

# Formation Control of Omnidirectional Mobile Robots using Distributed Model Predictive Control

Kiattisin Kanjanawanishkul

Department of Computer Architecture

University of Tübingen

Sand 1, Tübingen, 72076, Germany

Email: kiattisin.kanjanawanishkul@uni-tuebingen.de

**Abstract**—This paper presents experimental results of formation control problems of omnidirectional mobile robots using distributed nonlinear model predictive control (NMPC). Two main objectives are (i) to maintain a desired flexible formation pattern and (ii) to follow a reference path. Both pose errors and formation errors are included into a local objective function, which is minimized at each update time. In the formation control, the curvilinear abscissa  $s$  has been used as a coupling term with neighboring robots. The strategy in such a way that the exchange of the most recent optimal state trajectory between coupled subsystems has been employed. The distinct features of NMPC are that constraints can be explicitly accommodated, as well as nonlinear and time-varying systems can be easily handled. Experiments with three omnidirectional mobile robots are presented to illustrate the validity of our proposed method.

## I. INTRODUCTION

Formation control is a popular topic of study in multi-robot systems [1], [2]. The problem of formation control is defined as the coordination of multiple mobile robots to follow given paths (in case of given trajectories, to track time parameterized reference trajectories) and to maintain a desired spatial formation. Solutions of formation control can be applied to a wide range of applications, such as search and rescue missions and land mine removal. Various control strategies have been investigated for solving formation control problems (e.g., [3], [4]). One of them is distributed nonlinear model predictive control.

The NMPC controller is based on a finite-horizon continuous time minimization of nonlinear predicted tracking errors with constraints on the control inputs and the state variables. It predicts system outputs based on current states and the system model, finds an open-loop control profile by numerical optimization, and applies the first control signal in the optimized control profile to the system. However, due to the use of a finite horizon, control stability becomes one of the main problems. To guarantee control stability, many approaches have been investigated, e.g., using terminal region constraints and/or a terminal penalty term (see [5], [6]).

In general, the centralized implementation of formation control is not practical due to high computation requirements. Thus, our research directs at decomposing the centralized system into smaller subsystems, which are independently controlled in the NMPC framework. In [7], Jia and Krogh solve a local min-max problem for each subproblem, where

coupling comes in the dynamics and the neighboring subsystem states are treated as bounded disturbances. Linear dynamically decoupled subsystems with coupled constraints were considered in [8]. Keviczky et. al. [10] examine decoupled dynamical systems, where cost function and constraints couple the dynamical behavior of the systems. Recently, Dunbar and Murray [9] have used the exchange of the most recent optimal control trajectory between coupled subsystems prior to each update. The main difference of these control approaches is the kind of interaction between two subsystems via state variables, constraints or objectives. In this paper, we focus on multi-robot systems whose state variables and constraints are decoupled, while each robot's cost function is coupled with its neighbors. We adapted the strategies proposed in [9] to our path following problem. Each robot computes a solution to its local problem and exchanges the most recent information with its neighbors. Although very fast updates are not able to be fulfilled with our multi-robot systems, experimental results show that our formation control strategy is promising to be further investigated.

Unlike most NMPC controllers, which have been employed to solve a trajectory tracking problem, our NMPC controller is used to solve a path following problem for our omnidirectional mobile robots. Three key advantages of using NMPC in this paper are (i) integrating the velocity of a virtual vehicle,  $\dot{s}$ , into the local cost function explicitly to solve the path following problem and to avoid stringent initial condition constraints as stated in [11], (ii) controlling robot motions with input constraints, and (iii) utilizing future information of a reference path to produce an optimal predicted trajectory of a robot. This predicted trajectory is used as a coupled term in the cost function of the neighboring robots. Omnidirectional mobile robots have become increasingly popular, since they have simultaneously and independently controlled rotational and translational motion capabilities [12]. The annual RoboCup competition is an example of a highly dynamic environment where they have been exploited highly successfully.

This paper is organized in the following way: in Section II, the problem statement is explained. The path following problem and the formation control problem of omnidirectional mobile robots are modeled. The distributed NMPC approach with stability proof is presented in Section III. Section IV shows experimental results. Finally, conclusions and future

work are given in Section V.

## II. PROBLEM STATEMENT

The main problem to be solved in this paper is that we want a group of omnidirectional mobile robots to move along only one reference path and also to keep a flexible formation. The flexible formation in this sense means that when a group of robots makes a turn, an outer robot has to move faster while an inner robot has to move slower. In this case, there are two subproblems, i.e., (i) a path following problem: a robot is required to converge to a reference path, and (ii) a formation control problem: a group of robots is required to maintain a flexible formation.

### A. Formation Configuration

We consider the problem of controlling  $N$  omnidirectional mobile robots, namely  $R_i$  where  $i = 1, \dots, N$ , to follow a reference path. In this work, only one reference path with a desired forward velocity  $u_o$  is prespecified. The reference path is parameterized by the curvilinear abscissa  $s \in \mathbb{R}$ . The individual coordinate of robot  $i$  can be defined as  $[s_i, q_i]$ , where  $q_i$  is the offset distance perpendicular to  $s_i$  (see Fig. 1). Thus, we can compute the desired pose of robot  $i$  by using the following equations:

$$\mathbf{x}_{i,r} = \begin{bmatrix} x_{i,r} \\ y_{i,r} \\ \theta_{i,r} \end{bmatrix} = \begin{bmatrix} x_{i,p} - q_i \sin \theta_{i,p} \\ y_{i,p} + q_i \cos \theta_{i,p} \\ \theta_{i,p} \end{bmatrix}, \quad (1)$$

where  $[x_{i,p}, y_{i,p}, \theta_{i,p}]$  is the state vector at  $s_i$  and  $[x_{i,r}, y_{i,r}, \theta_{i,r}]$  is the vector of the reference pose at  $s_i$  with offset  $q_i$  (see Fig. 2). Also, we can calculate the desired velocities by

$$u_{i,r} = u_o(1 - k_{i,p}(s)q_i), \quad \omega_{i,r} = k_i(l)u_{i,r}, \quad (2)$$

where

$$k_i(l) = \frac{k_{i,p}(s)}{1 - q_i k_{i,p}(s)}, \quad (3)$$

$u_{i,r}$  and  $\omega_{i,r}$  are the desired translational and rotational velocities of robot  $i$ , respectively.  $k_i(l)$  and  $k_{i,p}(s)$  are the curvature of the robot's path and of the reference path, respectively. We locate a virtual vehicle at  $s_i$  and  $\dot{s}_i$  is defined as the velocity of a virtual vehicle moving along the reference path. However, in our case, robot  $i$  will follow this virtual vehicle with offset  $q_i$ . Then we have the relationship between the velocity of the virtual vehicle and the velocity of the robot's path as follows

$$\dot{l}_i = \dot{s}_i(1 - k_{i,p}(s)q_i). \quad (4)$$

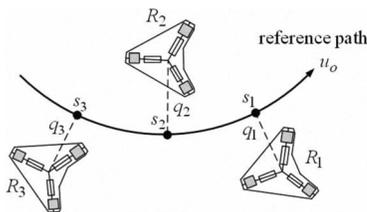


Fig. 1. Example of a formation following a given reference path.

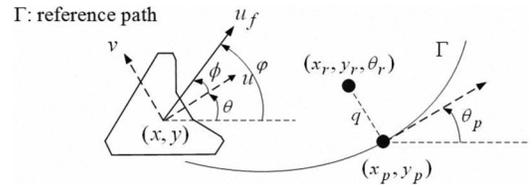


Fig. 2. Illustration of the path following problem.

It has to be noted that (1), (2) and (4) are not valid if  $q_i$  is not constant. Also, cusp or singularity at the robot's path is not considered in this work.

The formation graph is defined as  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , where  $\mathcal{V} = 1, \dots, N$  is the set of robots and  $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$  is the set of relative vectors between robots. Two robots  $i$  and  $j$  are called neighbors if  $(i, j) \in \mathcal{E}$ , and the set of neighbors of robot  $i$  is denoted by  $\mathcal{N}_i \in \mathcal{V}$ . All graphs considered in this paper are undirected and we assume that the undirected  $\mathcal{G}$  is connected. The desired distance between two neighbors  $i$  and  $j$  can be determined by  $p_{ij}(s)$ , where  $s_i + p_{ij} = s_j$  and  $p_{ij} = -p_{ji}$ . The formation error vector  $E \in \mathbb{R}^M$ , where  $M$  is the number of edges, has components  $e_i \in \mathbb{R}$  defined as  $e_i = s_i - s_j + p_{ij}$ . Let  $E_i$  denote the vector of all components of  $E$ , which have a coupled term with robot  $i$ . Then we have  $E = Cs + \mathbf{p}$ , where the vector  $\mathbf{p} = [\dots, p_{ij}, \dots]^T$ ,  $s = [\dots, s_i, \dots]^T$ , and the matrix  $C \in \mathbb{R}^{N \times M}$  is the incidence matrix [13].

### B. Path Following Problem

The kinematic model of an omnidirectional mobile robot (see Fig. 3) can be given by (we drop subscript  $i$  for convenience in this subsection)

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ \omega \end{bmatrix}, \quad (5)$$

where  $[x, y, \theta]^T$  is the state vector.  $u$  and  $v$  are the translational velocities and  $\omega$  is the rotational velocity. When the wheel velocities are considered, the lower level kinematic model can be described by

$$\begin{bmatrix} \nu_1 \\ \nu_2 \\ \nu_3 \end{bmatrix} = \begin{bmatrix} \cos \alpha & \sin \alpha & L_w \\ -\cos \alpha & \sin \alpha & L_w \\ 0 & -1 & L_w \end{bmatrix} \begin{bmatrix} u \\ v \\ \omega \end{bmatrix}, \quad (6)$$

where  $\nu = [\nu_1, \nu_2, \nu_3]^T$  is the vector of wheel velocities,  $L_w$  is the distance from the wheel to the robot's center, and  $\alpha$

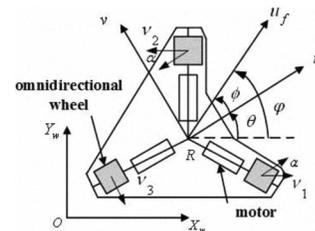


Fig. 3. Coordinate frames of an omnidirectional mobile robot.

refers to the wheel orientation in the body frame. The wheel velocities are subject to constraints:  $\nu_{\min} \leq \nu \leq \nu_{\max}$ .

This kinematic model can be formulated with respect to a Serret-Frenet frame moving along the reference path. This frame plays the role of the body frame of a virtual vehicle. Robot  $i$  will follow this virtual vehicle with offset  $q$ . Given the error state  $\mathbf{x}_e$  between the robot state vector and the reference state vector, and the kinematic model (5), the error state dynamic model expressed in the Serret-Frenet frame is derived as follows

$$\begin{aligned} \dot{x}_e &= y_e k(l) \dot{l} - \dot{l} + u_f \cos \varphi_e \\ \dot{y}_e &= -x_e k(l) \dot{l} + u_f \sin \varphi_e, \\ \dot{\varphi}_e &= \dot{\varphi} - k(l) \dot{l} \end{aligned} \quad (7)$$

where  $u_f = \sqrt{u^2 + v^2}$  is the forward speed, and  $\phi = \arctan \frac{v}{u} = \varphi - \theta$  is the angle of the moving direction in the body frame.

Since translation and rotation of omnidirectional mobile robots can be controlled separately [12], we can drive the robot orientation  $\theta$  to the desired orientation at the same time. In this paper, the desired orientation is the angle of the tangent direction to the reference path  $\theta_r$ . The orientation error is defined as  $\theta_e = \theta - \theta_r$ , then we have  $\dot{\theta}_e = \omega - \omega_r$ , where  $\omega = \dot{\theta}$  and  $\omega_r = \dot{\theta}_r$ .

In the formation control problem, the forward speed of each robot has to be coordinated with its neighbors. Thus, we introduce a new state variable  $\eta$ , given by  $\eta = u_f - u_r$ . Then we have  $\dot{\eta} = a - \dot{u}_r$ , where  $a = \dot{u}_f$ , as a new input variable. We define the control inputs of each robot as:

$$\mathbf{u}_e = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} -\dot{l} + (\eta + u_r) \cos \phi_e \\ \dot{\varphi} - k(l) \dot{l} \\ \omega - \omega_r \\ a - \dot{u}_r \end{bmatrix}. \quad (8)$$

Finally, the error state kinematic model becomes

$$\dot{\mathbf{x}}_e = \begin{bmatrix} \dot{x}_e \\ \dot{y}_e \\ \dot{\varphi}_e \\ \dot{\theta}_e \\ \dot{\eta} \end{bmatrix} = \begin{bmatrix} y_e k_i(l) \dot{l} + u_1 \\ -x_e k(l) \dot{l} + (\eta + u_r) \sin \varphi_e \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}. \quad (9)$$

### III. DISTRIBUTED NMPC APPROACH

The distributed cost function for each robot is defined as

$$\begin{aligned} L_i(\mathbf{x}_{i,e}, E_i, \mathbf{u}_{i,e}) &= \mathbf{x}_{i,e}^T Q_i \mathbf{x}_{i,e} + \mathbf{u}_{i,e}^T R_i \mathbf{u}_{i,e} \\ &+ \sum_{(i,j) \in \mathcal{E}} (W(s_i - s_j + p_{ij})^2), \end{aligned} \quad (10)$$

where the term  $(s_i - s_j + p_{ij})$  couples the states of neighboring robots and the deviation from the desired values is weighted by the positive definite matrices  $Q_i$  and  $R_i$ , and the positive constant  $W$ . Since each cost function depends on the neighbors' trajectories, each robot has to exchange an assumed trajectory with its neighbors at each update. Based on notations given in [9], over any prediction interval  $\tau \in [t_k, t_k + T_p]$ ,  $k \in \mathbb{N}$ , associated with current time  $t_k$ , for each robot we denote

- $s_i^p(\tau; t_k)$ : the predicted trajectory,
- $s_i^*(\tau; t_k)$ : the optimal predicted trajectory,
- $\hat{s}_i(\tau; t_k)$ : the assumed trajectory.

The corresponding robot state and control trajectories are denoted by  $\mathbf{x}_{i,e}^p(\tau; t_k)$ ,  $\mathbf{x}_{i,e}^*(\tau; t_k)$ ,  $\hat{\mathbf{x}}_{i,e}(\tau; t_k)$  and  $\mathbf{u}_{i,e}^p(\tau; t_k)$ ,  $\mathbf{u}_{i,e}^*(\tau; t_k)$ ,  $\hat{\mathbf{u}}_{i,e}(\tau; t_k)$ , respectively. We concatenate the states and inputs into vectors as  $\mathbf{x}_e = (\mathbf{x}_{1,e}, \dots, \mathbf{x}_{N,e})$  and  $\mathbf{u}_e = (\mathbf{u}_{1,e}, \dots, \mathbf{u}_{N,e})$ , respectively. The problem is to find

$$J^*(\mathbf{x}_{i,e}(t_k), E_i(t_k)) = \min_{\mathbf{u}_{i,e}^p} J_i(\mathbf{x}_{i,e}(t_k), E_i(t_k), \mathbf{u}_{i,e}^p(\cdot; t_k)), \quad (11)$$

where  $J_i(\mathbf{x}_{i,e}(t_k), E_i(t_k), \mathbf{u}_{i,e}^p(\cdot; t_k)) = V_i(\mathbf{x}_{i,e}^p(t_k + T_p; t_k)) + \int_{t_k}^{t_k + T_p} L_i(\mathbf{x}_{i,e}^p(\gamma; t_k), \hat{E}_i(\gamma; t_k), \mathbf{u}_{i,e}^p(\gamma; t_k)) d\gamma$

subject to:  $\dot{\mathbf{x}}_{i,e}^p(\tau; t_k) = \mathbf{f}_i(\mathbf{x}_{i,e}^p(\tau; t_k), \mathbf{u}_{i,e}^p(\tau; t_k))$

$$\dot{\hat{\mathbf{x}}}_{i,e}(\tau; t_k) = \mathbf{f}_i(\hat{\mathbf{x}}_{i,e}(\tau; t_k), \hat{\mathbf{u}}_{i,e}(\tau; t_k))$$

$$\mathbf{u}_{i,e}^p(\tau; t_k) \in \mathcal{U}_i \quad \forall \tau \in [t_k, t_k + T_c]$$

$$\mathbf{x}_{i,e}^p(\tau; t_k) \in \mathcal{X}_i \quad \forall \tau \in [t_k, t_k + T_p]$$

$$\hat{\mathbf{x}}_{i,e}(\tau; t_k) \in \mathcal{X}_i \quad \forall \tau \in [t_k, t_k + T_p]$$

$$\mathbf{x}_{i,e}^p(t_k + T_p; t_k) \in \Omega_i$$

$$\hat{\mathbf{x}}_{i,e}(t_k; t_k) = \mathbf{x}_{i,e}^p(t_k; t_k) = \mathbf{x}_{i,e}(t_k)$$

$$|s_i^p(\tau; t_k) - \hat{s}_i(\tau; t_k)| \leq \delta^2 \kappa \quad \forall \tau \in [t_k, t_k + T_p]$$

$V_i(\mathbf{x}_{i,e}^p(t_k + T_p))$  is the decoupled terminal penalty and  $\Omega_i$  is the terminal region of robot  $i$ ,  $\delta$  is the sampling time, and  $\kappa$  is a constant.  $T_c$  and  $T_p$  are the control horizon and the prediction horizon, respectively, with  $T_c \leq T_p$ .  $\mathcal{X}_i \subseteq \mathbb{R}^n$  and  $\mathcal{U}_i \subseteq \mathbb{R}^m$  denote the set of feasible  $n$  dimensional states and  $m$  dimensional inputs of robot  $i$ , respectively.

As proposed in [9], the *compatibility constraint* has been employed. The optimized trajectory  $s_i^p$  for robot  $i$  is constrained to be at most a distance of  $\delta^2 \kappa$  from the assumed trajectory  $\hat{s}_i$ . The constraint is a means of enforcing a degree of consistency between what a robot plans to do and what neighbors believe that robot will plan to do. Before giving any analysis, the following assumptions need to be made:

**Assumption 1.** Let  $X_\Sigma \subset \mathbb{R}^{(n+1)N}$  denote the set of initial states  $(\mathbf{x}_e(t), s(t_0))$ , which can be steered to  $\Omega$  by  $\mathbf{u}_e^p(\tau; t) \in \mathcal{U}$ ,  $\tau \in [t_0, t_0 + T_p]$ .

When Assumption 1 holds, the problem is feasible at initialization. The initial feasibility of the implementation implies subsequent feasibility, following the standard arguments in [5], [6], [9] by induction. Due to space limitations, we do not repeat the proof in this paper.

**Assumption 2.** There exists a constant  $\rho_{\max} \in (0, \infty)$  such that  $\|(\mathbf{x}_e^*(t; t_k), s^*(t; t_k)) - (\mathbf{x}_e^c, s^c)\| \leq \rho_{\max}$  and  $\|(\hat{\mathbf{x}}_e(t; t_k), \hat{s}(t; t_k)) - (\mathbf{x}_e^c, s^c)\| \leq \rho_{\max}$ ,  $\forall t \in [t_k, t_k + T_p]$ , where  $(\mathbf{x}_e^c, s^c)$  is the desired equilibrium state.

The symbol  $\|\cdot\|$  denotes any vector norm in  $\mathbb{R}^n$ , and dimension  $n$  follows from the context. For any vector  $x \in \mathbb{R}^n$ ,  $\|x\|_P$  denotes the  $P$ -weighted 2-norm, defined by  $\|x\|_P^2 = x^T P x$ , and  $P$  is any positive-definite real symmetric matrix.

When Assumption 2 holds, the optimal and assumed state trajectories remain bounded. Also, let  $u_{\max}$  be the positive

scalar constant  $u_{\max} = \{\max \|v(t)\| \mid v(t) \in \mathcal{U}^N\}$ . Then the control algorithm can be given by

**Algorithm 1.** The distributed model predictive controller for any robot is given as follows:

- 1) Over any interval  $[t_k, t_{k+1})$ ,  $k \in \mathbb{N}$ :
  - a) Apply  $\mathbf{u}_{i,e}^*(\tau, t_k)$ ,  $\tau \in [t_k, t_{k+1})$ .
  - b) Compute  $\hat{s}_i(\tau; t_{k+1}) = \hat{s}_i(\tau)$  as

$$\hat{s}_i(\tau) = \begin{cases} s_i^*(\tau; t_k), & \tau \in [t_{k+1}, t_k + T_p] \\ s_{i,eT}(\tau), & \tau \in [t_k + T_p, t_{k+1} + T_p] \end{cases}$$

where  $s_{i,eT}(\tau)$  is the solution from the terminal feedback controllers.

- c) Transmit  $\hat{s}_i(\cdot; t_{k+1})$  to every neighbor and receive  $\hat{s}_j(\tau; t_{k+1})$  from every neighbor  $j$ .

- 2) At any time  $t_k$ ,  $k \in \mathbb{N}$ :

- a) Measure current state  $\mathbf{x}_{i,e}(t_k)$ .
- b) Solve (11), yielding  $\mathbf{u}_{i,e}^*(\tau, t_k)$ ,  $\tau \in [t_k, t_k + T_p)$ .

We note that  $Q_\Sigma$  and  $W_\Sigma$ , positive definite and symmetric, can be defined as follows:  $Q_\Sigma = \text{diag}(Q_1, \dots, Q_N)$ ,  $W_\Sigma = [WC^T C]$ , respectively.  $\lambda_{\min}(Q_\Sigma)$  and  $\lambda_{\max}(Q_\Sigma)$  denote the smallest and the largest eigenvalues of  $Q_\Sigma$ , respectively. Since we assume that our formation graph  $\mathcal{G}$  is connected, the second smallest eigenvalue  $\lambda_2$  of  $W_\Sigma$  (a weighted Laplacian of graph  $\mathcal{G}$ ) is positive. The following characteristics of eigenvalues hold (see [13])

$$\lambda_2 = \min \frac{\mathbf{x}^T W_\Sigma \mathbf{x}}{\mathbf{x}^T \mathbf{x}}, \quad \lambda_{\max} = \max \frac{\mathbf{x}^T W_\Sigma \mathbf{x}}{\mathbf{x}^T \mathbf{x}},$$

for any vector  $\mathbf{x}$  and  $\mathbf{x}^T W_\Sigma \mathbf{x} = \sum_{(i,j) \in \mathcal{E}} W(x_i - x_j)^2$ .

Now we find the terminal penalty and terminal region, and then we will give the stability analysis of the distributed NMPC approach based on the idea proposed in [9].

### A. Decoupled Terminal Controllers

One elementary question in NMPC is whether a finite horizon NMPC strategy guarantees closed-loop stability. Many strategies have been proposed to solve this problem, e.g., the quasi-infinite horizon approach [5], the dual mode approach [6], a global control Lyapunov function (CLF) approach [14]. In this paper, a Lyapunov function for the decoupled terminal-state penalty is defined as follows:

$$V(\mathbf{x}_e(t_k + T_p)) = \frac{1}{2} \mathbf{x}_e(t_k + T_p)^T P \mathbf{x}_e(t_k + T_p), \quad (12)$$

where  $P = \text{diag}(P_1, \dots, P_N)$  is a positive definite matrix. Under the terminal-state controller  $\mathbf{u}_e^L(t)$ , the following condition is satisfied:

$$\dot{V}(\mathbf{x}_e(t)) + L(t, \mathbf{x}_e(t), \mathbf{u}_e(t)) \leq 0, \quad (13)$$

for any state  $\mathbf{x}_e(t)$  belonging to the terminal region  $\Omega$ . Then, stability condition for each subsystem  $i$  becomes (from here

in this subsection, we drop subscript  $i$  for convenience)

$$\begin{aligned} & \dot{V}(\mathbf{x}_e(t)) + L(t, \mathbf{x}_e(t), \mathbf{u}_e(t)) \\ &= p_{11} x_{eT} \dot{x}_{eT} + p_{22} y_{eT} \dot{y}_{eT} + p_{33} \varphi_{eT} \dot{\varphi}_{eT} + p_{44} \theta_{eT} \dot{\theta}_{eT} \\ & \quad + p_{55} \eta_{eT} \dot{\eta}_{eT} + L(t, \mathbf{x}_e(t), \mathbf{u}_e(t)) \\ &= p_{11} x_{eT} u_1^L + p_{22} y_{eT} (\eta_{eT} + u_r) \sin \varphi_{eT} + p_{33} \varphi_{eT} u_2^L \\ & \quad + p_{44} \theta_{eT} u_3^L + p_{55} \eta_{eT} u_4^L + q_{11} x_{eT}^2 + q_{22} y_{eT}^2 + q_{33} \varphi_{eT}^2 \\ & \quad + q_{44} \theta_{eT}^2 + q_{55} \eta_{eT}^2 + r_{11} u_1^{L^2} + r_{22} u_2^{L^2} + r_{33} u_3^{L^2} \\ & \quad + r_{44} u_4^{L^2} + w_T (s_{eT} - s^c)^2, \end{aligned}$$

where  $w_T = \lambda_{\max}(W_\Sigma)$ . By construction,  $\lambda_{\max}(W_\Sigma) \mathbf{I}_{n \times n} \geq W_\Sigma$ . The terminal state feedback controller  $\mathbf{u}_e^L = [u_1^L, u_2^L, u_3^L, u_4^L]^T$  can be selected as follows:  $u_1^L = -k_1 x_{eT}$ ,  $u_2^L = -k_2 \varphi_{eT}$ ,  $u_3^L = -k_3 \theta_{eT}$ ,  $u_4^L = -k_4 \eta_{eT}$ , where  $k_1, k_2, k_3, k_4 \geq 0$ . The stability condition is changed to

$$\begin{aligned} & \dot{V}(\mathbf{x}_e(t)) + L(t, \mathbf{x}_e(t), \mathbf{u}_e(t)) \\ &= x_{eT}^2 (-p_{11} k_1 + q_{11} + k_1^2 r_{11}) + y_{eT} p_{22} (\eta_{eT} + u_r) \sin \varphi_{eT} \\ & \quad + q_{22} y_{eT}^2 + \varphi_{eT}^2 (-p_{33} k_2 + q_{33} + k_2^2 r_{22}) \\ & \quad + \theta_{eT}^2 (-p_{44} k_3 + q_{44} + k_3^2 r_{33}) \\ & \quad + \eta_{eT}^2 (-p_{55} k_4 + q_{55} + k_4^2 r_{44}) + w_T (s_{eT} - s^c)^2. \end{aligned}$$

All weight parameters have to be selected such that (13) is satisfied. To have a negative derivative of the value function, the following requirement for the weight parameters is required:

$$\begin{aligned} -p_{11} k_1 + q_{11} + k_1^2 r_{11} &\leq 0, & -p_{33} k_2 + q_{33} + k_2^2 r_{22} &\leq 0, \\ -p_{44} k_3 + q_{44} + k_3^2 r_{33} &\leq 0, & -p_{55} k_4 + q_{55} + k_4^2 r_{44} &\leq 0, \end{aligned} \quad (14)$$

and the terminal-state region is defined as follows:

$$w_T (s_{eT} - s^c)^2 + y_{eT} p_{22} (\eta_{eT} + u_r) \sin \varphi_{eT} + q_{22} y_{eT}^2 < 0. \quad (15)$$

### B. Stability Analysis

The main idea based on [9] is to show that by applying Algorithm 1, the closed-loop state  $(\mathbf{x}_e, s)$  converges to a neighborhood of the equilibrium state, with a sufficiently small upper bound on the update period  $\delta_{\max}$ . At any time  $t_k$ , the sum of the optimal distributed value functions is denoted

$$J_\Sigma^*(\mathbf{x}_e(t_k), s(t_k)) = \sum_{i=1}^N J_i^*(\mathbf{x}_{i,e}(t_k), E_i(t_k)). \quad (16)$$

The following lemma gives a bounding result on the decrease in  $J_\Sigma^*(\cdot)$  from one update to the next. The *compatibility constraints* are applied at each update times  $t_k$  with  $k \geq 1$ , thus the result holds for  $k \in \{1, 2, \dots\}$ .

**Lemma 1.** Suppose Assumptions 1 and 2 hold and  $(\mathbf{x}_e(t_0), s(t_0)) \in X_\Sigma$ . Then, by the implementation of Algorithm 1 with the constant  $\xi$  defined by

$$\xi = W \kappa (4\rho_{\max}) 2|\mathcal{E}| T_p, \quad (17)$$

the function  $J_\Sigma^*(\cdot)$  satisfies

$$\begin{aligned} & J_\Sigma^*(\mathbf{x}_e(t_k + \delta), s(t_k + \delta)) - J_\Sigma^*(\mathbf{x}_e(t_k), s(t_k)) \leq \\ & - \int_{t_k}^{t_k + \delta} \sum_{i=1}^N L_i(\mathbf{x}_{i,e}^*(\gamma, t_k), \hat{E}_i(\gamma, t_k)) d\gamma + \delta^2 \xi. \end{aligned}$$

*Proof:* For any  $k \geq 1$ ,

$$J_{\Sigma}^*(\mathbf{x}_e(t_k), s(t_k)) = V(\mathbf{x}_e^*(t_k + T_p; t_k)) + \int_{t_k}^{t_k + T_p} \sum_{i=1}^N L_i(\mathbf{x}_{i,e}^*(\gamma; t_k), \hat{E}_i(\gamma; t_k), \mathbf{u}_{i,e}^*(\gamma; t_k)) d\gamma.$$

The optimal control is applied for  $\delta \in (0, T_p]$  seconds, then at time  $t_{k+1} = t_k + \delta$ , we get new state update  $\mathbf{x}_{i,e}$  and  $s_i$ . A feasible control for (11) at time  $t_{k+1}$  is  $\mathbf{u}_e^*(\cdot; t_{k+1}) = \hat{\mathbf{u}}_e(\cdot; t_{k+1})$ ; therefore,

$$\begin{aligned} J_{\Sigma}^*(\mathbf{x}_e(t_{k+1}), s(t_{k+1})) &\leq V(\hat{\mathbf{x}}_e(t_{k+1} + T_p; t_{k+1})) \\ &+ \int_{t_{k+1}}^{t_{k+1} + T_p} L(\hat{\mathbf{x}}_e(\gamma; t_{k+1}), \hat{E}(\gamma; t_{k+1}), \hat{\mathbf{u}}_e(\gamma; t_{k+1})) d\gamma \\ J_{\Sigma}^*(\mathbf{x}_e(t_{k+1}), s(t_{k+1})) - J_{\Sigma}^*(\mathbf{x}_e(t_k), s(t_k)) &\leq \\ &- \int_{t_k}^{t_{k+1}} \sum_{i=1}^N L_i(\mathbf{x}_{i,e}^*(\gamma; t_k), \hat{E}_i(\gamma; t_k), \mathbf{u}_{i,e}^*(\gamma; t_k)) d\gamma \\ &+ \int_{t_{k+1}}^{t_k + T_p} \sum_{i=1}^N L_i(\hat{\mathbf{x}}_{i,e}(\gamma; t_{k+1}), \hat{E}_i(\gamma; t_{k+1}), \hat{\mathbf{u}}_{i,e}(\gamma; t_{k+1})) d\gamma \\ &- \int_{t_{k+1}}^{t_k + T_p} \sum_{i=1}^N L_i(\mathbf{x}_{i,e}^*(\gamma; t_k), \hat{E}_i(\gamma; t_k), \mathbf{u}_{i,e}^*(\gamma; t_k)) d\gamma \\ &+ \int_{t_k + T_p}^{t_{k+1} + T_p} \sum_{i=1}^N L_i(\hat{\mathbf{x}}_{i,e}(\gamma; t_{k+1}), \hat{E}_i(\gamma; t_{k+1}), \hat{\mathbf{u}}_{i,e}(\gamma; t_{k+1})) \\ &d\gamma + V(\hat{\mathbf{x}}_e(t_{k+1} + T_p; t_{k+1})) - V(\mathbf{x}_e^*(t_k + T_p; t_k)). \end{aligned}$$

Because of  $\hat{\mathbf{x}}_e(t_k + T_p; t_{k+1}) = \mathbf{x}_e^*(t_k + T_p; t_k)$ ,  $\hat{\mathbf{x}}_e(\tau; t_{k+1})$  obtained by the terminal feedback controllers for  $\tau \in [t_k + T_p, t_{k+1} + T_p]$ , (12) and (13), the sum of the last three terms in the inequality above is nonpositive, and thus the inequality holds after removing these three terms. Because of  $L_i(\mathbf{x}_{i,e}^*(\gamma; t_k), \hat{E}_i(\gamma; t_k), \mathbf{u}_{i,e}^*(\gamma; t_k)) \geq L_i(\mathbf{x}_{i,e}^*(\gamma; t_k), \hat{E}_i(\gamma; t_k))$ , the lemma has been proven if we can prove that

$$\int_{t_{k+1}}^{t_k + T_p} \sum_{i=1}^N \{L_i(\hat{\mathbf{x}}_{i,e}(\gamma; t_{k+1}), \hat{E}_i(\gamma; t_{k+1}), \hat{\mathbf{u}}_{i,e}(\gamma; t_{k+1})) - L_i(\mathbf{x}_{i,e}^*(\gamma; t_k), \hat{E}_i(\gamma; t_k), \mathbf{u}_{i,e}^*(\gamma; t_k))\} d\gamma \leq \delta^2 \xi.$$

Because of  $\hat{\mathbf{x}}_{i,e}(\gamma; t_{k+1}) = \mathbf{x}_{i,e}^*(\gamma; t_k)$  and  $\hat{\mathbf{u}}_{i,e}(\gamma; t_{k+1}) = \mathbf{u}_{i,e}^*(\gamma; t_k)$ , for  $\gamma \in [t_{k+1}, t_k + T_p]$ , the integrand above is equal to

$$= \sum_{i=1}^N \sum_{j \in N_i} W_i \{ (s_i^*(\gamma; t_k) - s_j^*(\gamma; t_k) + p_{ij})^2 - (s_i^*(\gamma; t_k) - \hat{s}_j(\gamma; t_k) + p_{ij})^2 \}$$

Using some algebraic calculation, we have

$$\begin{aligned} &(s_i^*(\gamma; t_k) - s_j^*(\gamma; t_k) + p_{ij})^2 - (s_i^*(\gamma; t_k) - \hat{s}_j(\gamma; t_k) + p_{ij})^2 \\ &= (s_j^*(\gamma; t_k) - \hat{s}_j(\gamma; t_k))(-2(s_i^*(\gamma; t_k) + p_{ij}) \\ &\quad + s_j^*(\gamma; t_k) + \hat{s}_j(\gamma; t_k)) \\ &\leq \delta^2 \kappa (4\rho_{\max}), \end{aligned}$$

where we use  $p_{ij} = s_j^c - s_i^c$ , Assumption 2, and the compatibility constraint. Then we have

$$\delta^2 \kappa W \int_{t_{k+1}}^{t_k + T_p} \sum_{i=1}^N \sum_{j \in N_i} (4\rho_{\max}) d\gamma \leq \delta^2 \xi$$

with the total number of pairwise neighbors  $|\mathcal{E}| = \sum_{i=1}^N \sum_{j \in N_i} (1/2)$ . This completes the proof. ■

Now the result has been bounded by using Lemma 1. Then, we have to show that the closed-loop state trajectory converges to a closed neighborhood of the objective state. The neighborhood of convergence is a level set of the function  $J_{\Sigma}^*(\mathbf{x}_e(t), s(t))$ . We firstly define  $z_i(t) = (\mathbf{x}_{i,e}(t), s_i(t))$  and we then have  $\|z - z^c\|_G^2$ , where

$$G = \begin{bmatrix} Q_{\Sigma} & 0 \\ 0 & W_{\Sigma} \end{bmatrix}, \quad (18)$$

and  $z^c = (\mathbf{x}_e^c, s^c)$  is the desired equilibrium state.

We define  $\Omega_{\beta} = \{z \in \mathbb{R}^{(n+1)N} | J_{\Sigma}^*(\mathbf{x}_e(t), s(t)) \leq \beta\}$  with constant  $\beta \in (0, \infty)$  as the compact level set. The set  $\Omega_{\beta}$  is in the interior of  $X_{\Sigma}$  if  $\beta > 0$  is sufficiently small. We can choose a constant  $r = r(\beta) \in (0, \rho_{\max})$  with the following properties:

$$B(z^c; r) \subseteq \Omega_{\beta/2} \quad \text{and} \quad r^2 \leq \frac{8\beta}{\lambda_{\min}(Q_{\Sigma})}, \quad (19)$$

where  $B(z^c; r)$  denotes a closed ball in  $\mathbb{R}^{(n+1)N}$  with center  $z^c$  and radius  $r$ . We require the following assumptions (see [9]).

**Assumption 3.** The following holds: (a) the update period is sufficiently small that the following first-order Taylor series approximation is valid:

$$\begin{aligned} \sum_{i=1}^N L_i(\mathbf{x}_{i,e}^*(\gamma; t_k), \hat{E}_i(\gamma; t_k)) &\approx \|z(t_k) - z^c\|_G^2 \\ &+ 2(\gamma - t_k)(z(t_k) - z^c)^T G f(z(t_k), \mathbf{u}_e^*(t_k; t_k)), \end{aligned}$$

for all  $\gamma \in [t_k, t_k + \delta]$  and any  $k \in \mathbb{N}$ ; (b) there exists a Lipschitz constant  $\mathcal{K} \in [1, \infty)$  such that for any  $z, z' \in X_{\Sigma}$ ,  $u, u' \in \mathcal{U}^N$ ,

$$\|f(z, u) - f(z', u')\| \leq \mathcal{K}(\|z - z'\| + \|u - u'\|).$$

**Assumption 4.** The following holds:

$$\lambda_{\min}(Q_{\Sigma}) \leq \lambda_2(W_{\Sigma}), \quad \lambda_{\max}(Q_{\Sigma}) \leq \lambda_{\max}(W_{\Sigma}).$$

Then the theorem based on [9] can be now stated.

**Theorem 1.** Suppose Assumptions 1-4 hold,  $z(t_0) \in X_{\Sigma}$  and for a given constant  $\beta \in (0, \infty)$  with  $\Omega_{\beta} \subset X_{\Sigma}$ , the constant  $r = r(\beta) \in (0, \rho_{\max})$  is such that the properties in (19) are satisfied. Then by implementation of Algorithm 1 with

$$\delta_{\max} = \frac{(r/2)^2 \lambda_{\min}(Q_{\Sigma})}{\xi + \mathcal{K} \rho_{\max} (\rho_{\max} + u_{e, \max}) \lambda_{\max}(W_{\Sigma})}, \quad (20)$$

and  $\xi$  given by (17), the closed-loop state trajectory enters  $B(z^c; r)$  in finite time and remains in  $\Omega_{\beta}$  for all future time.

*Proof:* Substituting the Taylor series expressions to:

$$\begin{aligned} &J_{\Sigma}^*(\mathbf{x}_e(\tau), s(\tau)) - J_{\Sigma}^*(\mathbf{x}_e(t_k), s(t_k)) \leq \\ &- \int_{t_k}^{\tau} \sum_{i=1}^N L_i(\mathbf{x}_{i,e}(\gamma; t_k), \hat{E}_i(\gamma; t_k)) d\gamma + \delta^2 \xi, \end{aligned}$$

$\forall \tau \in (t_k, t_k + \delta]$ , for any constant  $\delta \in (0, \delta_{\max}]$ , we have

$$\begin{aligned} &J_{\Sigma}^*(\mathbf{x}_e(\tau), s(\tau)) - J_{\Sigma}^*(\mathbf{x}_e(t_k), s(t_k)) \leq \\ &-(\tau - t_k) \|z(t_k) - z^c\|_G^2 + (\tau - t_k)^2 H \end{aligned}$$

where  $H = -(z(t_k) - z^c)^T G f(z(t_k), \mathbf{u}_e^*(t_k; t_k))$  has the upper bound  $H \leq \|z(t_k) - z^c\| \|f(z(t_k), \mathbf{u}_e^*(t_k; t_k))\| \lambda_{\max}(W_\Sigma) \leq \rho_{\max} \mathcal{K}(\rho_{\max} + u_{\max}) \lambda_{\max}(W_\Sigma)$ .

Because of  $\tau - t_k \leq \delta \leq \delta_{\max}$ , we have

$$\begin{aligned} & J_\Sigma^*(\mathbf{x}_e(\tau), s(\tau)) - J_\Sigma^*(\mathbf{x}_e(t_k), s(t_k)) \\ & \leq -(\tau - t_k) \|z(t_k) - z^c\|_G^2 + \delta \delta_{\max} (H + \xi) \\ & \leq -(\tau - t_k) \lambda_{\min}(Q_\Sigma) \|z(t_k) - z^c\|^2 + \delta \delta_{\max} (H + \xi) \\ & \leq -\lambda_{\min}(Q_\Sigma) \{(\tau - t_k) \|z(t_k) - z^c\|^2 - \delta(r/2)^2\} \\ & \leq -\delta \lambda_{\min}(Q_\Sigma) \{\|z(t_k) - z^c\|^2 - (r/2)^2\}, \end{aligned}$$

with  $\tau = t_k + \delta = t_{k+1}$ . From this inequality, there exists a finite integer  $l \geq 1$  such that  $z(t_l) \in B(z^c; r)$ . If this were not the case, the inequality implies  $J_\Sigma^*(\mathbf{x}_e(t_k), s(t_k)) \rightarrow -\infty$  as  $k \rightarrow \infty$ . Since the cost functions are nonnegative,  $J_\Sigma^*(\mathbf{x}_e(t_k), s(t_k)) \geq 0$  for any  $z(t_k) \in X_\Sigma$ . Therefore, by contradiction, there exists a finite integer  $l \geq 1$  such that  $z(t_l) \in B(z^c; r) \subseteq \Omega_{\beta/2}$ , verifying the first statement of the theorem. Now there are two cases to be proved that  $(\mathbf{x}_e(t), s(t)) \in \Omega_\beta$  for all time  $t \geq t_l$ . First, if  $(z(t_k)) \in \Omega_{\beta/2} \setminus B(z^c; r/2)$ , then  $z(t) \in \Omega_\beta$  for all time  $t \in [t_k, t_{k+1}]$  and  $z(t_{k+1}) \in \Omega_{\beta/2}$ . This is shown first that the upper bound becomes

$$J_\Sigma^*(\mathbf{x}_e(\tau), s(\tau)) - J_\Sigma^*(\mathbf{x}_e(t_k), s(t_k)) \leq \delta_{\max} \lambda_{\min}(Q_\Sigma) (r/2)^2$$

for all  $\tau \in (t_k, t_{k+1}]$ . Also,  $\delta_{\max} < 1/4$ , since

$$\delta_{\max} < \frac{(r/2)^2 \lambda_{\min}(Q_\Sigma)}{\rho_{\max} \mathcal{K}(\rho_{\max} + u_{\max}) \lambda_{\max}(W_\Sigma)} \leq \frac{(r/2)^2}{\rho_{\max}^2}.$$

Therefore, the bound on  $J_\Sigma^*$  becomes

$$J_\Sigma^*(\mathbf{x}_e(\tau), s(\tau)) \leq J_\Sigma^*(\mathbf{x}_e(t_k), s(t_k)) + \frac{\lambda_{\min}(Q_\Sigma)(r/2)^2}{4} \leq \beta$$

for all  $\tau \in (t_k, t_{k+1}]$ , using  $J_\Sigma^*(\mathbf{x}_e(t_k), s(t_k)) \leq \beta/2$  and (19). Likewise,  $(z(t_k)) \in \Omega_{\beta/2} \setminus B(z^c; r/2)$  and (19) imply that  $J_\Sigma^*(\mathbf{x}_e(t_{k+1}), s(t_{k+1})) < J_\Sigma^*(\mathbf{x}_e(t_k), s(t_k))$  and so  $z(t_{k+1}) \in \Omega_{\beta/2}$ . In the second case, if  $z(t_k) \in B(z^c; r/2)$ , then  $z(t) \in B(z^c; r) \subseteq \Omega_{\beta/2}$  for all time  $t \in [t_k, t_{k+1}]$ . From the bounding argument, we have

$$\begin{aligned} \|z(t) - z^c\| & \leq \|z(t_k) - z^c\| + \left\| \int_{t_k}^t (f(z(\gamma), \mathbf{u}_e(\gamma))) d\gamma \right\| \\ & \leq r/2 + (t - t_k) \mathcal{K}(\rho_{\max} + u_{\max}) \\ & \leq r/2 + \delta_{\max} \mathcal{K}(\rho_{\max} + u_{\max}), \end{aligned}$$

for all time  $t \in [t_k, t_{k+1}]$ , and  $\delta_{\max} \mathcal{K}(\rho_{\max} + u_{\max}) \leq \delta_{\max} \mathcal{K}(\rho_{\max} + u_{\max}) \leq \frac{(r/2)^2 \lambda_{\min}(Q_\Sigma)}{\rho_{\max} (\lambda_{\max}(W_\Sigma))} < \frac{r}{2}$ . Combining two cases above, we have shown the following: there exists a finite update time  $t_l$  such that  $z(t_l) \in B(z^c; r) \subset \Omega_\beta$ ; at any subsequent update time  $t_k, k > l, z(t_k) \in \Omega_{\beta/2} \subset \Omega_\beta$ ; finally, for any two subsequent update times  $t_k$  and  $t_{k+1}$ , with  $k \geq l, z(t) \in \Omega_\beta$  for all time  $t \in [t_k, t_{k+1}]$ . This completes the proof. ■

The theorem guarantees that, by implementation of Algorithm 1 with  $\delta_{\max}$  given by (20), the closed-loop state trajectory enters the closed ball  $B(z^c; r)$  in finite time and remains in the level set  $\Omega_\beta$  for all future time [9].

## IV. EXPERIMENTAL RESULTS

We implemented our algorithm on real omnidirectional mobile robots shown in Fig. 4. Each has an omnidirectional camera as sole sensor, which is used for self localization [15]. In the experiments, three omnidirectional mobile robots were required to follow an ellipse  $\Gamma : x_p(t) = 1.5 \cos(t), y_p(t) = 1.0 \sin(t)$  (it was rotated  $\pi/6$  radian) with a desired forward speed  $u_o = 0.4m/s$  and to keep a flexible triangle formation (see Fig. 1). The formation error vector was given as  $p_{12} = p_{23} = -0.4m, p_{21} = p_{32} = 0.4m$ . The offset distances from the reference path are defined by  $q_1 = -0.4m, q_2 = 0.4m$ , and  $q_3 = -0.4m$ .

The free package DONLP2 [16] has been used to solve the online optimization problem. However, using hard constraints could make the numerical solution become difficult. To avoid this, we enforce the terminal constraints through the cost function as soft constraints. The average sampling time  $\delta$  can be achieved at approximately 0.12s. All parameters are listed as follows:

$$\begin{aligned} Q_i & = \text{diag}(0.05, 0.05, 0.0001, 0.0001, 0.05), \\ R_i & = \text{diag}(0.00001, 0.00001, 0.00001, 0.000005), \\ P_i & = \text{diag}(0.12, 0.12, 0.02, 0.02, 0.12), \\ k_i & = \text{diag}(0.5, 0.5, 0.5, 0.5), \quad W_i = 0.05, \quad \kappa = 10 \\ T_p & = T_c = 3\delta, \quad \nu_{\min} = -1.9m/s, \quad \nu_{\max} = 1.9m/s. \end{aligned}$$

Fig. 5 shows the superimposed snapshots of three omnidirectional mobile robots following the reference and keeping the flexible triangle formation. As be seen from the results, the outer robot moves faster and the inner robot moves slower when the formation makes a turn. Fig. 7 shows the formation error from the experiments. The pose errors of  $R_1$  and  $R_2$  are shown in Fig. 6(a) and Fig. 6(b), respectively. The velocities of all robots, compared with their reference velocities, are shown in Fig. 8(a) and Fig. 8(b), respectively. Experimental results for  $R_3$  are similar to those of  $R_1$ .

## V. CONCLUSIONS AND FUTURE WORK

In this paper, we present the solution for the formation and the path following problems, where a group of omnidirectional mobile robots is required to follow a reference path while keeping a desired flexible formation in time. Each robot can achieve the desired formation by adjusting the location of its virtual vehicle moving along the reference path, according to its neighbors' information. Distributed NMPC has been used to solve these problems by exchanging the optimal state trajectory between coupled subsystems. As stated in [9], the key

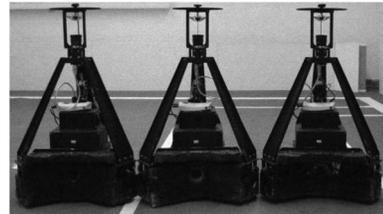


Fig. 4. Omnidirectional mobile robots used in the experiments.

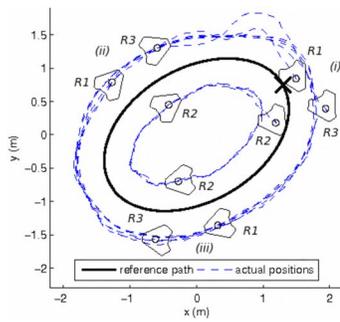


Fig. 5. The snapshots are taken at the following time (i) initial position at  $t = 0s$ , (ii) the formation obtained at  $8.4s$ , and (iii) the formation still maintained at  $t = 18.3s$ . The solid-line ellipse is the given reference path and  $\times$  denotes the starting point.

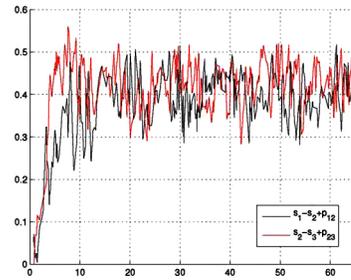


Fig. 7. The formation errors of  $R_1$  and  $R_2$ , and of  $R_2$  and  $R_3$ .

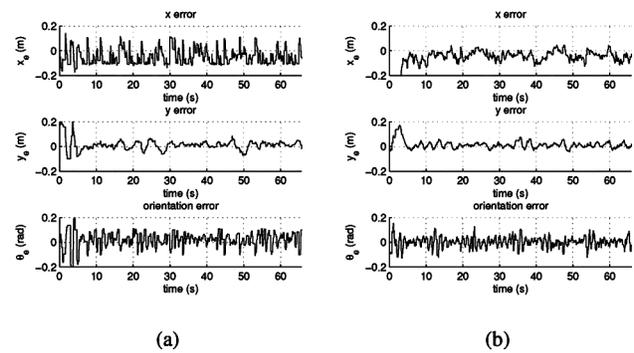


Fig. 6. Pose errors of (a)  $R_1$  and (b)  $R_2$ .

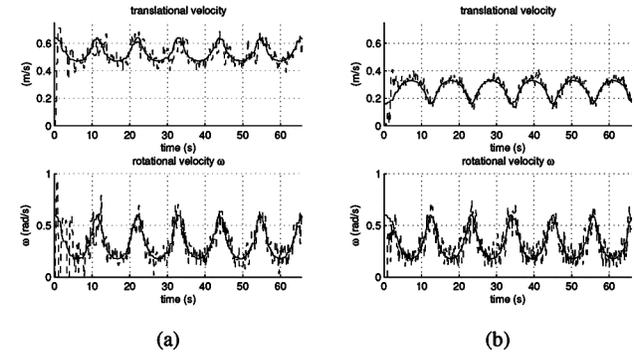


Fig. 8. Velocities of (a)  $R_1$  and (b)  $R_2$ .

requirements for stability are that each subsystem not deviate too far from the previous open-loop state trajectory, and that the predictive horizon updates happen sufficiently fast. Even though very fast updates cannot be achieved, experimental results show efficiency of our proposed algorithm. The less conservative method for stability proof is currently under our investigation.

In the future research, we would like to solve communication problems, e.g., communication delay, packet losses, asynchronous implementation, and time-varying dynamical network.

## REFERENCES

- [1] Y. Q. Chen and Z. M. Wang, "Formation control: a review and a new consideration," in *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Edmonton, Canada, Aug. 2005, pp. 3181-3186.
- [2] R. M. Murray, "Recent research in cooperative-control of multivehicle systems," *Journal of Dynamics, Systems, Measurement and Control*, 129(5), pp. 571-583, 2007.
- [3] A. K. Das, R. Fierro, V. Kumar, J. P. Ostrowski, J. Spletzer, and C. J. Taylor, "A vision-based formation control framework," *IEEE Trans. on Robotics and Automation*, 18(5), pp. 813-825, Oct. 2002.
- [4] W. Ren and R. W. Beard, "A decentralized scheme for spacecraft formation flying via the virtual structure approach," *Journal of Guidance, Control and Dynamics*, 27(1), pp. 73-82, Jan. 2004.
- [5] H. Chen and F. Allgöwer, "A quasi-infinite horizon nonlinear model predictive. control scheme with guaranteed stability," *Automatica*, 34(10), pp. 1205-1218, 1998.
- [6] H. Michalska and D.Q. Mayne, "Robust receding horizon control of constrained nonlinear systems," *IEEE Trans. on Automatic Control*, 38(11), pp. 1623-1633, Nov. 1993.

- [7] D. Jia and B. Krogh, "Min-max feedback model predictive control for distributed control with communication," in *Proc. of American Control Conference*, Anchorage, Alaska, May 2002, pp. 4507-4512.
- [8] A. G. Richards and J. P. How, "A decentralized algorithm for robust constrained model predictive control," in *Proc. of American Control Conference*, Boston, Massachusetts, June 30 - July 2, 2004, pp. 4261-4266.
- [9] W. B. Dunbar and R. M. Murray, "Distributed receding horizon control for multi-vehicle formation stabilization," *Automatica*, 42(4), pp. 549-558, 2006.
- [10] T. Keviczky, F. Borrelli, and G. J. Balas, "A study on decentralized receding horizon control for decoupled systems," in *Proc. of American Control Conference*, Boston, Massachusetts, June 30 - July 2, 2004, pp. 4921-4926.
- [11] D. Soeanto, L. Lapiere, and A. Pascoal, "Adaptive non-singular path-following, control of dynamic wheeled robots," in *Proc. of International Conference on Advanced Robotics*, Coimbra, Portugal, June 30 - July 3, 2003, pp. 1387-1392.
- [12] G. Campion, G. Bastin, and B. D'Andréa-Novel, "Structural properties and classification of kinematic and dynamic models of wheeled mobile robots," *IEEE Trans. on Robotics and Automation*, 12(1), pp. 47-62, Feb. 1996.
- [13] C. Godsil and G. Royle, *Algebraic Graph Theory*. New York: Springer-Verlag, 2001, vol. 207, Graduate Texts in Mathematics.
- [14] J. A. Primbs, V. Nevistic, and J. C. Doyle, "A receding horizon generalization of pointwise min-norm controllers," *IEEE Trans. on Automatic Control*, 45(5), pp. 898-909, May 2000.
- [15] P. Heinemann, J. Haase, and A. Zell, "A combined Monte-Carlo localization and tracking algorithm for RoboCup," in *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China, Oct. 2006, pp. 1535-1540.
- [16] P. Spellucci, "An SQP method for general nonlinear programs using only equality constrained subproblems," *Mathematical Programming*, 82(3), pp. 413-448, 1998.