IET Control Theory & Applications

Brief Paper

Distributed Lyapunov-based model predictive control for collision avoidance of multi-agent formation

ISSN 1751-8644
Received on 5th May 2018
Revised 27th August 2018
Accepted on 10th September 2018
E-First on 15th October 2018
doi: 10.1049/iet-cta.2018.5317
www.ietdl.org

Yaohua Guo[1] ✉, Jun Zhou[1], Yingying Liu[1]
[1]*School of Astronautics, Northwestern Polytechnical University, Xi'an, People's Republic of China*
✉ *E-mail: gyhua@mail.nwpu.edu.cn*

Abstract: This study addresses the problem of distributed formation control for a multi-agent system with collision avoidance between agents and with obstacles, in the presence of various constraints. The authors proposed solution incorporates a control Lyapunov function (CLF) into a distributed model predictive control scheme, which inherits the strong stability property of the CLF and optimises the formation performance. For each agent, the formation tracking objective is formulated through the CLF, while the collision avoidance objective being explicitly considered as constraints. A relaxation parameter is introduced into the CLF condition to make the trade-off between the two conflicting objectives. The terminal constraint is constructed based on the concept of velocity obstacle, which characterises the set of states that lead to collisions. They show that the terminal constraint together with the relaxed CLF-based constraint guarantees the recursive feasibility and stability of the multi-agent system for almost any prediction horizon. Furthermore, the theoretical effectiveness and advantageous implementation properties are demonstrated through simulation for multi-agent formation control with several obstacles.

1 Introduction

Formation control of multi-agent systems has attracted a significant amount of attention due to its efficiency and cost-saving in a wide range of applications, including mobile robots [1, 2], vessels [3], aerial vehicles [4, 5] and spacecraft [6, 7]. The goal of distributed formation control is to steer a team of agents to achieve a desired formation based on the local information from their neighbours (see more details in the recent survey paper [8]). To make formation control more practical, several issues have been addressed in the existing literature, such as switching topologies [9], uncertain dynamics [10], agent dynamics constraints [11, 12] and collision avoidance [13]. In particular, a collision avoidance problem of multi-agent systems has recently received increasing research efforts in the communities of robotics and control systems [14–19].

State-of-the-art techniques for collision-free of multi-agent formation control mainly include the Lyapunov-based method [13–15] and the optimisation-based method [16–18]. In the artificial potential function (APF) approach, one of the Lyapunov-based methods, the gradient of repulsive potential and an attractive potential are used for collision avoidance and the convergence to goals, respectively. Though characterised by ease of implementation and low-computational requirements, the APF approach has one main drawback, i.e. the existence of undesired equilibrium points [20]. To overcome this problem, one needs to choose appropriate potential fields (such as rotational potential field introduced in [13]) or carefully design the parameters of repulsive potential functions and the attractive ones to guarantee the global minima to be located at the desired formation positions [14]. Meanwhile, the APF method is not able to handle various constraints and consider optimality. Regarding the optimisation-based approach with various constraints, the multi-agent collision avoidance problem can be solved through distributed model predictive control (DMPC) [16–18], quadratic programming [19], mixed integer linear programming (MILP) [21], or sequential convex programming [22].

Among the existed methods mentioned above for multi-agent collision avoidance with complex constraints, DMPC is outstanding in terms of the structural flexibility, less computation cost and lower communication burden. Through the DMPC method, each agent can solve the local optimisation problem synchronously by communicating its assumed information with neighbours. To ensure the safety of each agent, Wang and Ding [16] imposed the constraint that depends on the deviation between the assumed and actual predictive information of agents. Using a similar strategy, Dai *et al*. [17] also address the obstacle avoidance problem and Hosseinzadeh Yamchi and Mahboobi Esfanjani [18] further consider the communication delay among agents. The closed-loop stability of the whole multi-agent system in [16–18] is ensured by the development of the positively invariant terminal region and proper terminal controller. The intuition of this method is to let the agents get close enough to the desired goals such that tracking a controller can be applied without considering the collisions with agents and obstacles. However, the development of the terminal region and controller requires much effort and depends on the specific desired formation (see the difference between [16, 17]), which makes it hard to adapt to a variety of control objectives that depend on the desired trajectories. Meanwhile, when the goal positions are far away from the initial positions, a long prediction horizon or a sampling interval is needed to satisfy the terminal constraints, which will lead to the increasing of computational burden. Fortunately, Lyapunov-based model predictive control (LMPC) allows for an explicit characterisation of the stability region and a reduced complexity optimisation problem [23–25]. The main idea behind this method is to realise excellent optimal performance in a receding horizon scheme, while guaranteeing the stability based on the presence of a control Lyapunov function (CLF). Some important recent research studies on DMPC for multi-agent systems include distributed two-stage (primal decomposition and distributed consensus) parametric consensus optimisation [26], reachability analysis concepts based on DMPC for formation control in the presence of time-varying delay and data loss [27], and also DMPC for static formation control using both agents' own and neighbouring inputs as decision variables of optimisation [28], all of which, however, do not take collision avoidance into consideration.

In this study, based on the above discussions, we propose a Lyapunov-based DMPC (LDMPC) algorithm for multi-agent formation control subject to collision avoidance between agents and with obstacles. For implementing the algorithm synchronously, each agent uses the assumed trajectories obtained in the previous time step of its neighbours. We view the realisation of formation control and collision avoidance as conflict objectives. Based on

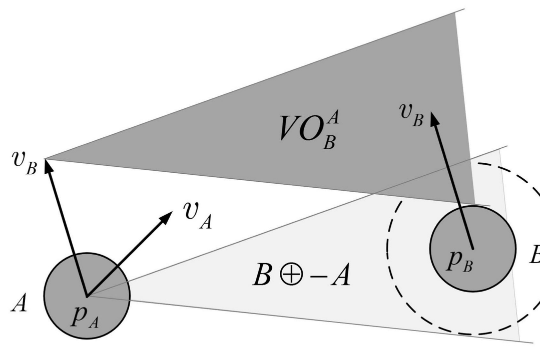

Fig. 1 *VO VO_B^A of obstacle B to agent A*

this idea, we devise a CLF condition related to the objective of formation control for each agent to steer agents towards their goals, while handling the collision avoidance constraints explicitly. The CLF conditions are relaxed to satisfy the collision avoidance constraints by introducing relaxation parameters. To handle collision avoidance, we also impose the constraints of the deviation between the real predictive trajectories and the assumed ones. Meanwhile, different from [16–18], we eliminate the terminal penalty term and construct the terminal constraints based on the concept of velocity obstacle (VO) [29, 30]. The terminal region of each agent is only restricted to be safe, rather than close enough to its desired position. The main contributions of this study can be summarised as follows: (i) compared with the LMPC scheme used in [23–25], we use the relaxed CLF condition rather than the Lyapunov-based controller-based constraint (hard to obtain for multi-agent systems with complex constraints) to guarantee stability, meanwhile, the CLF condition is modified to take into account all the prediction horizons instead of only the first time step; (ii) through introducing relaxation parameters into CLF conditions, feasibility of the formation optimisation problem can always be guaranteed, while making a trade-off between the conflict objectives, i.e. formation control performance and collision avoidance, and (iii) with the novel VO-based terminal constraints and the inherent stability property of CLF different from [16–18], the recursive feasibility and stability of the multi-agent system can be guaranteed for almost any prediction horizon, regardless of where the desired positions are assigned.

The remaining part of this paper is organised as follows. In the next section, necessary notations, the concept of VO and CLFs are given, and then our control problem is formulated. Section 3 presents the main results of LDMPC algorithm with its feasibility and stability. Simulation results for multi-agent formation control with collision avoidance are provided in Section 4, showing the effectiveness of our proposed control strategy. Finally, the conclusion is drawn in the last section.

2 Preliminaries and problem formulation

2.1 Basic notions and definitions

Let the set of integers and positive integers be $\mathbb{Z}$ and $\mathbb{Z}_+$, respectively, and the set of reals be $\mathbb{R}$, we use the notation $\mathbb{R}_{\geq c_1}$ to denote the sets $\{c \in \mathbb{R} \mid c \geq c_1\}$, $\mathbb{R}^{n \times m}$ denotes the set of all $n \times m$ real matrices, for a vector x, $\| x \|$ is the two-norm of x, $\mathbf{0}$ is the zero vector or matrix with appropriate dimensions. Furthermore, the symbol '$\setminus$' denotes set substraction such that $\mathbb{A} \setminus \mathbb{B} \triangleq \{x \in \mathbb{A}, x \notin \mathbb{B}\}$; the Minkowski sum of two sets is denoted by $\mathbb{A} \oplus \mathbb{B} \triangleq \{a + b \mid a \in \mathbb{A}, b \in \mathbb{B}\}$; A function $\gamma : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ belongs to a class $\mathscr{K}$ if it is continuous, strictly increasing and $\gamma(0) = 0$, if additionally $\gamma(s) \to \infty$ as $s \to \infty$, γ is said to belong to class $\mathscr{K}_\infty$.

2.2 Velocity obstacle

We briefly review the concept of VO in this section, which is introduced in [29, 30] for local collision avoidance and navigation of an agent among multiple moving or static obstacles. In two dimensions, VO is defined as follows.

Let A be an agent moving with velocity v_A and B be a dynamic obstacle moving with velocity v_B, whose current positions are denoted by p_A and p_B, respectively. VO_B^A stands for the VO of agent A introduced by obstacle B, which is the set of all velocities of robot A that will result in a collision with obstacle B at some future instant, given by [29]

$$\mathrm{VO}_B^A = \{v \mid \lambda(p_A, v - v_B) \cap B \oplus -A \neq \varnothing\} \tag{1}$$

in which $-A \triangleq \{-a \mid a \in A\}$ denotes the object A reflected in its reference point, and let $\lambda(p, v)$ denote a ray starting at p and heading in the direction v

$$\lambda(p, v) = \{p + tv \mid t > 0\}. \tag{2}$$

A geometric interpretation of region VO_B^A can be seen in Fig. 1 when both agent A and dynamic obstacle B are disc shaped with radius r_A and r_B, respectively. In this case, the definition of VO in (1) can be simplified to

$$\mathrm{VO}_B^A = \{v \mid \exists t > 0, :: t(v - v_B) \in D(p_B - p_A, r_A + r_B)\}, \tag{3}$$

where $D(p, r)$ is an open disc of radius r centred at p.

It can be seen that the agent A and dynamic obstacle B will potentially collide at some future point in time when agent A chooses a velocity inside the region of VO VO_B^A. Also, if the chosen velocity of agent A is outside VO_B^A, the collision will not happen.

2.3 Control Lyapunov function

In this section, the concept of CLF is defined, which is the basis of our later analysis.

Let a general discrete-time system be modelled by

$$x(k+1) = f(x(k)), \quad k \in \mathbb{Z}_+, \tag{4}$$

where $x(k) \in \mathbb{X} \subseteq \mathbb{R}^n$ is the state and $f : \mathbb{R}^n \to \mathbb{R}^n$ is an arbitrary function satisfying $f(0) = 0$.

Definition 1: $V(x)$ is called a CLF for (4), if the following holds [31]:

1) There exist $\mathscr{K}_\infty$-functions γ_1 and γ_2 such that

$$\gamma_1(\| x \|) \leq V(x) \leq \gamma_2(\| x \|). \tag{5}$$

2) There exists a $\mathscr{K}_\infty$-function γ_3 such that

$$V(f(x)) - V(x) \leq -\gamma_3(\| x \|). \tag{6}$$

Lemma 1: Consider system (4) and let V be a Lyapunov function in $\mathbb{X}$, then the origin is an asymptotically stable equilibrium in $\mathbb{X}$ [31].

2.4 Control objectives

Consider a multi-agent system of N_a agents sharing a previously known constrained workspace, with linear second-order, discrete-time dynamics described as follows:

$$z_i(k+1) = Az_i(k) + Bu_i(k), \quad i \in \mathbb{N}_a, \quad k > 0, \tag{7}$$

with $\mathbb{N}_a \triangleq \{1, 2, \ldots, N_a\}$ the index set of the multi-agent system. $z_i(k) = [p_i^\mathrm{T}(k)\, v_i^\mathrm{T}(k)]^\mathrm{T} \in \mathbb{R}^{2n}$ is the ith agent's state, with $p_i(k) \in \mathbb{R}^n$ and $v_i(k) \in \mathbb{R}^n$ denote, respectively, the position and velocity; $u_i(k) \in \mathbb{R}^n$ is the control input of agent i. Under this setting, A and B are, respectively, written as

$$A = \begin{bmatrix} I_n & hI_n \\ 0_n & I_n \end{bmatrix}, \quad B = \begin{bmatrix} \frac{h^2}{2} I_n \\ I_n \end{bmatrix}, \tag{8}$$

where h is the sampling interval.

The state and control input of agent i belong to the admissible sets, i.e.

$$z_i(k) \in \mathbb{Z}_i, \quad u_i(k) \in \mathbb{U}_i, \tag{9}$$

where $\mathbb{Z}_i$ is a compact set; $\mathbb{U}_i \triangleq \{u_i \mid \| u_{il} \| \le u_{il}^M, l = 1, 2\ldots, n\}$ is a closed set containing the origin, with u_{il} the lth component of u_i and u_{il}^M the upper limits of control input constraints.

For agent $i \in \mathbb{N}_a$, the control objective is to steer it to its desired goal, while taking into account constraints imposed by obstacles and other members in the formation, which can be formulated by

$$\lim_{k \to \infty} (p_i(k) - p_r(k)) = p_{ir}^d, \tag{10}$$

$$\lim_{k \to \infty} (p_i(k) - p_j(k)) = p_{ij}^d, \quad \forall j \in \mathbb{N}_i, \tag{11}$$

$$\| p_i(k) - p_j(k) \| \ge 2R_s, \qquad \forall j \in \mathbb{N}_a \setminus \{i\}, k \ge 0, \tag{12}$$

$$\| p_i(k) - p_m^o \| \ge R_s + R_o, \qquad \forall m \in \mathbb{N}_o, k \ge 0, \tag{13}$$

where p_r is the position of a virtual reference agent of the formation, p_{ir}^d is the desired tracking vector between agent i and the reference trajectory, p_{ij}^d is the desired relative position vector between agents i and j, and p_m^o stands for the position of obstacle m; $\mathbb{N}_i \triangleq \{j \in \mathbb{N}_a \mid p_{ij}^d \text{ is desired}\}$ represents the neighbour index set of agent i, $\mathbb{N}_o \triangleq [1, 2, \ldots, M]$ with M the number of obstacles; R_s is the safety radius of each agent, and R_o is the radius of disc-shaped obstacles. To guarantee the collision avoidance constraint (12) between any two agents, each agent i has to obtain the information of all other agents. Hence, a global communication topology among agents is required in this study (similar to the existing works [16, 17]).

Meanwhile, to ensure the consistency of the control objective, the desired positions of agents are assumed to satisfy

$$\begin{aligned} &\| p_{ir}^d - p_{jr}^d \| \ge 2R_s, \quad i \in \mathbb{N}_a, \quad j \in \mathbb{N}_i, \\ &\| p_r + p_{ir}^d - p_m^o \| \ge R_s + R_o, \quad i \in \mathbb{N}_a, \quad m \in \mathbb{N}_o, \end{aligned} \tag{14}$$

which guarantees that the desired position of each agent is consistent with the desired relative positions and with the obstacle avoidance constraint.

3 Main results

In this section, we discuss the proposed distributed multi-agent formation control algorithm through theLDMPC. In our synchronous model predictive control (MPC) framework, the information including the predicted control and predicted trajectory of its neighbours is not available for each agent $i \in \mathbb{N}_a$. Hence, before the construction of a multi-agent formation optimisation problem, we introduce the assumed predictive trajectories commonly used in DMPC-based formation control (see more details in [16–18]). To distinguish the different predictive trajectories, we further clarify the following notations:

$u_i^*(k + l \mid k)$ optimal predictive input of agent i obtained in the kth time step, whose first element is applied to agent i; $\hat{u}_i(k + l \mid k)$ assumed predictive input of agent i, which is used for the prediction of its trajectory.

The corresponding states including position and velocity evolved according to $u_i^*(k + l \mid k)$ and $\hat{u}_i(k + l \mid k)$ are distinguished in the same manner, i.e. $z_i^*(k + l \mid k) = [p_i^{*\mathrm{T}}(k + l \mid k)\ v_i^{*\mathrm{T}}(k + l \mid k)]^{\mathrm{T}}$ stands for the actual (i.e. optimal) predictive states and $\hat{z}_i(k + l \mid k) = [\hat{p}_i^{\mathrm{T}}(k + l \mid k)\ \hat{v}_i^{\mathrm{T}}(k + l \mid k)]^{\mathrm{T}}$ stands for the assumed predictive states.

Let N denote the prediction horizon of MPC and k be the current time step. Then, the assumed control and states of agent i are defined as

$$\hat{u}_i(k + l \mid k) = \begin{cases} u_i^*(k + l \mid k - 1), & l = 0, 1, \ldots, N - 2, \\ \kappa_i(z_i^*(k - 1 + N \mid k - 1)), & l = N - 1, \end{cases} \tag{15}$$

with $\kappa_i(\cdot)$ one of the feasible controls when $l = N - 1$, a similar terminal controller in conventional MPC. Correspondingly, agent i's assumed state is

$$\hat{z}_i(k + l \mid k) = \begin{cases} z_i^*(k + l \mid k - 1), & l = 0, 1, \ldots, N - 1m, \\ z_i^{\kappa}(k + N \mid k - 1), & l = Nmm, \end{cases} \tag{16}$$

where $z_i^{\kappa}(k + N \mid k - 1)$ stands for the next state of $z_i^*(k + N \mid k - 1)$ by using the assumed control $\kappa_i(z_i^*(k - 1 + N \mid k - 1))$.

3.1 Collision avoidance constraint design

To make the assumed states be able to be utilised to optimise the formation of control problem in a distributed manner, the deviation between the optimal predictive trajectories and the assumed ones should be carefully restricted. In this section, constraints for collision avoidance are given.

For agent i, to ensure collision avoidance with agent $j \in \mathbb{N}_a \setminus \{i\}$ by utilising the assured predictive positions of other agents, $\forall l = 0, 1, \ldots, N - 1$, two constraints are imposed: the constraint for position compatibility

$$\| p_i(k + l \mid k) - \hat{p}_i(k + l \mid k) \| = \xi_i^p(k + l \mid k) \le \mu_i(k + l \mid k) \tag{17}$$

and

$$\| p_i(k + l \mid k) - \hat{p}_j(k + l \mid k) \| \ge 2R_s + \mu_{ij}(k + l \mid k), \tag{18}$$

where $\mu_i(k + l \mid k) = \min_{j \in \mathbb{N}_a \setminus \{i\}} \mu_{ij}(k + l \mid k)$, with the position deviation $\mu_{ij}(k + l \mid k)$ chosen as [17]

$$\mu_{ij}(k + l \mid k) = \frac{\| \hat{p}_i(k + l \mid k) - \hat{p}_j(k + l \mid k) \| - 2R_s}{2}. \tag{19}$$

Following the constraints (17) and (18) and the triangle inequality property, as shown in (see (20)) (where $\xi_j^p(k + l \mid k)$ is the position compatibility constraint of agent j), the collision avoidance can be guaranteed for the actual predicted positions.

To avoid collisions with obstacle $m \in \mathbb{N}_o$, the following constraint should also be considered:

$$\begin{aligned} &\| p_i(k + l \mid k) - p_j(k + l \mid k) \| \\ &\quad = \| p_i(k + l \mid k) - \hat{p}_j(k + l \mid k) + \hat{p}_j(k + l \mid k) - p_j(k + l \mid k) \| \\ &\quad \ge \| p_i(k + l \mid k) - \hat{p}_j(k + l \mid k) \| - \xi_j^p(k + l \mid k) \\ &\quad \ge \| p_i(k + l \mid k) - \hat{p}_j(k + l \mid k) \| - \mu_{ij}(k + l \mid k) \\ &\quad \ge 2R_s, \end{aligned} \tag{20}$$

$$\| p_i(k+l|k) - p_m^o \| \geq R_s + R_o, \quad l = 0, 1, \ldots, N-1. \tag{21}$$

Note that the position deviation bound $\mu_{ij}(k+l|k)$ plays an important role for the trade-off between optimised performance and the feasibility of collision avoidance. Specifically, the larger bound $\mu_{ij}(k+l|k)$ is chosen, the larger margin can be left to satisfy (17) for optimising the formation performance, while resulting in a smaller margin to satisfy the collision avoidance constraint (18). Considering the obstacle avoidance constraint (21), we can see that larger radii R_s and R_o require a larger region to avoid obstacles, while leaving a smaller margin to improve the trajectories, and thus degrading the optimal performance.

3.2 Terminal constraints design

Different from the terminal region used in the traditional DMPC for multi-agent formation control (such as [16–18]), in this section, we aim to construct a terminal constraint for each agent such that it only needs to be safe, instead of getting close enough to its desired position when $l = N$. The concept of VO is utilised to characterise the trajectory of each agent that leads to collision with neighbouring agents and with static obstacles, assuming that the agents are disc-shaped and maintain the velocities in the next time step. For this purpose, we modify (3) in finite horizon h to

$$\mathrm{VO}_B^A(h) = \{v | \| p_A - p_B + h(v - v_B) \| < r_A + r_B\} \tag{22}$$

with h the sampling interval.

To avoid collisions among agents and with static obstacles, the terminal constraint is designed for agent $i \in \mathbb{N}_a$ based on the VO (22), which is described by, $\forall j \in \mathbb{N}_a \setminus \{i\}, m \in \mathbb{N}_o$

$$\| p_i(k+N|k) - \hat{p}_i(k+N|k) \| \leq \mu_i(k+N|k), \tag{23a}$$

$$\| p_i(k+N|k) - \hat{p}_i(k+N|k) \| \geq 2R_s + \mu_{ij}(k+N|k), \tag{23b}$$

$$\| p_i(k+N|k) - p_m^o \| m, \qquad \geq R_s + R_o, \tag{23c}$$

$$\begin{aligned} \| p_i(k+N|k) - \hat{p}_i(k+N|k) + h(v_i(k+N|k) - \hat{v}_i(k+N|k)) \| \\ \leq \chi_i(k+N|k), \end{aligned} \tag{23d}$$

$$\begin{aligned} \| p_i(k+N|k) - \hat{p}_j(k+N|k) + h(v_i(k+N|k) - \hat{v}_j(k+N|k)) \| \\ \geq 2R_s + \chi_{ij}(k+N|k), \end{aligned} \tag{23e}$$

$$\| p_i(k+N|k) - p_m^o + hv_i(k+N|k) \| \geq R_s + R_o, \tag{23f}$$

where $\chi_i(k+N|k) = \min_{j \in \mathbb{N}_a \setminus \{i\}} \chi_{ij}(k+N|k)$ with $\chi_{ij}(k+N|k)$ chosen as $\chi_{ij}(k+N|k) = (\| \hat{p}'_{ij} \| - 2R)/2$, in which $\hat{p}'_{ij}$ is used to represent the term

$$\| \hat{p}_i(k+N|k) - \hat{p}_i(k+N|k) + h(\hat{v}_i(k+N|k) - \hat{v}_i(k+N|k)) \| .$$

It is noted that constraints (23d)–(23f) are derived through the complement of VO, while the first three constraints being directly obtained from the results of the previous section by letting $l = N$. More specifically, the first three constraints ensure that the collisions will not occur at the last prediction horizon, and the last three constraints make sure that the choosing velocities (outside the VO) will not lead to collisions in the next time step. Meanwhile, it can be seen that the terminal constraints are not related to agents' desired positions, and one feasible control input $\hat{u}_i(k+N|k) = \mathbf{0}, i \in \mathbb{N}_a$ (result in unchanging velocity) can be directly obtained at prediction horizon $(k+N|k)$.

Remark 1: The terminal constraints based on the concept of VO benefit in two aspects compared with the positively invariant terminal-state region constraints developed in [16–18]. First of all, the agents need not to be restricted to get close enough to their desired positions in the last prediction horizon, and thus avoid the use of the long MPC control horizon or long sampling interval when the goal positions are far away. Meanwhile, the recursive feasibility can be easily guaranteed without the designing of auxiliary or terminal controllers, since the collision-free trajectories can be selected from the VO's complement.

3.3 Optimisation problem construction

This section formulates the Lyapunov-based finite horizon optimisation problem of agent $i \in \mathbb{N}_a$.

Problem 1: At time step k, given z_i, z_i^d and $\hat{z}_{-i}(k+l|k), l = 0, 1, \ldots, N-1$, the following distributed Lyapunov-based MPC optimisation problem is solved:

$$\min_{(u_i, \delta_i)(k+l|k)} J_i(k, z_i, \hat{z}_{-i}, z_i^d, u_i, \delta_i) \tag{24}$$

subject to

$$z_i(k|k) = z_i(k), \tag{25}$$

$$z_i(k+l+1|k) = Az_i(k+l|k) + Bu_i((k+l|k)), \tag{26}$$

$$z_i(k+l|k) \in \mathbb{Z}_i, \tag{27}$$

$$u_i(k+l|k) \in \mathbb{U}_i, \tag{28}$$

$$\begin{aligned} &V_i(k+l+1|k, z_i, \hat{z}_{-i}, z_i^d) - V_i(k+l|k, z_i, \hat{z}_{-i}, z_i^d) \\ &\leq -(\rho_i - \delta_i(k+l|k))V_i(k+l|k, z_i, \hat{z}_{-i}, z_i^d), \end{aligned} \tag{29}$$

$$\delta_i(k+l|k)) \leq \rho_i, \tag{30}$$

$$\text{and (17), (18), (21), and (23),}$$

where the cost function of agent i is defined as

$$\begin{aligned} &J_i(k, z_i, \hat{z}_{-i}, z_i^d, u_i, \delta_i) \\ &= \sum_{l=0}^{N-1} \Omega_i(k+l|k, z_i, \hat{z}_{-i}, z_i^d, u_i) + \sum_{l=0}^{N-1} W_i(k+l|k, \delta_i) \end{aligned} \tag{31}$$

with $z_i^d = [p_r + p_{ir}^d; \mathbf{0}]$ and the formation control objective function Ω_i, given weight parameters $\alpha_i > 0, \beta_i > 0, \eta_i > 0$

$$\begin{aligned} &\Omega_i(k+l|k, z_i, \hat{z}_{-i}, z_i^d, u_i) \\ &= \alpha_i \| z_i(k+l|k) - z_i^d \|^2 \\ &+ \beta_i \sum_{j \in \mathbb{N}_i} \| z_i(k+l|k) - \hat{z}_j(k+l|k) - z_{ij}^d \|^2 \\ &+ \eta_i \| u_i(k+l|k) \|^2 \end{aligned} \tag{32}$$

and

$$W_i(k+l|k, \delta_i) = \zeta_i \| \delta_i(k+l|k) \|^2 \tag{33}$$

in which $\zeta_i > 0$ is a weight constant and $\delta_i(k+l|k)$ is the relaxation parameter to be optimised. Furthermore, (29) is the Lyapunov-based constraint with ρ_i a positive constant and $(\rho_i - \delta_i)$ determining the convergence speed of the multi-agent formation, in which the CLF V_i is chosen as the first two terms of Ω_i, i.e.

$$\begin{aligned} &V_i(k+l|k, z_i, \hat{z}_{-i}, z_i^d) \\ &= \alpha_i \| z_i(k+l|k) - z_i^d \|^2 \\ &+ \beta_i \sum_{j \in \mathbb{N}_i} \| z_i(k+l|k) - \hat{z}_j(k+l|k) - z_{ij}^d \|^2 , \end{aligned} \tag{34}$$

where the first term and the second one on the right side of (34) are the tracking potential function and the formation potential function, respectively. Constraint (30) for relaxation parameter is utilised to guarantee the stability of the multi-agent system. Constraint (30) is reasonable due to the fact that even when the two objectives (i.e. formation control and collision avoidance) conflict in the worst case, the agents can stop steering towards their goals and focus on collisions avoidance, which is realised by choosing the relaxation parameter $\delta_i(k+l|k)$ to be equal to ρ_i.

The constructed Lyapunov-based optimisation problem exhibits several attractive features as concluded by the following remarks.

Remark 2: The benefit of Lyapunov-based constraint (29) taking into account all the prediction horizon is twofold: (i) providing the trajectories of the multi-agent formation for collision avoidance over the whole prediction horizon, and (ii) ensuring that the formation can always converge over the prediction horizon, in contrast to the traditional LMPC that considers only the first time step. The only disadvantage of considering stability constraint over all the prediction horizon, compared with the conventional LMPC method, is a small increase in computation load.

Remark 3: Constraint (29) does not need an explicit Lyapunov controller, which is hard to design and may become infeasible when considering the collision avoidance and the bounded control input. Meanwhile, constraint (29) can always be satisfied through adjusting the relaxation parameter δ_i.

Remark 4: The relaxation parameter δ_i offers a trade-off between formation control and collision avoidance behaviour, i.e. when the two objectives conflict, δ_i adjusts to slow down the convergence rate, and when they do not conflict, δ_i can be chosen as 0. In this sense, it works similar to parameter governors in [32].

Remark 5: The terminal constraints (23) constructed through the concept of VO are used to ensure the safety of agents, while terminal constraints in traditional DMPC playing an important role in guaranteeing the stability of the closed-loop system. Furthermore, we avoid the complex process of designing the terminal controller in [16–18], which depends on the desired goals.

3.4 Distributed controller implementation

We describe the implementation procedure of our synchronous LDMPC algorithm as follows, which includes the initialisation stage and on-line state.

Algorithm 1: Initialisation: For the ith agent, choose the weight parameters $\alpha_i, \beta_i, \eta_i$, and ζ_i, convergence rate constant ρ_i, safety radius R_s, sampling interval h, prediction horizon N, and maximum control steps $N_{\max}$.

On-line stage: At the initial step $k=0$, the assumed state of agent i is set such that $\hat{z}_i(0+l|0)=z_i(0), l=0,1,\dots,N$, sends $\hat{z}_i(0+l|0)$ to $j\in\mathbb{N}_a\setminus\{i\}$, and waits for the initial optimal control inputs $u_i^*(0+l|0), l=0,1,\dots,N-1$, and the initial optimal states $z_i^*(0+l|0)=z_i(0), l=0,1,\dots,N$. When $k\in\mathbb{Z}_+$, it performs the following procedure:

(1) At time step $k\in\mathbb{Z}_+$, for each agent i

(a) using its own sampled state and other agents' assumed information, compute $\mu_{ij}(k+l|k), \mu_i(k+l|k)$ for $l=0,1,\dots,N-1$ and $\chi_{ij}(k+N|k), \chi_i(k+N|k)$,

(b) solve the optimisation problem to obtain the optimal control inputs $u_i^*(k+l|k), l=0,1,\dots,N-1$ and the optimal states $z_i^*(k+l|k), l=0,1,\dots,N$.

(2) Over the time steps k to $k+1$,

(a) apply $u_i(k)=u_i^*(0|k)$,

(b) obtain its assumed control input $\hat{u}_i(k+1+l|k+1), l=0,1,\dots,N-1$ and the corresponding assumed state $\hat{z}_i(k+1+l|k+1), l=0,1,\dots,N$ using (15) and (16),

(c) send its assumed state $\hat{z}_i(k+1+l|k+1)$ to other agents and receive $\hat{z}_j(k+1+l|k+1)$ from agent $j\in\mathbb{N}_a\setminus\{i\}$.

(3) When $k=N_{\max}$, stop.

Remark 6: Compared with the iterative or sequential update strategy in other optimisation-based formation control, such as [21, 22], the synchronous DMPC is more desired due to its efficiency in terms of communication and computation.

3.5 Formation feasibility and stability

In this section, the main properties of the proposed distributed Lyapunov-based model predictive formation control of the multi-agent system are presented in a theorem. Before presenting the theorem, two necessary and reasonable assumptions are given.

Assumption 1: There exist $\bar{a}_2, a_e\in\mathbb{R}_{\geq 0}$, such that

$$\Omega_i(k+l|k, z_i, \hat{z}_{-i}, z_i^d, u_i) \leq \bar{a}_2 \parallel \Delta z_i(k+l|k) \parallel^{a_e}, \tag{35}$$

where $\Omega_i(k+l|k, z_i, \hat{z}_{-i}, z_i^d, u_i)$ is the formation control objective function (32), and $\Delta z_i(k+l|k)=z_i(k+l|k)-z_i^d$.

Assumption 2: There exists $N_R\in\mathbb{Z}_+, \underline{a}_3\in\mathbb{R}_{>0}$, such that when the prediction horizon $N\geq N_R$, there is at least one step in which the relax parameter δ_i can be chosen less than ρ_i, i.e.

$$\sum_{l=0}^{N-1}(\rho_i-\delta_i(k+l|k)) \geq \underline{a}_3. \tag{36}$$

Assumption 1 ensures the control objective function of the multi-agent system is bounded, which always hold since both the states and inputs of all the agents are restricted to be bounded. Assumption 2 ensures that over a sufficiently long prediction horizon, constraint (29) in Problem 1 can be satisfied, and therefore the agents can steer towards their desired positions in at least one horizon. Assumption 2 is reasonable since long prediction horizons means that the constraints can be guaranteed more easily with a given admissible set of control inputs. In fact, if the environment is not so crowed, a short prediction horizon is enough to satisfy this assumption, which is also illustrated through our simulation.

Theorem 1: For the multi-agent system with dynamics (7), each agent solves the Lyapunov-based MPC optimisation problem 1 according to Algorithm 1, suppose there exists a feasible solution for each agent at time step $k=0$, under Assumptions 1 and 2 and choose $N\geq N_R$, then, the optimisation problem is feasible for all $k\in\mathbb{Z}+$, and the collision between agents and with obstacles can be avoided for all the future time steps. Furthermore, all the agents asymptotically converge to their desired positions and realise the formation, i.e. $p_i(k)-p_r(k)\to p_{ir}^d$ and $(p_i(k)-p_j(k))\to p_{ij}^d$ as $k\to\infty$.

Proof: Feasibility : It is assumed that at time step $k=0$ the optimisation problem of agent i is feasible. The corresponding optimising state and control trajectory are denoted by $z_i^*(k+l|k), l=0,1,\dots,N$ and $u_i^*(k+l|k), l=0,1,\dots,N-1$, respectively. Based on these information, the assumed control input and assumed state in the next time step can be obtained through (15) and (16). We denote hereafter by $\tilde{u}_i(\cdot)$ and $\tilde{z}_i(\cdot)$ the feasible control and state of agent i in certain time step. Now, Let

$\tilde{u}_i(k+1+l|k+1)=\hat{u}(k+1+l|k+1)$ and

$\tilde{z}_i(k+1+l|k+1)=\hat{z}_i(k+1+l|k+1)$.

It will be shown that at time step $k+1$, the input sequence

$\{\tilde{u}_i(k+1+l|k+1) = \hat{u}_i(k+1+l|k+1), l = 0, 1, \ldots, N-1\}$ with the last element be $\mathbf{0}$ is one feasible solution to agent i. Applying $\tilde{u}_i(k+1+l|k+1)$ to agent i, due to the definition of the assumed control input, constraints (17), (18), (21), and (26)–(30) are satisfied directly. Meanwhile, according to the construction of the VO-based terminal constraint of agent i, which assumes that its velocity remains at the end of the prediction horizon, choosing $\tilde{u}_i(k+1+(N-1)|k+1) = \kappa_i(z_i^*(k+N|k)) = \mathbf{0}$ can thus satisfy the terminal constraint. Therefore, the feasibility of the optimisation problem at time step k leads to the feasibility at next time step $k+1$. By recursion, it can be concluded that the optimisation problem is feasible for all the control steps. Moreover, the collision avoidance will never occur due to the feasibility of the constraints (17), (18), and (21).

Stability : Consider the summation of optimal costs of all the agents by

$$\Omega_{\Sigma}^*(k) = \sum_{i \in \mathbb{R}_a} \sum_{l=0}^{N-1} \Omega_i^*(k+l|k, z_i^*, \hat{z}_{-i}, z_i^d, u_i^*) \tag{37}$$

then, under Assumption 1, $\gamma_1 \in \mathscr{K}_\infty$ and $\gamma_2 \in \mathscr{K}_\infty$ exist to satisfy

$$\gamma_1(\parallel \Delta Z^*(k) \parallel) \le \Omega_{\Sigma}^*(k) \le \gamma_2(\parallel \Delta Z^*(k) \parallel), \tag{38}$$

where $\Delta Z^*(k) \triangleq [\Delta Z_1^{*\mathrm{T}}(k), \Delta Z_2^{*\mathrm{T}}(k), \ldots, \Delta Z_{N_a}^{*\mathrm{T}}(k)]^{\mathrm{T}}$ is the error states of all the agents over the prediction horizon with $\Delta Z_i^*(k) \triangleq [\Delta z_i^{\mathrm{T}}(k|k), \Delta z_i^{\mathrm{T}}(k+1|k), \ldots, \Delta z_i^{\mathrm{T}}(k+N-1|k)]^{\mathrm{T}}$.

The summation costs at next time step $k+1$, associated with feasible control solutions $\tilde{u}_i(k+1+l|k+1) = \hat{u}_i(k+1+l|k+1)$, can be formulated as

$$\tilde{\Omega}_{\Sigma}(k+1) = \sum_{i \in \mathbb{N}_a} \sum_{l=0}^{N-1} \Omega_i(k+1, \tilde{z}_i, \hat{z}_{-i}, z_i^d, \tilde{u}_i). \tag{39}$$

The difference between the feasible cost and the optimal cost is

$$\begin{aligned}
&\tilde{\Omega}_{\Sigma}(k+1) - \Omega_{\Sigma}^*(k) \\
&= \sum_{i \in \mathbb{N}_a} \sum_{l=0}^{N-1} \Big(\Omega_i(k+1+l|k+1, \tilde{z}_i, \hat{z}_{-i}, z_i^d, \tilde{u}_i) \\
&\quad -\Omega_i(k+l|k, z_i^*, \hat{z}_{-i}, z_i^d, u_i^*)\Big) \\
&= \sum_{i \in \mathbb{N}_a} \sum_{l=0}^{N-1} \Big[V_i(k+1+l|k+1, \hat{z}_i, \hat{z}_{-i}, z_i^d, \tilde{u}_i) \\
&\quad -V_i(k+l|k, z_i^*, \hat{z}_{-i}, z_i^d, u_i^*) \\
&\quad +\eta_i\Big(\parallel \hat{u}_i(k+l+1|k+1) \parallel^2 - \parallel u_i^*(k+l|k) \parallel^2\Big)\Big] \\
&= \sum_{i \in \mathbb{N}_a} \Big\{\sum_{l=0}^{N-1} \Big(-(\rho_i - \delta_i(k+l|k)) \\
&\quad \times V_i(k+l|k, z_i^*, \hat{z}_{-i}, z_i^d, u_i^*)\Big) \\
&\quad +\eta_i \parallel \kappa_i(z_i^*(k+N|k)) \parallel^2 - \eta_i \parallel u_i^*(k|k) \parallel^2\Big\} \\
&\le \sum_{i \in \mathbb{N}_a} \Big\{\sum_{l=0}^{N-1} -\alpha_i(\rho_i - \delta_i(k+l|k)) \parallel \Delta z_i^*(k+l|k) \parallel^2 \\
&\quad +\mu_i \parallel \kappa_i(z_i^*(k+N|k)) \parallel^2\Big\}
\end{aligned} \tag{40}$$

in which we use the following inequality:

$$\alpha_i \parallel \Delta z_i^*(k+l|k) \parallel^2 \le V_i(k+l|k, z_i^*, \hat{z}_{-i}, z_i^d, u_i^*). \tag{41}$$

Under Assumption 2 and the fact that $\kappa_i(z_i^*(k+N|k)) = \mathbf{0}$ is one feasible solution, it directly follows from (40) that there exists $\gamma_3 \in \mathscr{K}_\infty$ such that

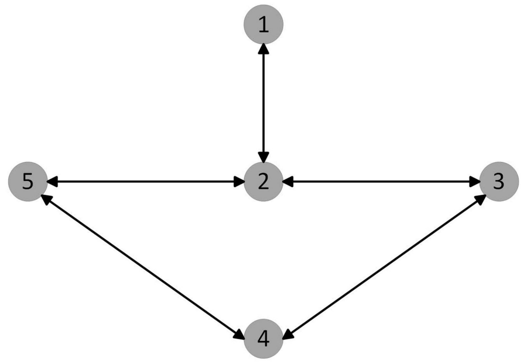

Fig. 2 *Network for the neighbours of the multi-agent system*

$$\tilde{\Omega}_{\Sigma}(k+1) - \Omega_{\Sigma}^*(k) \le -\gamma_3(\parallel \Delta Z^*(k) \parallel). \tag{42}$$

Considering the optimality of the solution in time step $k+1$, we derive that

$$\begin{aligned}
\Omega_{\Sigma}^*(k+1) - \Omega_{\Sigma}^*(k) &\le \tilde{\Omega}_{\Sigma}(k+1) - \Omega_{\Sigma}^*(k) \\
&\le -\gamma_3(\parallel \Delta Z^*(k) \parallel).
\end{aligned} \tag{43}$$

According to Lemma 1, together with (38) and (43), $\Omega_{\Sigma}^*(k)$ asymptotically converges to zero, with $\Delta z_i(k) = \mathbf{0}, k \to 0, i \in \mathbb{N}_a$ its stable equilibrium, which implies that all the agents converge to their desired positions and realise the formation configuration. □

Remark 7: The closed-loop stability of the multi-agent formation does not depend on the terminal region as the traditional DMPC method, but on the Lyapunov-based constraint. In case that the environment is not so crowd, choosing a short prediction horizon (which we call almost every prediction horizon) is enough to guarantee the stability.

4 Simulation results

This section presents the simulation results of a formation composed by five linear second-order agents. The network among neighbours of the multi-agent system is shown by Fig. 2, whose adjacent matrix $\mathscr{E} = [e_{ij}], i, j = 1, 2, \ldots, 5$ is given by

$$\mathscr{E} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix},$$

where $e_{ij} = 1$ means that agent j is the neighbour of agent i. The dynamics of agents is given by (7) with $n = 2$. The control input constraints are given by

$u_i \in \mathbb{U}_i \triangleq \{u_i | -0.5\mathrm{m/s^2} \le u_{i1}, u_{i2} \le 0.5\mathrm{m/s^2}\}, i = 1, 2, \ldots, 5.$

The constant weights of the optimisation problem, for all the agents $i \in \mathbb{N}_a$, are chosen to be $\alpha_i = 1$, $\beta_i = 0.2$, $\eta_i = 1$, and $\zeta_i = 1$. The convergence rate is chosen as $\rho_i = 0.1$. All the agents are assumed to be disc-shaped with safety distance $R_s = 0.2\,\mathrm{m}$. Two circle obstacles with $R_o = 0.4\,\mathrm{m}$ are located in the workplace, whose positions' vectors are $p_1^o = [-3; -1]$ and $p_2^o = [1; -2]$. The desired positions of the multi-agent system are given by

$$\begin{aligned}
&p_r = [2; 0], \quad p_{1r}^d = [\sqrt{2}; -\sqrt{2}], \quad p_{2r}^d = [0; 0], \\
&p_{3r}^d = [-2; 0], \quad p_{4r}^d = [-2; 2], \quad p_{5r}^d = [0; 2].
\end{aligned}$$

The corresponding desired relative positions between neighbours are given as

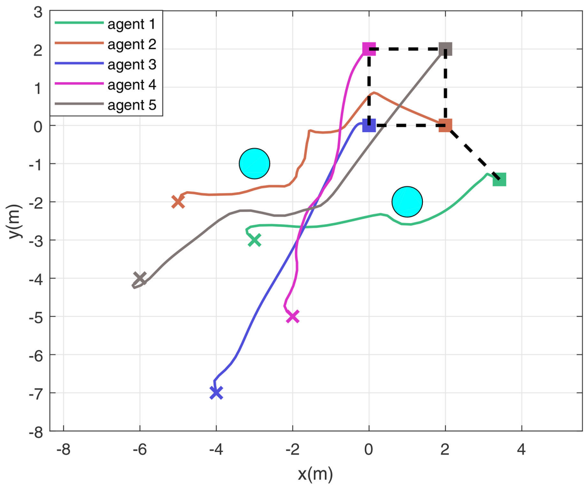

Fig. 3 *Trajectories history of the multi-agent system*

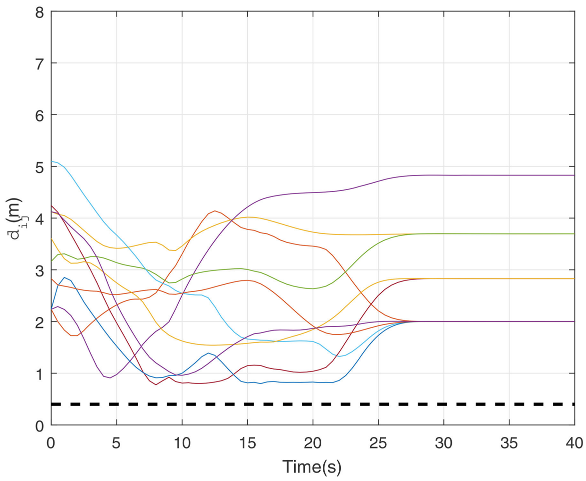

Fig. 4 *Inter-agent distances d_{ij}*

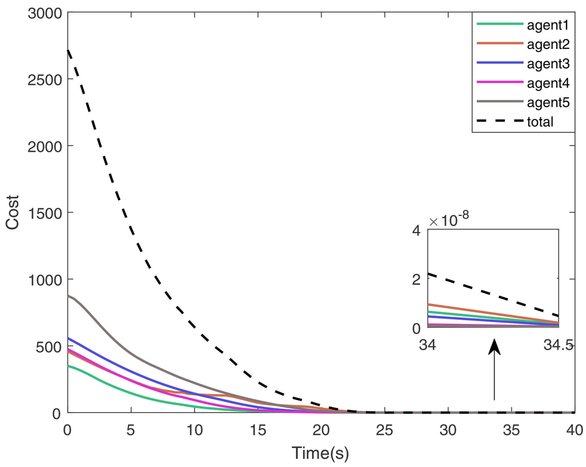

Fig. 5 *Cost function of each agent*

$$
\begin{aligned}
p_{12}^d &= [-\sqrt{2}; \sqrt{2}], & p_{21}^d &= [\sqrt{2}; -\sqrt{2}],\\
p_{23}^d &= [-2; 0], & p_{32}^d &= [2; 0],\\
p_{25}^d &= [0; 2], & p_{52}^d &= [0; -2],\\
p_{34}^d &= [0; 2], & p_{43}^d &= [0; -2],\\
p_{45}^d &= [2; 0], & p_{54}^d &= [-2; 0] .
\end{aligned}
$$

The initial positions and velocities of agents are

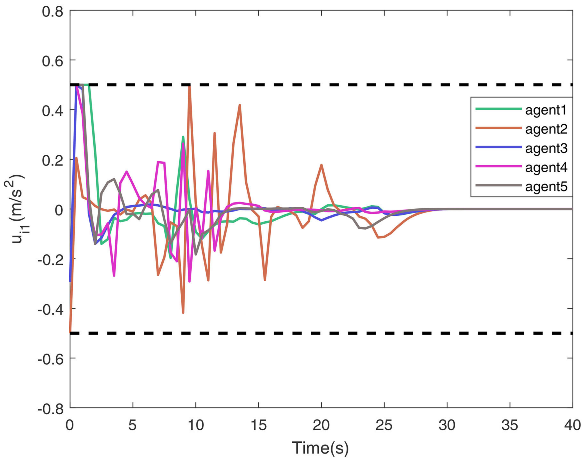

Fig. 6 *First control input element of agents*

$$
\begin{aligned}
z_1(0) &= [-3 \ -3 \ -0.5 \quad 0.5]^{\mathrm{T}},\\
z_2(0) &= [-5 \ -2 \quad 0.5 \quad 0.5]^{\mathrm{T}},\\
z_3(0) &= [-4 \ -7 \quad 0.0 \quad 0.5]^{\mathrm{T}},\\
z_4(0) &= [-2 \ -5 \ -0.5 \quad 0.5]^{\mathrm{T}},\\
z_5(0) &= [-6 \ -4 \ -0.5 \ -0.5]^{\mathrm{T}} .
\end{aligned}
$$

Then, follow the procedure of the LDMPC algorithm in Section 3.4, the constructed multi-agent formation optimisation problem is implemented in Matlab with the toolboxes ICLOCS [33] and IPOPT [34].

To begin with, the prediction horizon is chosen to be $N = 8$, the maximum control step $N_{\max} = 80$, and the sampling interval $h = 0.5\,\mathrm{s}$. The trajectories history of agents are shown in Fig. 3, from which we can see that all the agents avoid the obstacles and finally reach their desired positions. Fig. 4 further shows that the relative distances d_{ij} between each two agents are larger than the safety distance $2R_s$ that is depicted by the dashed line, namely, they never collide with each other. After about 28 s, the inter-agent distances remain the same, which implies the realisation of the desired formation configuration. The values of the cost function of each agent and the total cost are shown in Fig. 5 to demonstrate the stability of the formation control through our LDMPC algorithm. Fig. 6 shows the control inputs of agents that always stay within the control bounds. The attractive feature of our introduced relaxation parameters can be seen in Fig. 7, which adjust themselves according to the level of conflicts between formation control and collision avoidance objective.

Furthermore, we illustrate the stability of our LDMPC formation control framework for almost any prediction horizon. For this purpose, choosing the minimum prediction horizon in MPC, i.e. $N = 2$. Similar to case $N = 8$, we can see from Fig. 8 that the agents converge to their desired positions without obstacle collision. Meanwhile, Fig. 9 shows that the collisions among agents never occur since the distances d_{ij} between them remain larger than $2R_s$.

5 Conclusion

In this study, a Lyapunov-based synchronous DMPC for multi-agent formation with collision avoidance is studied. The concept of VO is first designed to construct the terminal constraint for each agent, which does not depend on its desired position and thus can apply to various formation missions. Then, the CLF condition contributes to the stability of the multi-agent system, while the MPC framework solves the formation optimisation problem under several constraints. To guarantee the feasibility of formation control, we introduce an adjustable relaxation parameter into the CLF condition to make a trade-off between formation control objective and collision avoidance. Also, it is shown that the proposed LDMPC algorithm makes all the agents asymptotically

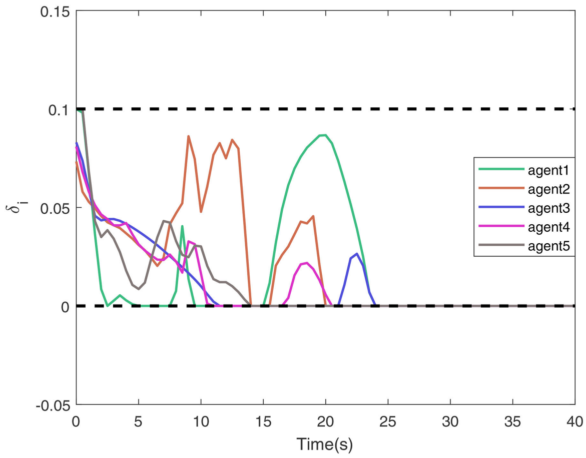

Fig. 7 *Relax parameter δ_i histories*

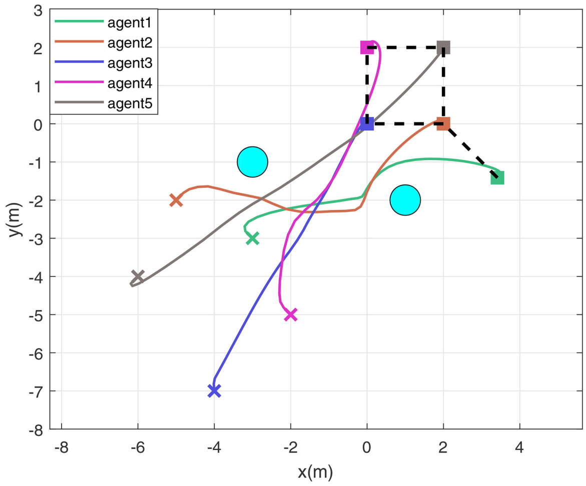

Fig. 8 *Multi-agent trajectories when $N = 2$*

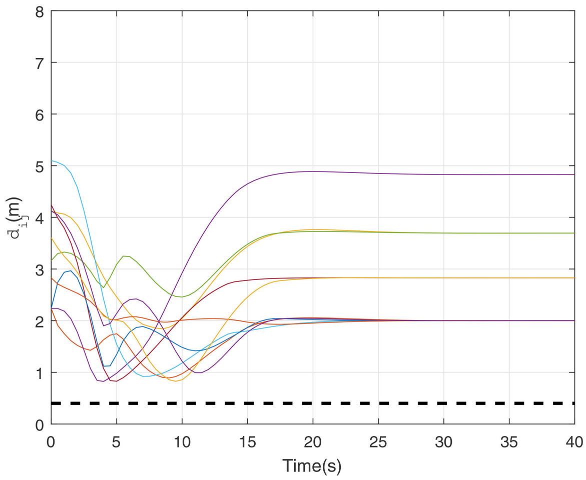

Fig. 9 *Inter-agent distances when $N = 2$*

converge to their desired positions for almost any prediction horizon.

Interesting extensions of our proposed approach may include non-linear multi-agent formation control [35], robustness against external disturbances and time-delays among agents [36], presence of dynamic obstacles. Meanwhile, event-triggered formulation of formation control for the reduction of energy consumption and communication load is another future research direction [37–39].

6 References

[1] Liang, X., Liu, Y.H., Wang, H., *et al.*: 'Leader-following formation tracking control of mobile robots without direct position measurements', *IEEE Trans. Autom. Control*, 2016, **61**, (12), pp. 4131–4137

[2] Zhao, S., Dimarogonas, D.V., Sun, Z., *et al.*: 'A general approach to coordination control of mobile agents with motion constraints', *IEEE Trans. Autom. Control*, 2018, **63**, (5), pp. 1509–1516

[3] Jin, X.: 'Fault tolerant finite-time leader–follower formation control for autonomous surface vessels with LOS range and angle constraints', *Automatica*, 2016, **68**, pp. 228–236

[4] Dong, X., Yu, B., Shi, Z., *et al.*: 'Time-varying formation control for unmanned aerial vehicles: theories and applications', *IEEE Trans. Control Sys. Technol.*, 2015, **23**, (1), pp. 340–348

[5] Du, H., Zhu, W., Wen, G., *et al.*: 'Finite-time formation control for a group of quadrotor aircraft', *Aerosp. Sci. Technol.*, 2017, **69**, pp. 609–616

[6] Schlanbusch, R., Kristiansen, R., Nicklasson, P.J.: 'Spacecraft formation reconfiguration with collision avoidance', *Automatica*, 2011, **47**, pp. 1443–1449

[7] Lee, D., Sanyal, A.K., Butcher, E.A.: 'Asymptotic tracking control for spacecraft formation flying with decentralized collision avoidance', *J. Guid. Control Dyn.*, 2015, **38**, (4), pp. 587–600

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

[9] Dong, X., Hu, Q.: 'Time-varying formation control for general linear multi-agent systems with switching directed topologies', *Automatica*, 2016, **73**, pp. 47–55

[10] Peng, Z., Wang, D., Hu, X.: 'Robust adaptive formation control of underactuated autonomous surface vehicles with uncertain dynamics', *IET Control Theory Appl.*, 2011, **5**, (12), pp. 1378–1387

[11] Consolini, L., Morbidi, F., Prattichizzo, D., *et al.*: 'Leader-follower formation control of nonholonomic mobile robots with input constraints', *Automatica*, 2008, **44**, pp. 1343–1349

[12] Yu, X., Liu, L.: 'Distributed formation control of nonholonomic vehicles subject to velocity constraints', *IEEE Trans. Ind. Electron.*, 2016, **63**, (2), pp. 1289–1298

[13] Rezaee, H., Abdollahi, F.: 'A decentralized cooperative control scheme with obstacle avoidance for a team of mobile robots', *IEEE Trans. Ind. Electron.*, 2014, **61**, (1), pp. 347–354

[14] Liu, X., Ge, S.S., Goh, C.H.: 'Formation potential field for trajectory tracking control of multi-agents in constrained space', *Int. J. Control*, 2017, **90**, (10), pp. 2137–2151

[15] Liao, F., Teo, R., Wang, J.L., *et al.*: 'Distributed formation and reconfiguration control of VTOL UAVs', *IEEE Trans. Control Syst. Technol.*, 2017, **25**, (1), pp. 270–277

[16] Wang, P., Ding, B.: 'Distributed RHC for tracking and formation of nonholonomic multi-vehicle systems', *IEEE Trans. Autom. Control*, 2014, **59**, (6), pp. 1439–1453

[17] Dai, L., Cao, Q., Xia, Y., *et al.*: 'Distributed MPC for formation of multi-agent systems with collision avoidance and obstacle avoidance', *J. Franklin Inst.*, 2017, **354**, pp. 2068–2085

[18] Hosseinzadeh Yamchi, M., Mahboobi Esfanjani, R.: 'Distributed predictive formation control of networked mobile robots subject to communication delay', *Robot. Auton. Syst.*, 2017, **91**, pp. 194–207

[19] Wang, L., Ames, A.D., Egerstedt, M.: 'Safety barrier certificates for collision-free multirobot systems', *IEEE Trans. Robot.*, 2017, **33**, (3), pp. 661–674

[20] Panagou, D.: 'A distributed feedback motion planning protocol for multiple unicycle agents of different classes', *IEEE Trans. Autom. Control*, 2017, **62**, (3), pp. 1178–1193

[21] Kuwata, Y., How, J.P.: 'Cooperative distributed robust trajectory optimization using receding horizon MILP', *IEEE Trans. Control Syst. Technol.*, 2011, **19**, (2), pp. 423–431

[22] Morgan, D., Chung, S.-J., Hadaegh, F.Y.: 'Model predictive control of swarms of spacecraft using sequential convex programming', *J. Guid., Control, Dyn.*, 2014, **37**, (6), pp. 1725–1740

[23] Primbs, J.A., Nevistić, V., Doyle, J.C.: 'A receding horizon generalization of pointwise min-norm controllers', *IEEE Trans. Autom. Control*, 2000, **45**, (5), pp. 898–909

[24] Mahmood, M., Mhaskar, P.: 'Lyapunov-based model predictive control of stochastic nonlinear systems', *Automatica*, 2012, **48**, pp. 2271–2276

[25] Zhu, B., Xia, X.: 'Lyapunov-based adaptive model predictive control for unconstrained non-linear systems with parameter uncertainties', *IET Control Theory Appl.*, 2016, **10**, (15), pp. 1937–1943

[26] Shi, X., Cao, J., Huang, W.: 'Distributed parametric consensus optimization with an application to model predictive consensus problem', *IEEE Trans. Cybern.*, 2018, **48**, (7), pp. 2024–2035

[27] Franzè, G., Gasavola, A., Famularo, D., *et al.*: 'Distributed receding horizon control of constrained networked leader–follower formations subject to packet dropouts', *IEEE Trans. Control Syst. Technol*, 2018, **26**, (5), pp. 1798–1809

[28] Zhu, B., Guo, K., Xie, L.: 'A new distributed model predictive control for unconstrained double-integrator multi-agent systems', *IEEE Trans. Autom. Control*, to be published, doi: 10.1109/TAC.2018.2819429

[29] Snape, J., van den Berg, J., Guy, S.J., *et al.*: 'The hybrid reciprocal velocity obstacle', *IEEE Trans. Robot.*, 2011, **27**, (4), pp. 696–706

[30] Rashid, A.T., Ali, A.A., Frasca, M., *et al.*: 'Multi-robot collision-free navigation based on reciprocal orientation', *Robot. Auton. Syst.*, 2017, **60**, pp. 1221–1230

[31] Picasso, B., Desiderio, D., Scattolini, R.: 'Robust stability analysis of nonlinear discrete-time systems with application to MPC', *IEEE Trans. Autom. Control*, 2012, **57**, (1), pp. 185–191

[32] Frey, G.R., Petersen, C.D., Leve, F.A., *et al.*: 'Parameter governors for coordinated control of n-spacecraft formations', *J. Guid. Control Dyn.*, 2017, **40**, (11), pp. 3014–3019

[33] Falugi, P., Kerrigan, E.C., Wyk, E.V.: 'Imperial College London Optimal Control Software User Guide (ICLOCS)', 2010. Available at http://www.ee.ic.ac.uk/ICLOCS/

[34] Wächter, A., Biegler, L.T.: 'On the implementation of an interiorpoint filter line-search algorithm for large-scale nonlinear programming', *Math. Program.*, 2006, **106**, (1), pp. 25–57

[35] Gao, Y., Dai, L., Xia, Y., *et al.*: 'Distributed model predictive control for consensus of nonlinear second-order multi-agent systems', *Int. J. Robust Nonlinear Control*, 2017, **27**, (5), pp. 830–842

[36] Sun, Z., Dai, L., Liu, K., *et al.*: 'Robust MPC for tracking constrained unicycle robots with additive disturbances', *Automatica*, 2018, **90**, pp. 172–184

[37] Sun, Z., Dai, L., Xia, Y., *et al.*: 'Event-based model predictive tracking control of nonholonomic systems with coupled input constraint and bounded disturbances', *IEEE Trans. Autom. Control*, 2018, **63**, (2), pp. 608–615

[38] Hashimoto, K., Adachi, S., Dimarogonas, D.V.: 'Distributed aperiodic model predictive controlf for multi-agent systems', *IET Control Theory Appl.*, 2015, **9**, (1), pp. 10–20

[39] Li, H., Shi, Y.: 'Event-triggered robust model predictive control of continuous time nonlinear systems', *Automatica*, 2014, **50**, pp. 1507–1513