{2des} - Y_{1des})^2} \\
&\quad + \sqrt{X_{2des}^2 + Y_{2des}^2} \\
\psi_{21des} &= \arg(X_{2des} - X_{1des}, Y_{2des} - Y_{1des}) \\
&\quad - \arg(X_{1des}, Y_{1des}) \\
a &= \sqrt{X_{1des}^2 + Y_{1des}^2} + 2h.
\end{aligned}
$$

Fig. 7 indicates the shape of the outside local navigation function, and its domain is illustrated in Fig. 8. The navigation function is positive-definite, and its origin is chosen to be the objective point. The inside of the ellipse in Fig. 8, which fences the leader and follower 1, is not included in the domain of the local navigation function. Collision avoidance is automatic, because the navigation function diverges to infinity near the prohibited area. Moreover, the non-differentiability prevents sluggish motion far from the goal position.

A control law can be derived from the navigation function. The formation error dynamics in Fig. 6 can be written as

$$
\begin{bmatrix} \dot{l}_{2out} \\ \dot{\psi}_{21} \end{bmatrix} = F_{20out} \begin{bmatrix} v_2 \\ \omega_2 \end{bmatrix} - G_{21out} \begin{bmatrix} v_1 \\ \omega_1 \end{bmatrix} - G_{20out} \begin{bmatrix} v_0 \\ w_0 \end{bmatrix}, \quad (8)
$$

$$
F_{2out} = \begin{bmatrix} \cos(\psi_{21} + \phi_{12}) + \cos(\psi_{20} + \phi_{12}) & \\ -\frac{\sin(\psi_{21}+\phi_{12})}{l_{21}} & \\ d(\sin(\psi_{21} + \phi_{12}) + \sin(\psi_{20} + \phi_{12})) & \\ \frac{d\cos(\psi_{21}+\phi_{12})}{l_{21}} & \end{bmatrix}
$$

$$
G_{21out} = \begin{bmatrix} \cos(\psi_{21} + \phi_{11}) & \\ -\{\frac{\sin\phi_{11}}{l_{10}} + \frac{\sin(\psi_{21}+\phi_{11})}{l_{21}}\} & \\ -d\sin(\psi_{21} + \phi_{11}) & \\ d\{\frac{\cos\phi_{11}}{l_{10}} + \frac{\cos(\psi_{21}+\phi_{11})}{l_{21}}\} & \end{bmatrix}
$$

$$
G_{20out} = \begin{bmatrix} \cos(\psi_{20} + \psi_{10}) & 0 \\ \frac{\sin\psi_{10}}{l_{10}} & 0 \end{bmatrix},
$$

where $\phi_{12} = \psi_{10} + \theta_0 - \theta_2$ and $\phi_{11} = \psi_{10} + \theta_0 - \theta_1$. Using the input-to-output linearization technique, the decreasing input value of the navigation function can be designed as

$$
\begin{bmatrix} v_2 \\ \omega_2 \end{bmatrix} = F_{2out}^{-1} \begin{bmatrix} -k_1 \widetilde{l}_{2out}(1 + \frac{k_a}{l_{2out}-a}) \\ -k_2 \widetilde{\psi}_{21} \end{bmatrix}, \quad (9)
$$

where $\widetilde{l}_{2out} = l_{2out} - l_{2outdes}$ and $\widetilde{\psi}_{21} = \psi_{21} - \psi_{21des}$. The determinant of F_{2out} becomes zero on the segment between the leader and follower 1. Hence, it seems that input-to-output linearization is impossible on this segment. Fortunately, the segment is on the inside of the ellipse, and F_{2out} is regular on the domain of the control law.

2) Inside function: According to the control law (9), follower 2 cannot transit between the leader and follower 1 because this gap is excluded from its domain. follower 2 often takes a circuitous route due to this constraint, e.g., when its initial position is opposite the desired position. Thus, we construct another local navigation function using the coordinate system in Fig. 9. In Fig. 9, l_{2in} and ψ_{21} are defined as

$$
\begin{aligned}
l_{2in} &= \{(X_2 + d\cos\theta_2 - X_c)^2 + (Y_2 + d\sin\theta_2 - Y_c)^2\}^{\frac{1}{2}} \\
&\quad + \{(X_2 + d\cos\theta_2 - X_1 - d\cos\theta_1)^2 \\
&\quad + (Y_2 + d\sin\theta_2 - Y_1 - d\sin\theta_1)^2\}^{\frac{1}{2}} \\
\psi_{21} &= \arg(X_2 + d\cos\theta_2 - X_1 - d\cos\theta_0, Y_2 + d\sin\theta_2 \\
&\quad - Y_1 - d\sin\theta_1) - \arg(X_1 + d\cos\theta_1 - X_0, \\
&\quad Y_1 + d\sin\theta_1 - Y_0),
\end{aligned}
$$

where

$$
\begin{aligned}
X_c &= X_1 + d\cos\theta_1 + r_c\cos\{\psi_{21des} \\
&\quad + \arg(X_1 + d\cos\theta_1 - X_0, Y_1 + d\sin\theta_1 - Y_0)\} \\
Y_c &= Y_1 + d\sin\theta_1 + r_c\sin\{\psi_{21des} \\
&\quad + \arg(X_1 + d\cos\theta_1 - X_0, Y_1 + d\sin\theta_1 - Y_0)\} \\
r_c &= \sqrt{(X_{2des} - X_{1des})^2 + (Y_{2des} - Y_{1des})^2} \\
&\quad - \sqrt{X_{1des}^2 + Y_{1des}^2}/2 \\
\psi_{21des} &= \arg(X_{2des} - X_{1des}, Y_{2des} - Y_{1des}) \\
&\quad - \arg(X_{1des}, Y_{1des}).
\end{aligned}
$$

The inside local navigation function V_{2in} for follower 2 is defined as

$$
\begin{aligned}
V_{2in}(l_{2in}, \psi_{21}) &= \min_{\psi \in \psi_{21}} k_1 \left[\frac{1}{2}(l_{2in} - l_{12indes})^2 \right. \\
&+ k_a \left\{ (l_{2in} - l_{2indes}) + (a - l_{2indes}) \log\left(\frac{l_{2in} - a}{l_{2indes} - a} \right) \right\} \\
&\left. - k_b \left\{ (l_{2in} - l_{2indes}) + (b - l_{2indes}) \log\left(\frac{b - l_{2in}}{b - l_{2indes}} \right) \right\} \right] \\
&+ \frac{1}{2} k_2 (\psi_{21} - \psi_{21des})^2,
\end{aligned} \quad (10)
$$

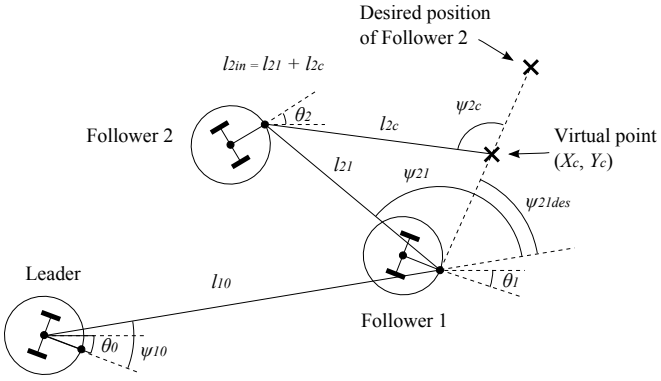

Fig. 9. Specification for the leader–follower 1–follower 2 relationship in the inside function

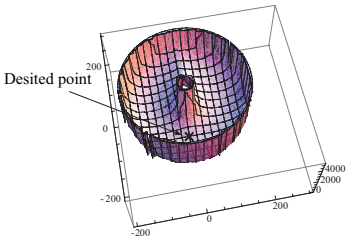

Fig. 10. Inside function for follower 2

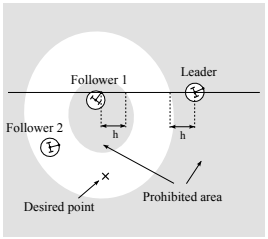

Fig. 11. Domain of the inside function

where

$$l_{2indes} = \sqrt{(X_{2des} - X_{1des})^2 + (Y_{2des} - Y_{1des})^2} + [\{X_{2des} - X_{1des} - r_c\cos(\psi_{21des} + \arg(X_{1des}, Y_{1des}))\}^2 + \{Y_{2des} - Y_{1des} - r_c\sin(\psi_{21des} + \arg(X_{1des}, Y_{1des}))\}^2]^{\frac{1}{2}}$$

$$a = h + \sqrt{(r_c\cos\psi_{21des} + h)^2 + (r_c\sin\psi_{21des})^2}$$

$$b = \sqrt{X_{1des}^2 + Y_{1des}^2} - h + \{(r_c\cos\psi_{21des} + \sqrt{X_{1des}^2 + Y_{1des}^2} - h))^2 + (r_c\sin\psi_{21des})^2\}^{\frac{1}{2}}.$$

Fig. 10 shows the shape of the inside local navigation function, and Fig. 11 indicates its domain. The navigation function is positive-definite and its origin is chosen to be the objective point. The domain of the inside function excludes the positions of follower 1 and the leader. Thus, collision avoidance is performed automatically, because the navigation function diverges to infinity near the prohibited area.

We derive the control law from the inside navigation function. The formation error dynamics in Fig. 9 can be written as

$$\begin{bmatrix} \dot{l}_{2out} \\ \dot{\psi}_{21} \end{bmatrix} = F_{2in}\begin{bmatrix} v_2 \\ \omega_2 \end{bmatrix} - G_{21in}\begin{bmatrix} v_1 \\ \omega_1 \end{bmatrix} - G_{20in}\begin{bmatrix} v_0 \\ w_0 \end{bmatrix},$$

$$F_{2in} = \begin{bmatrix} \cos(\psi_{21} + \phi_{12}) + \cos(\psi_{2c} + \psi_{21des} + \phi_{12}) & -\frac{\sin(\psi_{21}+\phi_{12})}{l_{21}} \\ d(\sin(\psi_{21} + \phi_{12}) + \sin(\psi_{2c} + \psi_{21des} + \phi_{12})) & \frac{d\cos(\psi_{21}+\phi_{12})}{l_{21}} \end{bmatrix}$$

The components of the matrices G_{21in} and G_{20in} are functions of the robot positions, which are bounded in the domain of the inside function. The precise expression of these matrices is omitted in this paper, as they are not required for the control law design. These matrices are also unnecessary for verifying the LFS property. Using the input-to-output linearization, the input of follower 2 can be derived as

$$\begin{bmatrix} v_2 \\ \omega_2 \end{bmatrix} = F_{2in}^{-1}\begin{bmatrix} -k_1\widetilde{l}_{2in}(1 + \frac{k_a}{l_{2in}-a})(1 + \frac{k_b}{b-l_{2in}}) \\ -k_2\widetilde{\psi}_{21} \end{bmatrix}, \quad (11)$$

where $\widetilde{l}_{2in} = l_{2in} - l_{2indes}$ and $\widetilde{\psi}_{21} = \psi_{21} - \psi_{21des}$. Note that the points where the matrix F_{2in} becomes singular are not included in the domain of the inside function.

follower 2 adopts the control law corresponding to the active function in (6).

B. Scaling

In the design of the navigation functions proposed in the previous subsection, it is assumed that follower 1 has already moved to its desired position. In actual systems, the navigation function of follower 2 must be adjusted depending on the position of follower 1. For example, in the outside function, when the distance between the leader and follower 1 is greater than desired, the prohibited ellipse should become larger and the position of follower 2 must be evaluated by a new scale. To make such a scaling, we define

$$l'_{2out} = \frac{l_{10des}}{l_{10}}l_{2out},\ l'_{2in} = \frac{l_{10des}}{l_{10}}l_{2in}, \quad (12)$$

and use these variables instead of l_{2out} and l_{2in}, respectively, in the control laws.

C. LFS property in formation error dynamics

We show that the formation given by the outside function has the LFS property. By substituting the input (9) into the leader–follower formation error dynamics (8), we obtain

$$\begin{bmatrix} \dot{l}_{2out} \\ \dot{\psi}_{21} \end{bmatrix} = -\begin{bmatrix} k_1\widetilde{l}_{2out}(1 + \frac{k_a}{l_{2out}-a}) \\ k_2\widetilde{\psi}_{21} \end{bmatrix} - G_{21out}\begin{bmatrix} v_1 \\ \omega_1 \end{bmatrix} - G_{20out}\begin{bmatrix} v_0 \\ w_0 \end{bmatrix}.$$

Thus, the time derivative of the outside function (7) is

$$\dot{V}_{2out} = \frac{\partial V_{2out}}{\partial l_{2out}}\dot{l}_{2out} + \frac{\partial V_{2out}}{\partial \psi_{21}}\dot{\psi}_{21} = -\widetilde{z}_2^T Q_2\widetilde{z}_2 + \widetilde{z}_2^T W_{21}w_1 + \widetilde{z}_2^T W_{20}w_0,$$

$$Q_2 = \begin{bmatrix} k_1^2(1 + \frac{k_a}{l_{2out}-a})^2 & 0 \\ 0 & k_2^2 \end{bmatrix}$$

$$W_{21} = \begin{bmatrix} k_1(1 + \frac{k_a}{l_{2out}-a})\cos(\psi_{21} + \phi_{11}) & k_1(1 + \frac{k_a}{l_{2out}-a})d\sin(\psi_{21} + \phi_{11}) \\ -k_2\{\frac{\sin\phi_{11}}{l_{10}} + \frac{\sin(\psi_{21}+\phi_{11})}{l_{21}}\} & k_2 d\{\frac{-\cos\phi_{11}}{l_{10}} + \frac{\cos(\psi_{21}+\phi_{11})}{l_{21}}\} \end{bmatrix}$$

$$W_{20} = \begin{bmatrix} k_1(1 + \frac{k_a}{l_{2out}-a})\cos(\psi_{20} + \psi_{10}) & 0 \\ k_2\frac{\sin(\psi_{20}+\psi_{10})}{l_{10}} & 0 \end{bmatrix},$$

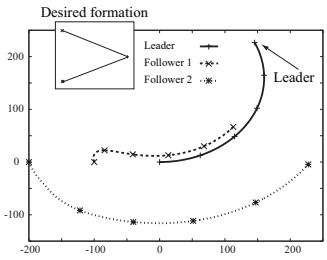

Fig. 12. Robot paths in which the outside function is adopted in the initial position

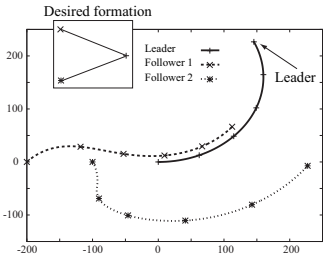

Fig. 13. Robot paths in which the inside function is adopted in the initial position

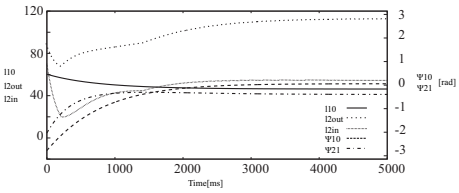

Fig. 14. Time series of the formation error

Fig. 15. Pololu 3pi robot

Fig. 16. USB camera image

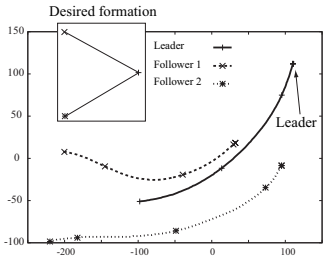

Fig. 17. Paths taken by real mobile robots

where $\tilde{z}_2^T = (\tilde{l}_{2out}, \tilde{\psi}_{21})^T$, $w_1^T = (v_1, \omega_1)^T$ and $w_0^T = (v_0, \omega_0)^T$. This leads to

$$\dot{V}_{2out} \leq -\frac{1}{3}\tilde{z}_2^T Q_2 \tilde{z}_2 + \frac{3}{4} w_1^T W_{21}^T Q_2^{-1} W_{21} w_1 + \frac{3}{4} w_0^T W_{20}^T Q_2^{-1} W_{20} w_0,$$

which satisfies the LFS condition (2). Therefore, the relative position dynamics have the LFS property.

When follower 2 is far enough from its reference point, the outside function is always adopted. Therefore, the formation error dynamics for follower 2 under the proposed control law have the LFS property.

V. SIMULATIONS

We performed a simulation for the case of three mobile robots in a triangular formation. Fig. 12 shows the paths of the three robots when the leader moves in a circle. In this simulation, follower 2 initially uses the outside function. Each follower tracks their desired position, assigned as a vertex of the triangle, according to the movement of the leader. Fig. 14 represents the time response of the formation errors, where the relative positions of the followers remain bounded as a consequence of the LFS.

The simulation result for the case where follower 2 initially uses the inside function is shown in Fig. 13. As in the case depicted in Fig. 12, the followers maintain the intended formation.

VI. EXPERIMENTS

We performed experiments using the "Pololu 3pi robot"(Fig. 15) as mobile robots in order to confirm the usefulness of the proposed method.

In the experiment environment, a PC obtains the position and orientation of each robot using an image from a USB camera fixed to the ceiling, calculates velocity inputs according to the navigation functions, and sends these inputs to the mobile robots through UART and ZIG-100B communication terminals. The leader input is manipulated by a human operator via a game controller.

The calculations for all robots are performed concurrently and separately by the PC, simulating a distributed cooperative control system. The USB camera captures markers placed on the robots (Fig. 16).

The desired formation is the triangular form used in the simulations described in the previous section. Fig. 17 shows the trajectories of the robots as the leader moves along a circle for a certain period and then stops. It can be seen that each follower maintains their desired position. Because of the robots' static friction, some formation error remains. We believe that these errors can be removed by using adaptive mechanisms, which estimate the constant friction torques satisfying the matching conditions.

VII. CONCLUSION

In this paper, we constructed continuous and non-differentiable navigation functions for three mobile robots to enable leader-following formation control. The proposed navigation functions established the LFS property in the formation, and removed the problem of saddle points encountered by conventional methods. We performed simulations and experiments using real mobile robots in order to confirm the effectiveness of the proposed method.

Future research will consider the construction of versatile navigation functions for three or more followers.

REFERENCES

[1] H. G. Tanner, G. J. Pappas and V. Kumar, "Leader-to-Formation Stability," *IEEE Trans. Robot. Automat.*, Vol. 20, pp. 443-454, June 2004.
[2] D. Swaroop and J. K. Hedrick, "String stability of interconnected systems," *IEEE Trans. Automat. Contr.*, Vol. 41, pp. 349-357, March 1996.
[3] D. J. Stilwell and B. E. Bishop, "Platoons of underwater vehicles," *IEEE Control Syst. Mag.*, pp. 45-52, 2000.
[4] J. P. Desai, J. P. Ostrowski, and V. Kumar, "Controlling formations of multiple mobile robots," *in Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 2864-2869, 1998.
[5] D. E. Koditchek and E. Rimon, "Robot Navigation Functions on Manifolds with Boundary," *Adv. Appl. Math.*, pp. 412-442, 1990.
[6] E. D. Sontag, Y. Wang, "New characterizations of input to state stability," *IEEE Trans. Automat. Contr.*, Vol. 4, pp. 1283-1294, 1996.
[7] G. Bradski, A. Kaehler, *Learning OpenCV*, O'Reilly Media Press, 2008.
[8] Pololu Robotics & Electronics, "Pololu 3pi Robot User's Guide," http://www.pololu.com/docs/0J21.