Received: 21 January 2024 | Revised: 9 March 2024 | Accepted: 12 April 2024

DOI: 10.1002/rnc.7390

RESEARCH ARTICLE

WILEY

Convex distributed robust model predictive control for collision and obstacle avoidance

Haidi Sun | **Huahui Xie** | **Li Dai** | **Yuanqing Xia**

School of Automation, Beijing Institute of Technology, Beijing, China

Correspondence
Li Dai, School of Automation, Beijing Institute of Technology, Beijing, China.
Email: daili1887@gmail.com

Funding information
National Natural Science Foundation of China, Grant/Award Numbers: 62173036, 62122014

Abstract

The article proposes a convex distributed robust model predictive control algorithm for collision avoidance in multi-agent systems with additive disturbances, which utilizes the concept of tube model predictive control to address the disturbances. To tackle the coupled collision avoidance constraints, each agent integrates assumed nominal position and input trajectories from its neighbors, rather than relying on actual ones. Compatibility constraints are then designed using the normal vector of the constructed separating hyperplanes for collision avoidance and the residual collision avoidance margin of the optimal solutions at the last time step to restrict deviations between assumed and actual trajectories and ensure consistency among the agents. The nonconvex collision and obstacle avoidance constraints are convexified using the concept of safe sets, transforming them into time-varying closed polyhedral constraints while accounting for the impact of disturbances. Particularly, the residual collision avoidance margin is incorporated into the construction of safe sets for the satisfaction of coupled collision avoidance constraints. Consequently, the original collision avoidance optimal control problems can be efficiently and simultaneously solved for all agents using standard quadratic programming techniques. Under the design of local terminal constraint sets for each agent, the proposed algorithm ensures a rigorous analysis of robust constraint satisfaction for collision avoidance constraints, as well as an assessment of recursive feasibility and closed-loop stability. The effectiveness of the algorithm is demonstrated through an illustrative example, and simulation results validate its ability to successfully achieve collision and obstacle avoidance even in the presence of disturbances.

KEYWORDS

collision avoidance, convex optimization, distributed robust model predictive control, multi-agent uncertain systems

1 | INTRODUCTION

Formation control in multi-agent systems is a significant area of research, which has been explored in various practical domains including satellites, unmanned vehicles, multiple quadrotors, and mobile robots .[1–5] In the context of formation control, it is critical for multi-agent systems to adhere to physical constraints, which encompass limitations on states and inputs, while also considering safety constraints such as collision avoidance .[6–8] Additionally, the presence of parameter uncertainties and external disturbances from the environment is inevitable. Robust model predictive control (RMPC) has emerged as a prominent approach for addressing constraints and uncertainties simultaneously .[9–11] Among RMPC methods, tube-based model predictive control (MPC) has gained considerable attention due to its notable computational efficiency .[12–17] In the area of constrained multi-agent systems, distributed model predictive control (DMPC) has garnered significant attention due to its structural flexibility, reduced computational requirements, and minimized communication burden .[18–20] Nevertheless, various challenges persist in the field of DMPC for formation control in multi-agent systems, especially in scenarios involving collision and obstacle avoidance. These challenges include integrating the coupled formation control problem into local decision-making processes in the presence of disturbances, converting collision avoidance constraints for each agent and among agents into a convex form, formulating a computationally tractable DMPC optimization problem, and designing terminal gradients to ensure the closed-loop theoretical properties of the overall system.

Numerous studies have focused on DMPC collision avoidance methods for addressing the aforementioned challenges. Formulating obstacle avoidance and coupled collision avoidance constraints is crucial for creating tractable collision avoidance optimization problems. The collision avoidance constraints characterized by mixed integer variables, Euclidean distance and Lagrange duality parameters are commonly used in DMPC schemes. Specifically, the studies[21,22] utilize mixed integer variables to tackle collision and obstacle avoidance problems for deterministic multiple agent systems. By introducing integer variables, these approaches formulate non-smooth collision avoidance constraints and employ mixed integer quadratic programming (MIQP) solvers to solve the resulting optimization problems. The proposed MIQP-based collision avoidance schemes offer distributed solutions and respective analyses of recursive feasibility and closed-loop stability. However, it is worth noting that the inclusion of integer variables in the non-smooth collision avoidance formulation leads to overwhelming computational efforts of the MIQP-based algorithm, which poses challenges for the online implementation of MIQP under DMPC in real-time applications. Consequently, the use of MIQP for collision avoidance in disturbed multi-agent systems is relatively rare in the literature. Alternatively, other studies [23–29] address collision and obstacle avoidance constraints smoothly by restricting Euclidean distances between the agents and the obstacles to be greater than the minimum safe distance for deterministic systems. The theoretical properties, such as recursive feasibility and closed-loop stability, are extensively discussed in References 25–29, addressing the lack of theoretical properties in several studies .[23,24] Additionally, considering disturbances in practical applications, Wang et al.[30,31] extend the collision avoidance scheme using the Euclidean distance and predefined safe distance to a robust case and provides a theoretical analysis of the proposed algorithm. Duality optimization is another popular approach to formulating smooth collision avoidance constraints by using Lagrange duality. This dual optimization approach is developed in Reference 32 for deterministic systems and then extended in Reference 33 to incorporate coupled collision avoidance constraints for disturbed multi-agent systems. In Reference 33, compatibility constraints designed with assumed state trajectories are used to ensure the satisfaction of coupled collision avoidance constraints and guarantee the agreements among agents. By using a synchronous update strategy, the local MPC optimization problems for all agents are solved simultaneously in a distributed manner at each time step. Furthermore, the recursive feasibility of each local optimization problem is derived and input-to-state stability of disturbed multi-agent systems is ensured through a sufficient condition on controller parameters. Nevertheless, the formulating collision avoidance constraints by using Euclidean distance and Lagrange duality are both nonconvex, incurring a significant challenge for efficiently solving distributed collision avoidance optimization problems, particularly in the presence of disturbances. The closed polyhedral safe sets between a point and obstacle constraint sets are introduced to convexify the nonconvex obstacle avoidance constraints in Reference 34 for a controlled deterministic system. This convexification allows for the formulation of an efficient quadratic programming (QP) problem that can be solved effectively. To address the case of a disturbed system, Reference 35 extends the approach by constructing closed polyhedral safe sets between uncertain state sets and obstacle constraint sets. However, both of these proposals are not applicable to multiple agent systems, as they are designed for single-agent scenarios. To tackle collision avoidance constraints in deterministic multi-agent systems, Reference 36 introduces the concept of a safe distance set. This notion enables the convexification of collision avoidance constraints. However, the proposed strategy is based on centralized MPC, which imposes a higher computational burden when applied to multiple agent systems compared to DMPC approaches.

The aim of this article is to devise a convex DMPC algorithm that addresses the formation control problem for multiple linear disturbed systems while considering collision avoidance and obstacle avoidance constraints. Additionally, the goal is to improve the computational efficiency and ensure safety and theoretical properties during the closed-loop operation.

This proposal differs from the DMPC collision avoidance algorithms existing in literature, and the contributions of this article can be summarized as follows:

(*i*) To tackle the disturbances on system states, rigid tube MPC [12] is used to constrain the effect of uncertainties on the system states into separate robust positively invariant sets for each agent. Then, a safe box formed by a safe distance is integrated into these robust positively invariant sets to formulate collision and obstacle avoidance constraints. To deal with the coupled collision avoidance constraints, each agent utilizes assumed nominal position trajectories instead of the actual ones. As a result, the collision avoidance actually occurs between the agent's state sets integrating the safe distance and the neighbors' state sets considering the deviations of assumed and optimal nominal position trajectories.

(*ii*) Compatibility constraints are tailored to restrict the deviations between the assumed and optimal nominal states, guarantee the satisfaction of coupled collision avoidance constraints and ensure agreements among agents. Specifically, the design of compatibility constraints encompasses two key components: the residual collision avoidance margin of the optimal solutions acquired at the last time step, and the normal vector of the separating hyperplanes. These hyperplanes are established between the agent's optimal state sets, which incorporate the safe distance, and its neighbors' optimal state sets at the last time step. Then collision and obstacle avoidance constraints with uncertainties are convexified into time-varying closed polyhedral constraints by using the constructed time-varying separating hyperplanes and safe sets. In particular, the residual collision avoidance margin used in the design of compatibility constraints is incorporated into the generation of the safe sets for collision avoidance. The resulting convex DMPC collision avoidance optimal control problems, accounting for additive disturbances, are simultaneously and efficiently solved by using standard QP techniques at each time step.

(*iii*) Separating hyperplanes constructed on the middle of the Euclidean distance of the sets centered on the equilibrium points of agents and the static obstacles are utilized to design local terminal constraint sets for each agent, ensuring the collision and obstacle avoidance in the terminal area. The recursive feasibility of each local convex RMPC collision avoidance optimization problem is derived with the designed local terminal constraint sets and compatibility constraints. The closed-loop stability of the disturbed multi-agent systems is ensured with the design of local terminal constraint sets.

The following sections of this article are organized as follows: Section 2 describes the problem setting and control objective. Section 3 outlines the prototype optimal control problem, highlighting three critical issues that require addressing and briefly discussing potential solutions. In Section 4, we develop a convex distributed RMPC algorithm tailored for multi-agent systems with collision avoidance and disturbances. This section also thoroughly assesses the recursive feasibility and closed-loop stability of the proposed algorithm. Section 5 presents simulation results to validate the proposed approach's effectiveness. Finally, Section 6 provides the conclusion of this article.

Notations: The sets of reals and non-negative integers are represented by $\mathbb{R}$ and $\mathbb{N}$, respectively. Given $m, n \in \mathbb{N}$ with $m < n$, denote $\mathbb{N}_{[m:n]} := \{m, m+1, \dots, n-1, n\}$ and $\mathbb{N}_n$ for $\mathbb{N}_{[0:n]}$. A set $\mathcal{X} \subset \mathbb{R}^n$ is a PC-set if it is compact, convex, and contains the origin in its nonempty interior. A polyhedron is the intersection of a finite number of open and/or closed half-spaces and a polytope is a closed and bounded polyhedron. A set $\mathcal{X} \subset \mathbb{R}^n$ is a polytopic PC-set if it is a polytope which is a PC-set. For a nonempty set $\mathcal{X} \subset \mathbb{R}^n$, we denote $\mathbf{relint}(\mathcal{X})$ as the relative interior of $\mathcal{X}$. Given two sets $\mathcal{X} \subset \mathbb{R}^n$ and $\mathcal{Y} \subset \mathbb{R}^n$, and a vector $z \in \mathbb{R}^n$, Minkowski set addition is defined by $\mathcal{X} \oplus \mathcal{Y} := \{x + y \ : \ x \in \mathcal{X}, y \in \mathcal{Y}\}$, while $z \oplus \mathcal{X}$ denotes $\{z\} \oplus \mathcal{X}$. Given a vector $x \in \mathbb{R}^n$ and a matrix $Q \in \mathbb{R}^{n\times n}$, $Q \succ 0$ ($Q \succcurlyeq 0$) denotes a positive (semi-) definite matrix, $||x|| \triangleq \sqrt{x^\top x}$ and $||x||_Q \triangleq \sqrt{x^\top Q x}$ with $Q \succcurlyeq 0$. Besides, let $\rho(Q)$ denote the spectral radius of matrix Q. The unit ball of the Euclidean norm is expressed as $\mathcal{B}_2 = \{x \in \mathbb{R}^n \ : \ \|x\| \le 1\}$. The support function $\mathrm{h}(\mathcal{X}, \cdot)$ of a nonempty closed convex set $\mathcal{X} \subset \mathbb{R}^n$ is given, for all $y \in \mathbb{R}^n$, by

$$\mathrm{h}(\mathcal{X}, y) := \sup_x \{y^T x \ : \ x \in \mathcal{X}\}.$$

Given a matrix $G \in \mathbb{R}^{n\times m}$, $\mathrm{cone}(G)$ is a cone formed by the rows G_r with $r \in \mathbb{N}_{[1:n]}$, given by

$$\mathrm{cone}(G) = \left\{ \sum_{r\in\mathbb{N}_{[1:n]}} \varepsilon_r G_r \ : \ \forall r \in \mathbb{N}_{[1:n]},\ \varepsilon_r \ge 0 \right\}.$$

Given a set $\mathcal{X}$ and a real matrix G of compatible dimensions, the image and preimage of $\mathcal{X}$ under A are denoted by $A\mathcal{X} := \{Ax \ : \ x \in \mathcal{X}\}$ and $A^{-1}\mathcal{X} := \{x \ : \ Ax \in \mathcal{X}\}$, respectively. A scalar in algebraic expressions denotes a vector (with all its components equal to the scalar) of compatible dimensions.

2 | PROBLEM STATEMENT

2.1 | Setting

Consider a formation of N_a discrete-time linear agents with additive disturbances, which can be modeled, for $i \in \mathbb{N}_a := \{1, 2, \dots, N_a\}$, by

$$x_i^+ = A_i x_i + B_i u_i + w_i, \tag{1}$$

where $x_i \in \mathbb{R}^n$, $u_i \in \mathbb{R}^m$ and $w_i \in \mathbb{R}^n$ denote the current state, control input and additive disturbance for agent i, and x_i^+ is the successor state.

System (1) is subject to the state, input and disturbance constraints, given by

$$x_i \in \mathcal{X}_i, \ u_i \in \mathcal{U}_i \text{ and } w_i \in \mathcal{W}_i. \tag{2}$$

Assumption 1.

(*i*) The matrix pair $(A_i, B_i) \in \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times m}$ is known exactly and strictly stabilizable.
(*ii*) The state constraint set $\mathcal{X}_i$, input constraint set $\mathcal{U}_i$ and disturbance constraint set $\mathcal{W}_i$ are polytopic PC-sets in $\mathbb{R}^n$, $\mathbb{R}^m$ and $\mathbb{R}^n$, respectively, described irreducibly by

$$\mathcal{X}_i := \{x \ : \ X_i x \leq 1\}, \tag{3}$$

$$\mathcal{U}_i := \{u \ : \ U_i u \leq 1\}, \tag{4}$$

$$\mathcal{W}_i := \{w \ : \ W_i w \leq 1\}, \tag{5}$$

for known exactly matrices $X_i \in \mathbb{R}^{p_{xi}\times n}$, $U_i \in \mathbb{R}^{p_{ui}\times m}$ and $W_i \in \mathbb{R}^{p_{wi}\times n}$.

In view of the idea of rigid tube MPC ,[12] system (1) can be decomposed into two subsystems, which are nominal system and local disturbed system, given by

$$z_i^+ = A_i z_i + B_i v_i, \tag{6}$$

$$s_i^+ = (A_i + B_i K_{iS}) s_i + w_i, \tag{7}$$

where z_i and v_i represent the current state and input of the nominal system, and s_i represents the current state of the local disturbed system for the ith agent. z_i^+ and s_i^+ are the corresponding successor states for dynamics (6) and (7). The state of the ith agent, x_i, can be represented as the sum of the states of the two subsystems: $x_i = z_i + s_i$. The control input u_i is a combination of the nominal control input v_i and the feedback control term $K_{iS}s_i$: $u_i = v_i + K_{iS}s_i$, where $K_{iS} \in \mathbb{R}^{m\times n}$ is the local state feedback gain matrix. The matrix K_{iS} is designed such that the spectral radius of $(A_i + B_iK_{iS})$ is less than 1, denoted as $\rho(A_i + B_iK_{iS}) < 1$.

For the local disturbed system (7), there exists a robust positively invariant set $\mathcal{S}_i$ such that for any $s_i \in \mathcal{S}_i$ and any disturbance $w_i \in \mathcal{W}_i$, the state s_i^+ remains within $\mathcal{S}_i$. As a result, the state and control input sets for the ith agent can be represented as $z_i \oplus \mathcal{S}_i$ and $v_i \oplus K_{iS}\mathcal{S}_i$, respectively. Based on (2), the following set inclusions should be satisfied to ensure that the state and control input remain within their respective constraints:

$$z_i \oplus \mathcal{S}_i \subseteq \mathcal{X}_i, \tag{8}$$

$$v_i \oplus K_{iS}\mathcal{S}_i \subseteq \mathcal{U}_i. \tag{9}$$

Assumption 2. The robust positively invariant set $\mathcal{S}_i$ is a polytopic PC-set in $\mathbb{R}^n$, denoted by

$$\mathcal{S}_i := \{s \; : \; S_i s \leq 1\}, \tag{10}$$

for a known exactly matrix $S_i \in \mathbb{R}^{p_{si} \times n}$.

2.2 | Obstacle avoidance and collision avoidance constraints

In the environment with multiple static obstacles and agents, the hth obstacle is denoted by $\widetilde{\mathcal{O}}_h$ and the representation of the space occupied by all obstacles is denoted by $\widetilde{\mathcal{O}}$,

$$\widetilde{\mathcal{O}} := \bigcup_{h \in \mathbb{N}_o} \widetilde{\mathcal{O}}_h, \tag{11}$$

where $\mathbb{N}_o := \{1, 2, \dots, n_o\}$ denotes the index set of all obstacles. To ensure the safe and collision-free movement of all agents, a safe distance d and safe box $\mathcal{T}$ are defined, satisfying $d\mathcal{B}_2 \subseteq \mathcal{T}$. The safe box $\mathcal{T}$ acts as a buffer around each agent.

At any given time instant, the following obstacle avoidance constraint ensures that the ith agent avoids obstacles:

$$\mathbf{relint}(z_i \oplus \mathcal{S}_i \oplus \mathcal{T} \cap \widetilde{\mathcal{O}}) = \emptyset. \tag{12}$$

This constraint states that the state set of agent i, represented by $z_i \oplus \mathcal{S}_i$, combined with the safe box $\mathcal{T}$, should not intersect with any obstacles in $\widetilde{\mathcal{O}}$. In other words, the agent should stay clear of obstacles to guarantee safe movements.

Additionally, collision avoidance constraints need to be satisfied to ensure that agents do not collide with each other. For any neighboring agent j of agent i, with $j \in \mathbb{N}_a \setminus i$, the following constraint should hold:

$$\mathbf{relint}(z_i \oplus \mathcal{S}_i \oplus \mathcal{T} \cap z_j \oplus \mathcal{S}_j) = \emptyset. \tag{13}$$

This constraint states that the combination of the state set of agent i with the safe box, should not intersect with the state set of agent j. In other words, the agents should maintain a safe distance to avoid collisions with each other.

Assumption 3.

(i) For the obstacles, the hth obstacle $\widetilde{\mathcal{O}}_h$ is denoted by $\widetilde{\mathcal{O}}_h := o_h \oplus \mathcal{O}_h$, where $o_h \in \mathbb{R}^n$ and $\mathcal{O}_h \subset \mathbb{R}^n$ represent the position and shape of the hth obstacle, respectively. The set $\mathcal{O}_h$ is a polytopic PC-set and its representation is given by

$$\mathcal{O}_h := \{\chi \; : \; O_h \chi \leq 1\}, \tag{14}$$

for the matrix $O_h \in \mathbb{R}^{p_h \times n}$.

(ii) The safe box $\mathcal{T} \subset \mathbb{R}^n$ is a polytopic PC-set and determined by the given safe distance d,

$$\mathcal{T} := \{\chi \; : \; T\chi \leq 1\}, \tag{15}$$

with $T \in \mathbb{R}^{p_t \times n}$.

2.3 | Main objective

The control objective is to design a computationally efficient formation control algorithm with obstacle avoidance and collision avoidance for multiple agent systems with additive disturbances, such that

(i) all the agents reach their corresponding desired equilibrium positions z_i^d and inputs v_i^d, which satisfy $z_i^{d^+} = A_i z_i^d + B_i v_i^d$ for $i \in \mathbb{N}_a$, in a certain formation;

(*ii*) the state and control input constraints in (2) are satisfied in the presence of additive disturbances;
(*iii*) all the agents can efficiently avoid each other and every obstacle during the control process, that is, satisfying the obstacle avoidance and collision avoidance constraints (12) and (13).

To achieve this objective, a computationally efficient distributed MPC algorithm will be developed for obstacle avoidance and collision avoidance.

3 | HANDLING OF DISTRIBUTED PROTOTYPE OPTIMAL CONTROL PROBLEM

3.1 | Distributed prototype problem formulation

Towards the control objective in 2.3, we introduce a new decision variable denoted as $\mathbf{d}_i(N)$ for agent i. This variable comprises the nominal state and input variables of agent i and is represented as $\mathbf{d}_i(N) := (\mathbf{z}_i(N), \boldsymbol{v}_i(N-1))$. The nominal state and input variables, predicted over an N-step horizon, are represented by $\mathbf{z}_i(N) := \{z_i(k)\}_{k=0}^{N}$ and $\boldsymbol{v}_i(N-1) := \{v_i(k)\}_{k=0}^{N-1}$. Analogously, for agent j, $j \in \mathbb{N}_a \setminus i$, we denote $\mathbf{d}_j(N) := (\mathbf{z}_j(N), \boldsymbol{v}_j(N-1))$, with the nominal state $\mathbf{z}_j(N) := \{z_j(k)\}_{k=0}^{N}$ and nominal input $\boldsymbol{v}_j(N-1) := \{v_j(k)\}_{k=0}^{N-1}$ predicted over an N-step horizon. For a given initial state x_i of agent i and all $k \in \mathbb{N}_{N-1}$, the optimization problem prototype for agent i can be expressed as follows:

$$\min_{\mathbf{d}_i(N)} V_{iN}(x_i, z_i^d, v_i^d, \mathbf{d}_i(N)), \tag{16a}$$

subject to:

$$x_i \in z_i(0) \oplus \mathcal{S}_i, \tag{16b}$$

$$\forall k \in \mathbb{N}_{N-1}, \quad z_i(k+1) = A_i z_i(k) + B_i v_i(k), \tag{16c}$$

$$\forall k \in \mathbb{N}_{N-1}, \quad z_i(k) \oplus \mathcal{S}_i \subseteq \mathcal{X}_i, \tag{16d}$$

$$\forall k \in \mathbb{N}_{N-1}, \quad v_i(k) \oplus K_{iS}\mathcal{S}_i \subseteq \mathcal{U}_i, \tag{16e}$$

$$\forall k \in \mathbb{N}_{N-1}, \quad \textbf{relint}(z_i(k) \oplus \mathcal{C}_i \cap \widetilde{\mathcal{O}}) = \emptyset, \tag{16f}$$

$$\begin{aligned} &\forall k \in \mathbb{N}_{N-1}, \quad \forall j \in \mathbb{N}_a \setminus i, \\ &\qquad \textbf{relint}(z_i(k) \oplus \mathcal{C}_i \cap z_j(k) \oplus \mathcal{S}_j) = \emptyset, \end{aligned} \tag{16g}$$

$$z_i(N) \in \mathcal{Z}_{if}, \tag{16h}$$

where $\mathcal{C}_i := \mathcal{S}_i \oplus \mathcal{T} = \{\chi : C_i\chi \leq 1\}$ with $C_i \in \mathbb{R}^{p_{ci}\times n}$ and $\mathcal{Z}_{if}$ represents the terminal constraint set for the ith agent. The cost function $V_{iN}(\cdot)$ is defined by

$$V_{iN}(x_i, z_i^d, v_i^d, \mathbf{d}_i(N)) = \sum_{k=0}^{N-1} \ell(z_i(k), v_i(k)) + V_{if}(z_i(N)), \tag{17}$$

with the stage cost and terminal cost denoted by

$$\begin{aligned} \ell(z_i, v_i) &= ||z_i - z_i^d||_{Q_i}^2 + ||v_i - v_i^d||_{R_i}^2, \\ V_{if}(z_i) &= ||z_i - z_i^d||_{P_i}^2, \end{aligned} \tag{18}$$

where the matrices $Q_i \in \mathbb{R}^{n\times n}$, $R_i \in \mathbb{R}^{m\times m}$ and $P_i \in \mathbb{R}^{n\times n}$ are symmetric and positive definite, that is, $Q_i = Q_i^\top \succ 0$, $R_i = R_i^\top \succ 0$ and $P_i = P_i^\top \succ 0$. The properties of terminal ingredients associated with the ith agent, including terminal control law v_{if}, terminal constraint set $\mathcal{Z}_{if}$ and terminal cost V_{if}, are specified in Assumption 4.

Assumption 4.

(*i*) The terminal control law is specified by $v_{if} = v_i^d + K_{if}(z_i - z_i^d)$, with the terminal feedback gain matrix $K_{if} \in \mathbb{R}^{m\times n}$ satisfying $\rho(A_i + B_iK_{if}) < 1$.

(*ii*) The terminal constraint set $\mathcal{Z}_{if}$ is a maximal positively invariant set for dynamics (6) with $v_i = v_{if}$ subject to constraints $z_i \in \mathcal{X}_i \ominus \mathcal{S}_i$ and $v_{if} \in \mathcal{U}_i \ominus K_{iS}\mathcal{S}_i$, that is,

$$\begin{aligned}&\forall z_i \in \mathcal{Z}_{if},\ A_iz_i + B_iv_{if} \in \mathcal{Z}_{if},\ \text{and}\\ &\forall z_i \in \mathcal{Z}_{if},\ z_i \in \mathcal{X}_i \ominus \mathcal{S}_i,\ \text{and}\ v_{if} \in \mathcal{U}_i \ominus K_{iS}\mathcal{S}_i.\end{aligned} \tag{19}$$

(*iii*) The terminal constraint set $\mathcal{Z}_{if}$ is a polytopic PC-set in $\mathbb{R}^n$, denoted by

$$\mathcal{Z}_{if} = \{z_i \ :\ Z_iz_i \le 1\}, \tag{20}$$

with $Z_i \in \mathbb{R}^{p_{z_i}\times n}$.

(*iv*) The set $\mathcal{Z}_{if} \oplus \mathcal{C}_i$ is a safe region such that, $\textbf{relint}(\mathcal{Z}_{if} \oplus \mathcal{C}_i \cap \widetilde{\mathcal{O}}) = \emptyset$ and $\textbf{relint}(\mathcal{Z}_{if} \oplus \mathcal{C}_i \cap \mathcal{Z}_{jf} \oplus \mathcal{S}_j) = \emptyset$ for all $j \in \mathbb{N}_a \setminus i$.

(*v*) The terminal cost function $V_{if}(\cdot)$ satisfies the decrease condition $V_{if}(A_iz_i + B_iv_{if}) + \ell(z_i, v_{if}) \le V_{if}(z_i)$ for all $z_i \in \mathcal{Z}_{if}$, which is guaranteed by the matrix inequality

$$(A_i + B_iK_{if})^\top P_i(A_i + B_iK_{if}) + Q_i + K_{if}R_iK_{if} \le P_i. \tag{21}$$

The optimization problem prototype is formulated individually for each agent as in (16) and is solved in parallel. However, the presence of constraints (16f), (16g), and (16h) raises three key issues that require careful consideration and resolution in order to solve (16) in a distributed and efficient way. Specifically,

- The obstacle avoidance constraint (16f) and the collision avoidance constraint (16g) are intrinsically nonconvex, leading to (16) a nonconvex optimization problem, whose computational effort is overwhelming. To enhance the computational efficiency of the optimization problem, a convexification technique is employed in Section 3.2. This technique transforms the non-convex constraints into convex constraints, allowing for the use of efficient QP to solve the problem.
- In (16g), there is an unknown nominal system state $z_j(k)$ for agent j, leading to a coupled constraint. To address this issue, a decoupling operation is incorporated in Section 3.2, enabling independent optimization for each agent.
- Constraint (16h) involves the design of a proper terminal set $\mathcal{Z}_{if}$ to ensure that all agents reach their designated locations in a specific formation without colliding with obstacles or other agents. The construction of the terminal constraint set is detailed in Section 3.3, considering Assumption 4(ii)–(iv). This ensures that the terminal set satisfies the desired properties and guarantees the safe and collision-free movements of all agents to their target locations.

3.2 | Design of safe sets for obstacle and collision avoidance

In order to convexify the nonconvex obstacle and collision avoidance constraints, we first introduce the construction process of separating hyperplanes between two convex sets and then the intrinsically nonconvex obstacle and collision avoidance constraints are convexified into closed polyhedral constraints, which are regarded as safe sets.

Considering two general convex sets $\widetilde{\mathcal{M}}$ and $\widetilde{\mathcal{G}}$ in $\mathbb{R}^n$, they are given by $\widetilde{\mathcal{M}} := m \oplus \mathcal{M}$ and $\widetilde{\mathcal{G}} := g \oplus \mathcal{G}$, in which m and g are vectors in $\mathbb{R}^n$ and $\mathcal{M} \subset \mathbb{R}^n, \mathcal{G} \subset \mathbb{R}^n$ are polytopic PC-sets. The irreducible representations of $\mathcal{M}$ and $\mathcal{G}$ are $\mathcal{M} := \{\chi : M\chi \le 1\}$ with $M \in \mathbb{R}^{p_m\times n}$ and $\mathcal{G} := \{\chi : G\chi \le 1\}$ with $G \in \mathbb{R}^{p_g\times n}$. In line with the separation theorem,[37(Theorem 1.3.8),38(Theorem 11.3)] a separating hyperplane $\mathcal{H}(\widetilde{\mathcal{M}}, \widetilde{\mathcal{G}})$ can be constructed to separate the sets $\widetilde{\mathcal{M}}$ and $\widetilde{\mathcal{G}}$, provided that $\textbf{relint}(\widetilde{\mathcal{M}} \cap \widetilde{\mathcal{G}}) = \emptyset$ holds. The considered separating hyperplane is defined by

$$\mathcal{H}(\widetilde{\mathcal{M}}, \widetilde{\mathcal{G}}) := \{x \ :\ \alpha^\top(\widetilde{\mathcal{M}}, \widetilde{\mathcal{G}})x = \beta(\widetilde{\mathcal{M}}, \widetilde{\mathcal{G}})\}, \tag{22}$$

where $\alpha(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}}) \in \mathbb{R}^n$ is a nonzero vector and $\beta(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}}) \in \mathbb{R}$ is a scalar. These values can be obtained by solving a QP and a linear programming (LP) ,[35] which is detailed as follows.

The Euclidean distance of the sets $\widetilde{\mathcal{M}}$ and $\widetilde{\mathcal{G}}$ can be computed by

$$\delta(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}}) = \min_{y_m,y_g}\{\|m + y_m - g - y_g\| \,:\, y_m \in \mathcal{M}, y_g \in \mathcal{G}\}, \tag{23}$$

which can be solved by a standard QP. When the distance $\delta(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}}) > 0$, the separating hyperplane (22) can be constructed by the closest points pair $(m + y_m^0(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}}), g + y_g^0(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}}))$ from the sets $\widetilde{\mathcal{M}}$ and $\widetilde{\mathcal{G}}$, in which $y_m^0(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}})$ and $y_g^0(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}})$ are the optimal solutions of the optimization problem (23), denoted by

$$(y_m^0(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}}), y_g^0(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}})) = \arg\min_{y_m,y_g}\{\|m + y_m - g - y_g\| \,:\, y_m \in \mathcal{M}, y_g \in \mathcal{G}\}. \tag{24}$$

Then the separating hyperplane (22) can be obtained with

$$\begin{aligned} \alpha(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}}) &:= \frac{m + y_m^0(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}}) - g - y_g^0(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}})}{\delta(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}})}, \\ \beta(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}}) &:= \alpha^\top(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}})(g + y_g^0(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}})). \end{aligned} \tag{25}$$

When the distance $\delta(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}}) = 0$, the agents are just in contact with each other. In such a situation, $\alpha(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}})$ and $\beta(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}})$ cannot be obtained by (25), but can be calculated using the active constraints sets associated with the closest points, represented by

$$\begin{aligned} \mathcal{I}(m,\mathcal{M}) &= \{r \in \mathbb{N}_{[1:p_m]} \;:\; M^r y_m^0(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}}) = 1\}, \\ \mathcal{I}(m,\mathcal{G}) &= \{r \in \mathbb{N}_{[1:p_g]} \;:\; G^r y_g^0(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}}) = 1\}, \end{aligned} \tag{26}$$

where M^r is the rth row of $M \in \mathbb{R}^{p_m \times n}$ and G^r is the rth row of $G \in \mathbb{R}^{p_g \times n}$. Then, the separating hyperplane (22) for the case of $\delta(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}}) = 0$ is constructed by using convex cone properties,[35(Proposition 3)]

$$\begin{aligned} &\alpha^\top(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}}) \in \mathrm{cone}(\hat{M}) \cap -\mathrm{cone}(\hat{G}), \text{ and} \\ &\beta(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}}) := \alpha^\top(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}})(g + y_g^0(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}})), \end{aligned} \tag{27}$$

where $\hat{M}$ is a matrix with its rows M^r, $r \in \mathcal{I}(m,\mathcal{M})$ and $\hat{G}$ is a matrix with its rows G^r, $r \in \mathcal{I}(m,\mathcal{G})$. The values $\alpha^\top(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}})$ and $\beta(\widetilde{\mathcal{M}},\widetilde{\mathcal{G}})$ in (27) can be obtained by solving an LP problem using the approach described in the literature [35(Remark 1)].

Following the guidelines for constructing separating hyperplanes detailed above, we provide the expressions of separating hyperplanes for obstacle avoidance and collision avoidance. Then, we form the corresponding safe sets, which are closed polyhedral constraints.

3.2.1 | Construction of safe sets for obstacle avoidance

For any given feasible trajectory $\bar{\mathbf{d}}_i(N) := (\{\bar{z}_i(k)\}_{k=0}^{N}, \{\bar{v}_i(k)\}_{k=0}^{N-1})$ satisfying constraints (16b)–(16h), and any point $\bar{z}_i(k) \in \mathbb{R}^n$ in the feasible trajectory $\bar{\mathbf{d}}_i(N)$ for the ith agent, a separating hyperplane $\mathcal{H}(\bar{z}_i(k) \oplus C_i, o_h \oplus \mathcal{O}_h)$ can be constructed to separate the set $\bar{z}_i(k) \oplus C_i$ and the obstacle $o_h \oplus \mathcal{O}_h$, provided that $\mathbf{relint}(\bar{z}_i(k) \oplus C_i \cap o_h \oplus \mathcal{O}_h) = \emptyset$ holds. For typographical convenience, let $\widetilde{C}_i(k) := \bar{z}_i(k) \oplus C_i$. When $\mathbf{relint}(\widetilde{C}_i(k) \cap \widetilde{\mathcal{O}}_h) = \emptyset$ is satisfied, the separating hyperplane $\mathcal{H}(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h)$ for obstacle avoidance can be generated in accordance with (22)–(27).

In the context of safe sets, as introduced in Reference 35, the constructed separating hyperplanes form the following safe half-space

$$\mathcal{F}(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h) := \{x_i(k) \,:\, \alpha^\top(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h)x_i(k) \geq \beta(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h)\}, \tag{28}$$

which is equivalent to

$$\mathcal{L}(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h) := \{z_i(k) : \alpha^\top(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h)z_i(k) \geq \varrho(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h)\}, \tag{29}$$

where $\varrho(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h) = \beta(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h) + \vartheta(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h)$ with $\vartheta(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h) := \mathrm{h}(C_i, -\alpha(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h))$.

For all obstacles, the intersection of constructed safe half-spaces (29) forms the safe set for agent i, which is a safe region, ensuring obstacle avoidance for agent i at time instant k. Such a safe set with respect to the nominal state $z_i(k)$ is given by

$$\mathcal{R}_o(\widetilde{C}_i(k), \widetilde{\mathcal{O}}) := \bigcap_{h\in\mathbb{N}_o} \mathcal{L}(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h) = \{z_i(k) : R_o(\widetilde{C}_i(k), \widetilde{\mathcal{O}})z_i(k) \geq r_o(\widetilde{C}_i(k), \widetilde{\mathcal{O}})\}, \tag{30}$$

where $R_o(\widetilde{C}_i(k), \widetilde{\mathcal{O}}) \in \mathbb{R}^{p_{oi}\times n}$ is a matrix and $r_o(\widetilde{C}_i(k), \widetilde{\mathcal{O}}) \in \mathbb{R}^{p_{oi}}$ is a vector. The obstacle avoidance constraint (16f) can be ensured by $\forall k \in \mathbb{N}_{N-1}, z_i(k) \in \mathcal{R}_o(\widetilde{C}_i(k), \widetilde{\mathcal{O}})$.

3.2.2 | Construction of safe sets for collision avoidance

To process the coupled collision avoidance constraints (16g), an assumed state $\hat{z}_j(k)$ is introduced to be shared among the agents instead of transmitting the actual one $z_j(k)$. The assumed trajectory for agent j is given by $\hat{\mathbf{d}}_j(N) := (\{\hat{z}_j(k)\}_{k=0}^{N}, \{\hat{v}_j(k)\}_{k=0}^{N-1})$. It is worth noting that the incorporation of assumed states enables optimization in the distributed setting, while deviations between the actual predictive trajectory and the assumed trajectory will be constrained by the compatibility constraints, guaranteeing the agreements among agents.

For any given point $\bar{z}_i(k) \in \mathbb{R}^n$ in the feasible trajectory of agent i and assumed state $\hat{z}_j(k) \in \mathbb{R}^n$ in the assumed trajectory of agent j, a separating hyperplane $\mathcal{H}(\bar{z}_i(k) \oplus C_i, \hat{z}_j(k) \oplus \mathcal{S}_j)$ can be constructed to separate the sets $\bar{z}_i(k) \oplus C_i$ and $\hat{z}_j(k) \oplus \mathcal{S}_j$, when the collision avoidance constraints $\mathbf{relint}(\bar{z}_i(k) \oplus C_i \cap \hat{z}_j(k) \oplus \mathcal{S}_j) = \emptyset$ is satisfied. For typographical convenience, let $\widetilde{C}_i(k) := \bar{z}_i(k) \oplus C_i,\ \widetilde{\mathcal{S}}_j(k) := \hat{z}_j(k) \oplus \mathcal{S}_j$. Then the separating hyperplane $\mathcal{H}(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k))$ for collision avoidance can be generated to form the following assumed safe half-space,

$$\mathcal{F}(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k)) := \{x_i(k) \ : \ \alpha^\top(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k))x_i(k) \geq \beta(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k))\}, \tag{31}$$

which is equivalent to

$$\mathcal{N}(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k)) := \{z_i(k) : \alpha^\top(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k))z_i(k) \geq \mu(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k))\}, \tag{32}$$

where $\mu(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k)) = \beta(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k)) + \zeta(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k))$, with $\zeta(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k)) := \mathrm{h}(C_i, -\alpha^\top(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k)))$.

In order to account for the deviation between the actual state and the assumed state of agents, an additional compatibility constraint is introduced. This constraint serves as a representation of the agreement among the agents' behaviors and limits the extent of deviation from the actual state, promoting consistency and coherence in the agents' actions. By incorporating this constraint into the optimization problem, the agents are constrained to adhere to their designated plans while allowing for a certain level of flexibility to accommodate deviations. Define $\varepsilon(i,j,k) = \frac{\alpha^\top(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k))\hat{z}_i(k) - \mu(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k))}{2}$ and $\epsilon(i,k) = \min_{j\in\mathbb{N}_a\setminus i} \varepsilon(i,j,k)$. The compatibility constraint is represented by

$$\mathcal{D}_z(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k)) := \{z_i(k) : \alpha^\top(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k))(z_i(k) - \hat{z}_i(k)) \leq \epsilon(i,k)\}. \tag{33}$$

The collision avoidance constraint (16g) can be ensured by the safe half-space $\mathcal{N}_\epsilon(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k))$, which is represented by, for all $k \in \mathbb{N}_{N-1}$ and $j \in \mathbb{N}_a \setminus i$,

$$\mathcal{N}_\epsilon(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k)) := \{z_i(k) : \alpha^\top(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k))z_i(k) \geq \mu(\widetilde{C}_i(k), \widetilde{\mathcal{S}}_j(k)) + \varepsilon(i,j,k)\}. \tag{34}$$

For all agents $j \in \mathbb{N}_a \setminus i$, their occupied space at time instant k is denoted by $\widetilde{\mathcal{S}}(k) := \bigcup_{j\in\mathbb{N}_a\setminus i} \widetilde{\mathcal{S}}_j(k)$. The safe set for agent i to ensure collision avoidance at time instant k is obtained by the intersection of safe half-spaces (34), which is expressed as

$$\mathcal{R}_s(\widetilde{C}_i(k), \widetilde{S}(k)) := \bigcap_{j\in\mathbb{N}_a\setminus i} \mathcal{N}_\epsilon(\widetilde{C}_i(k), \widetilde{S}_j(k)) = \{z_i(k) : R_s(\widetilde{C}_i(k), \widetilde{S}(k))z_i(k) \geq r_s(\widetilde{C}_i(k), \widetilde{S}(k))\}, \tag{35}$$

where $R_s(\widetilde{C}_i(k), \widetilde{S}(k)) \in \mathbb{R}^{p_{ri}\times n}$ is a matrix and $r_s(\widetilde{C}_i(k), \widetilde{S}(k)) \in \mathbb{R}^{p_{ri}}$ is a vector. The collision avoidance constraint (16g) can be guaranteed by $\forall k \in \mathbb{N}_{N-1}, z_i(k) \in \mathcal{R}_s(\widetilde{C}_i(k), \widetilde{S}(k))$.

3.3 | Construction of terminal constraint set

To construct the terminal constraint set $\mathcal{Z}_{if}$ for all $i \in \mathbb{N}_a$, we calculate the separating hyperplanes and safe sets formed between the sets of $z_i^d \oplus C_i$ and $\widetilde{\mathcal{O}}_h, h \in \mathbb{N}_o$, as well as between the sets of $z_i^d \oplus C_i$ and $z_j^d \oplus \mathcal{S}_j, j \in \mathbb{N}_a \setminus i$ for ensuring Assumption 4(iv). For typographical convenience, we define $C_i^d := z_i^d \oplus C_i$, $\mathcal{S}_j^d := z_j^d \oplus \mathcal{S}_j$. The separating hyperplanes $\mathcal{H}(C_i^d, \widetilde{\mathcal{O}}_h)$ between the sets C_i^d and $\widetilde{\mathcal{O}}_h$, and $\mathcal{H}(C_i^d, \mathcal{S}_j^d)$ between the sets C_i^d and $\mathcal{S}_j^d$, can be obtained using the guidelines for constructing the separating hyperplanes described in Section 3.2. These separating hyperplanes, $\mathcal{H}(C_i^d, \widetilde{\mathcal{O}}_h)$ and $\mathcal{H}(C_i^d, \mathcal{S}_j^d)$, serve as boundaries or constraints in the construction of the terminal constraint set $\mathcal{Z}_{if}$, and their expressions are

$$\begin{aligned}\mathcal{H}(C_i^d, \widetilde{\mathcal{O}}_h) &:= \{x \ : \ \alpha^\top(C_i^d, \widetilde{\mathcal{O}}_h)x = \beta(C_i^d, \widetilde{\mathcal{O}}_h)\},\\ \mathcal{H}(C_i^d, \mathcal{S}_j^d) &:= \{x \ : \ \alpha^\top(C_i^d, \mathcal{S}_j^d)x = \beta(C_i^d, \mathcal{S}_j^d)\}.\end{aligned} \tag{36}$$

It is worth noting that the computations of $\beta(C_i^d, \widetilde{\mathcal{O}}_h)$ and $\beta(C_i^d, \mathcal{S}_j^d)$ are not exactly the same with equations in (25) and (27). Instead of constructing the separating hyperplanes $\mathcal{H}(C_i^d, \widetilde{\mathcal{O}}_h)$ and $\mathcal{H}(C_i^d, \mathcal{S}_j^d)$ on the closest points $o_h + y_o^0(C_i^d, \widetilde{\mathcal{O}}_h)$ and $z_j^d + y_s^0(C_i^d, \mathcal{S}_j^d)$ respectively, the midpoints of the closest point pairs $z_i^d + y_c^0(C_i^d, \widetilde{\mathcal{O}}_h)$ and $o_h + y_o^0(C_i^d, \widetilde{\mathcal{O}}_h)$ as well as $z_i^d + y_c^0(C_i^d, \mathcal{S}_j^d)$ and $z_j^d + y_s^0(C_i^d, \mathcal{S}_j^d)$ are utilized to calculate the corresponding $\beta(C_i^d, \widetilde{\mathcal{O}}_h)$ and $\beta(C_i^d, \mathcal{S}_j^d)$, given by

$$\begin{aligned}\beta(C_i^d, \widetilde{\mathcal{O}}_h) &:= \alpha^\top(C_i^d, \widetilde{\mathcal{O}}_h)\frac{z_i^d + y_c^0(C_i^d, \widetilde{\mathcal{O}}_h) + o_h + y_o^0(C_i^d, \widetilde{\mathcal{O}}_h)}{2},\\ \beta(C_i^d, \mathcal{S}_j^d) &:= \alpha^\top(C_i^d, \mathcal{S}_j^d)\frac{z_i^d + y_c^0(C_i^d, \mathcal{S}_j^d) + z_j^d + y_s^0(C_i^d, \mathcal{S}_j^d)}{2}.\end{aligned} \tag{37}$$

It means that the separating hyperplanes $\mathcal{H}(C_i^d, \widetilde{\mathcal{O}}_h)$ and $\mathcal{H}(C_i^d, \mathcal{S}_j^d)$ are constructed on the middle of the Euclidean distance between the sets $C_i^d, \widetilde{\mathcal{O}}_h$ and the sets $C_i^d, \mathcal{S}_j^d$, respectively. This design ensures relatively uniform terminal constraint sets in size for each agent and gets Assumption 4(iv) satisfied.

In light of Assumptions 4(ii),(iv), the terminal constraint set $\mathcal{Z}_{if}$ is a maximal positively invariant set for the terminal z-dynamics $z_i^+ = A_i z_i + B_i(v_i^d + K_{if}(z_i - z_i^d))$ and constraint $z_i \in \mathcal{Z}_i(0)$, with

$$\begin{aligned}\mathcal{Z}_i(0) := &\{z_i : \forall h \in \mathbb{N}_o, \ -\alpha^\top(C_i^d, \widetilde{\mathcal{O}}_h)z_i \leq -\beta(C_i^d, \widetilde{\mathcal{O}}_h) + \lambda_{(i,h)}\}\\ &\bigcap\{z_i : \forall j \in \mathbb{N}_a \setminus i, \ -\alpha^\top(C_i^d, \mathcal{S}_j^d)z_i \leq -\beta(C_i^d, \mathcal{S}_j^d) + \eta_{(i,j)}\}\\ &\bigcap\{z_i : X_i z_i \leq 1 - \gamma_i\}\\ &\bigcap\{z_i \ : \ U_i(v_i^d + K_{if}(z_i - z_i^d)) \leq 1 - \sigma_i\},\end{aligned} \tag{38}$$

where $\lambda_{(i,h)} \in \mathbb{R}$ and $\eta_{(i,j)} \in \mathbb{R}$ are scalars, and $\gamma_i \in \mathbb{R}^{p_{xi}}$ and $\sigma_i \in \mathbb{R}^{p_{ui}}$ are vectors, given by

$$\begin{aligned}\lambda_{(i,h)} &:= \mathrm{h}(C_i, \alpha^\top(C_i^d, \widetilde{\mathcal{O}}_h)),\\ \eta_{(i,j)} &:= \mathrm{h}(C_i, \alpha^\top(C_i^d, \mathcal{S}_j^d)),\\ \gamma_i &:= \mathrm{h}(\mathcal{S}_i, X_i),\\ \sigma_i &:= \mathrm{h}(\mathcal{S}_i, U_i K_{iS}).\end{aligned}$$

The constraint sets $\{z_i : X_i z_i \le 1-\gamma_i\}$ and $\{z_i : U_i(v_i^d + K_{if}(z_i - z_i^d)) \le 1-\sigma_i\}$ represent the state and input constraints under terminal control law, satisfying Assumption 4(ii). The constraint set $\{z_i : \forall h \in \mathbb{N}_o,\ -\alpha^\top(C_i^d, \widetilde{\mathcal{O}}_h) z_i \le -\beta(C_i^d, \widetilde{\mathcal{O}}_h) + \lambda_{(i,h)}\}$ represents the constraints $z_i \oplus C_i \subseteq \mathcal{F}(C_i^d, \widetilde{\mathcal{O}}_h)$, for which $\mathcal{F}(C_i^d, \widetilde{\mathcal{O}}_h)$ is a safe set constructed between the set $z_i^d \oplus C_i$ and the obstacle $\widetilde{\mathcal{O}}_h$. The constraint set $\{z_i : \forall j \in \mathbb{N}_a \setminus i,\ -\alpha^\top(C_i^d, \mathcal{S}_j^d) z_i \le -\beta(C_i^d, \mathcal{S}_j^d) + \eta_{(i,j)}\}$ represents the constraints $z_i \oplus C_i \subseteq \mathcal{F}(C_i^d, \mathcal{S}_j^d)$, for which $\mathcal{F}(C_i^d, \mathcal{S}_j^d)$ is a safe set constructed between the sets $z_i^d \oplus C_i$ and $z_j^d \oplus \mathcal{S}_j$. The safe sets $\mathcal{F}(C_i^d, \widetilde{\mathcal{O}}_h)$ and $\mathcal{F}(C_i^d, \mathcal{S}_j^d)$ ensure that $\forall x_i \in \mathcal{Z}_i(0) \oplus C_i$, $x_i \notin \widetilde{\mathcal{O}}_h$ for all $h \in \mathbb{N}_o$ and $x_i \notin z_j^d \oplus \mathcal{S}_j$ for all $j \in \mathbb{N}_a \setminus i$.

For computing $\mathcal{Z}_{if}$, we resort to the maximal positively invariant set $\mathcal{E}_{if}$ with $\mathcal{Z}_{if} = z_i^d \oplus \mathcal{E}_{if}$ for the terminal e-dynamics $e_i^+ = (A_i + B_i K_{if}) e_i$ with $z_i = e_i + z_i^d$ and constraint $e_i \in \mathcal{E}_i(0)$, with

$$\begin{aligned}\mathcal{E}_i(0) := &\{e_i : \forall h \in \mathbb{N}_o, -\alpha^\top(C_i^d, \widetilde{\mathcal{O}}_h) e_i \le -\beta(C_i^d, \widetilde{\mathcal{O}}_h) + T_\lambda(i,h)\}\\ &\bigcap \{e_i : \forall j \in \mathbb{N}_a \setminus i,\ -\alpha^\top(C_i^d, \mathcal{S}_j^d) e_i \le -\beta(C_i^d, \mathcal{S}_j^d) + T_\eta(i,j)\}\\ &\bigcap \{e_i : X_i e_i \le 1 - \gamma_i - X_i z_i^d\}\\ &\bigcap \{e_i : U_i K_{if} e_i \le 1 - \sigma_i - U_i v_i^d\},\end{aligned} \tag{39}$$

where $T_\lambda(i,h) = \lambda_{(i,h)} + \alpha^\top(C_i^d, \widetilde{\mathcal{O}}_h) z_i^d$ and $T_\eta(i,j) = \eta_{(i,j)} + \alpha^\top(C_i^d, \mathcal{S}_j^d) z_i^d$. In view of the approach of set iteration for obtaining the maximal positively invariant set ,[39] the set $\mathcal{E}_{if} = \mathcal{E}_i(\infty)$ can be worked out by

$$\mathcal{E}_i(k+1) = ((A_i + B_i K_{if})^{-1} \mathcal{E}_i(k)) \bigcap \mathcal{E}_i(0). \tag{40}$$

Consequently, the terminal constraint set $\mathcal{Z}_{if}$ can be obtained by $\mathcal{Z}_{if} = z_i^d \oplus \mathcal{E}_{if}$.

4 | DISTRIBUTED RMPC ALGORITHM FOR COLLISION AVOIDANCE

4.1 | Algorithm design

After the convexification of obstacle and collision avoidance constraints by utilizing the constructed safe sets, and the design of compatibility constraints as well as terminal constraints, for any given initial state $x_i \in \mathbb{X}_{iN}$ and all $k \in \mathbb{N}_{N-1}$, a convex optimization problem **OCP** can be formulated for agent i. Define $z_i = e_i + z_i^d, v_i = \pi_i + v_i^d, \bar{z}_i = \bar{e}_i + z_i^d, \bar{v}_i = \bar{\pi}_i + v_i^d, \hat{z}_i = \hat{e}_i + z_i^d, \hat{v}_i = \hat{\pi}_i + v_i^d$, and let the decision variable $\boldsymbol{\xi}_i(N) := (\mathbf{e}_i(N), \boldsymbol{\pi}_i(N-1))$ with the state sequence $\mathbf{e}_i(N) := \{e_i(k)\}_{k=0}^N$ and input sequence $\boldsymbol{\pi}_i(N-1) := \{\pi_i(k)\}_{k=0}^{N-1}$, the feasible trajectory $\bar{\boldsymbol{\xi}}_i(N) := (\bar{\mathbf{e}}_i(N), \bar{\boldsymbol{\pi}}_i(N-1))$ with feasible state sequence $\bar{\mathbf{e}}_i(N) := \{\bar{e}_i(k)\}_{k=0}^N$ and feasible input sequence $\bar{\boldsymbol{\pi}}_i(N-1) := \{\bar{\pi}_i(k)\}_{k=0}^{N-1}$ as well as the assumed trajectory $\hat{\boldsymbol{\xi}}_i(N) := (\hat{\mathbf{e}}_i(N), \hat{\boldsymbol{\pi}}_i(N-1))$, with the assumed state sequence $\hat{\mathbf{e}}_i(N) := \{\hat{e}_i(k)\}_{k=0}^N$ and assumed input sequence $\hat{\boldsymbol{\pi}}_i(N-1) := \{\hat{\pi}_i(k)\}_{k=0}^{N-1}$ predicted over an N-step horizon.

OCP :

$$\min_{\boldsymbol{\xi}_i(N)} V_{iN}(x_i, z_i^d, v_i^d, \boldsymbol{\xi}_i(N), \bar{\boldsymbol{\xi}}_i(N), \hat{\boldsymbol{\xi}}_j(N)), \tag{41a}$$

subject to:

$$x_i - z_i^d \in e_i(0) \oplus \mathcal{S}_i, \tag{41b}$$

$$\forall k \in \mathbb{N}_{N-1},\quad e_i(k+1) = A_i e_i(k) + B_i \pi_i(k), \tag{41c}$$

$$\forall k \in \mathbb{N}_{N-1},\quad e_i(k) + z_i^d \in \mathcal{X}_i \ominus \mathcal{S}_i, \tag{41d}$$

$$\forall k \in \mathbb{N}_{N-1},\quad \pi_i(k) + v_i^d \in \mathcal{U}_i \ominus K_{iS} \mathcal{S}_i, \tag{41e}$$

$$\forall k \in \mathbb{N}_{N-1},\quad e_i(k) + z_i^d \in \mathcal{R}_o(\widetilde{C}_i(k), \widetilde{\mathcal{O}}), \tag{41f}$$

$$\forall k \in \mathbb{N}_{N-1},\quad e_i(k) + z_i^d \in \mathcal{R}_s(\widetilde{C}_i(k), \widetilde{\mathcal{S}}(k)), \tag{41g}$$

$$\forall k \in \mathbb{N}_{N-1}, \quad e_i(k) \in \mathcal{D}_e(\widetilde{C}_i(k), \widetilde{S}_j(k)), \tag{41h}$$

$$e_i(N) + z_i^d \in \mathcal{Z}_{if}. \tag{41i}$$

The compatibility constraints (41h) is equivalent to (33) and $\mathcal{D}_e(\widetilde{C}_i(k), \widetilde{S}_j(k))$ is specified by

$$\mathcal{D}_e(\widetilde{C}_i(k), \widetilde{S}_j(k)) := \{e_i(k) : \alpha^\top(\widetilde{C}_i(k), \widetilde{S}_j(k))(e_i(k) - \hat{e}_i(k)) \le \epsilon(i,k)\}. \tag{42}$$

The cost function $V_{iN}(\cdot)$ is given by

$$V_{iN}(x_i, z_i^d, v_i^d, \xi_i(N), \overline{\xi}_i(N), \hat{\xi}_j(N)) = \sum_{k=0}^{N-1}(||e_i(k)||_{Q_i}^2 + ||\pi_i(k)||_{R_i}^2) + ||e_i(N)||_{P_i}^2. \tag{43}$$

The set $\mathbb{X}_{iN}$ is the N-step controllable set for agent i, that is the set of the ith system states for which **OCP** is feasible.

$$\mathbb{X}_{iN} := \{x_i : \mathbb{D}_{iN}(x_i, z_i^d, v_i^d, \xi_i(N), \overline{\xi}_i(N), \hat{\xi}_j(N)) \neq \emptyset\}, \tag{44}$$

where $\mathbb{D}_{iN}(x_i, z_i^d, v_i^d, \xi_i(N), \overline{\xi}_i(N), \hat{\xi}_j(N))$ denotes the set of feasible solutions for **OCP**, that is,

$$\mathbb{D}_{iN}(x_i, z_i^d, v_i^d, \xi_i(N), \overline{\xi}_i(N), \hat{\xi}_j(N)) := \{\xi_i(N) : \text{(41b)–(41i) hold}\}. \tag{45}$$

It is clear that the formulated **OCP** is a standard QP, which can be efficiently solved. The optimal solutions for **OCP** are denoted as $\xi_i^0(N) := (\{e_i^0(k)\}_{k=0}^N, \{\pi_i^0(k)\}_{k=0}^{N-1})$. The control law $\kappa_{iN}^0(\cdot)$ for the ith agent system (1) is then given by

$$\kappa_N^0(x_i, z_i^d, v_i^d, \xi_i(N), \overline{\xi}_i(N), \hat{\xi}_j(N)) := \pi_i^0(0) + v_i^d + K_{iS}(x_i - z_i^d - e_i^0(0)), \tag{46}$$

where $e_i^0(0)$ and $\pi_i^0(0)$ are the first elements of the optimal state and input trajectories from the optimal solutions $\xi_i^0(N)$.

To simplify the expression, we use κ_N^0 to denote $\kappa_N^0(x_i, z_i^d, v_i^d, \xi_i(N), \overline{\xi}_i(N), \hat{\xi}_j(N))$. Thus the ith agent system is then given by

$$x_i^+ = A_i x_i + B_i \kappa_N^0 + w_i. \tag{47}$$

The initial state x_i^+ in (41b) can be obtained by (47). Then the feasible trajectory $\overline{\xi}_i^+(N)$ and assumed trajectory $\hat{\xi}_i^+(N)$ at x_i^+ are set with the optimal trajectories $\xi_i^0(N)$, that is, $\overline{\xi}_i^+(N) = (\{\overline{e}_i^+(k)\}_{k=0}^N, \{\overline{\pi}_i^+(k)\}_{k=0}^{N-1})$, and $\hat{\xi}_i^+(N) = (\{\hat{e}_i^+(k)\}_{k=0}^N, \{\hat{\pi}_i^+(k)\}_{k=0}^{N-1})$, where $\forall\, k \in \mathbb{N}_{N-1}$, $\overline{e}_i^+(k) = \hat{e}_i^+(k) = e_i^0(k+1)$ and $\overline{e}_i^+(N) = \hat{e}_i^+(N) = (A_i + B_i K_{if})e_i^0(N)$, and $\forall\, k \in \mathbb{N}_{N-2}$, $\overline{\pi}_i^+(k) = \hat{\pi}_i^+(k) = \pi_i^0(k+1)$ and $\overline{\pi}_i^+(N-1) = \hat{\pi}_i^+(N-1) = K_{if} e_i^0(N)$.

The implementation process of the entire algorithm is summarized in Algorithm 1. In the online implementation of Algorithm 1, a set of computational procedures is executed, involving a maximum of three QPs at steps 2, 5 and 8 as well as two LPs at steps 3 and 6. This strategic design ensures a remarkable level of computational efficiency in addressing collision avoidance and obstacle avoidance with designated formation.

4.2 | Closed-loop property analysis

In this section, we analyze the recursive feasibility and robust closed-loop stability of the multi-agent systems.

Theorem 1. *Under Assumptions 1–4, if there exists a feasible solution to* **OCP** *at the initial time instant* $k = 0$, *then Algorithm 1 is recursively feasible at all subsequent time instances.*

Proof. Following the theoretical proof of the recursive feasibility in Reference 35, we observe that, for any $x_i \in \mathbb{X}_{iN}$ and any feasible trajectory $\overline{\xi}_i(N)$ and assumed trajectory $\hat{\xi}_j(N)$, **OCP** is feasible. Naturally, the optimal decision variable $\xi_i^0(N)$ is obtained by solving **OCP**. Since $\xi_i^0(N) \in \mathbb{D}_N(x_i, z_i^d, v_i^d, \xi_i(N), \overline{\xi}_i(N), \hat{\xi}_j(N))$, it

Algorithm 1. Convex distributed RMPC for collision avoidance

Offline: for each agent $i, i \in \mathbb{N}_a$

1: Obtain local state feedback gain matrix K_{iS} satisfying $\rho(A_i + B_i K_{iS}) < 1$.
2: Obtain the terminal state feedback gain matrix K_{if} satisfying Assumption 4(i).
3: Obtain the terminal cost matrix P_i satisfying Assumption 4(v).
4: Compute the robust positively invariant sets $\mathcal{S}_i$ according to the approach in Reference[40].
5: Obtain the terminal set $\mathcal{Z}_{if}$ with the method in Section 3.3.

Online: for each agent $i, i \in \mathbb{N}_a$

1: Initialize the feasible trajectory $\bar{\xi}_i(N)$ at $x_i \in \mathbb{X}_{iN}$ and assumed trajectory $\hat{\xi}_j(N)$ at $x_j \in \mathbb{X}_{jN}$ for all $j \in \mathbb{N}_a \setminus i$ by the approach in Appendix 5, which ensures that, for all $k \in \mathbb{N}_{N-1}$, $\forall h \in \mathbb{N}_o$, $\mathbf{relint}(\widetilde{C}_i(k) \cap \widetilde{\mathcal{O}}_h(k)) = \emptyset$ and $\forall j \in \mathbb{N}_a \setminus i$, $\mathbf{relint}(\widetilde{C}_i(k) \cap \widetilde{S}_j(k)) = \emptyset$.
2: For all $k \in \mathbb{N}_{N-1}$ and all $h \in \mathbb{N}_o$, calculate the distance $\delta(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h(k))$ and $(y_c^0(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h(k)), y_o^0(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h(k)))$ by (23) and (24), respectively.
3: For all $k \in \mathbb{N}_{N-1}$ and all $h \in \mathbb{N}_o$, construct the separating hyperplane $\mathcal{H}(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h(k))$ by (22) .
  If $\delta(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h(k)) > 0$
    Calculate $\alpha(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h(k))$ and $\beta(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h(k))$ by (25).
  Else
    Calculate $\alpha(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h(k))$ and $\beta(\widetilde{C}_i(k), \widetilde{\mathcal{O}}_h(k))$ by (27).
  end
4: For all $k \in \mathbb{N}_{N-1}$, form the safe set $\mathcal{R}_o(\widetilde{C}_i(k), \widetilde{\mathcal{O}})$ for obstacle avoidance with (28)–(30).
5: For all $k \in \mathbb{N}_{N-1}$ and all $j \in \mathbb{N}_a \setminus i$, calculate the distance $\delta(\widetilde{C}_i(k), \widetilde{S}_j(k))$ and $(y_c^0(\widetilde{C}_i(k), \widetilde{S}_j(k)), y_s^0(\widetilde{C}_i(k), \widetilde{S}_j(k)))$ by (23) and (24), respectively.
6: For all $k \in \mathbb{N}_{N-1}$ and all $j \in \mathbb{N}_a \setminus i$, construct the separating hyperplane $\mathcal{H}(\widetilde{C}_i(k), \widetilde{S}_j(k))$ by (22).
  If $\delta(\widetilde{C}_i(k), \widetilde{S}_j(k)) > 0$
    Calculate $\alpha(\widetilde{C}_i(k), \widetilde{S}_j(k))$ and $\beta(\widetilde{C}_i(k), \widetilde{S}_j(k))$ by (25).
  Else
    Calculate $\alpha(\widetilde{C}_i(k), \widetilde{S}_j(k))$ and $\beta(\widetilde{C}_i(k), \widetilde{S}_j(k))$ by (27).
  end
7: For all $k \in \mathbb{N}_{N-1}$, form the safe set $\mathcal{R}_s(\widetilde{C}_i(k), \widetilde{S}(k))$ for collision avoidance by (31)–(35).
8: Obtain $\xi_i^0(N)$ by solving **OCP** at x_i.
9: Obtain x_i^+ by (47).
10: Construct $\bar{\xi}_i^+(N) = (\{\bar{e}_i^+(k)\}_{k=0}^{N}, \{\bar{\pi}_i^+(k)\}_{k=0}^{N-1})$, and $\hat{\xi}_i^+(N) = (\{\hat{e}_i^+(k)\}_{k=0}^{N}, \{\hat{\pi}_i^+(k)\}_{k=0}^{N-1})$.
    For all $k \in \mathbb{N}_{N-1}$, $\bar{e}_i^+(k) = \hat{e}_i^+(k) = e_i^0(k+1)$ and $\bar{e}_i^+(N) = \hat{e}_i^+(N) = (A_i + B_i K_{if}) e_i^0(N)$.
    For all $k \in \mathbb{N}_{N-2}$, $\bar{\pi}_i^+(k) = \hat{\pi}_i^+(k) = \pi_i^0(k+1)$ and $\bar{\pi}_i^+(N-1) = \hat{\pi}_i^+(N-1) = K_{if} e_i^0(N)$.
11: Go to **Online** step 2.

follows directly that (41b)–(41i) are satisfied. The successor state for agent i is given by $x_i^+ = A_i x_i + B_i \kappa_N^0 + w_i$, $w_i \in \mathcal{W}_i$. At x_i^+, the feasible and assumed trajectories $\overline{\xi}_i^+(N)$ and $\hat{\xi}_i^+(N)$ are constructed in Section 4.1. For constraint (41b), since $x_i - z_i^d \in e_i^0(0) \oplus \mathcal{S}_i$ and the robust positive invariance of the set $\mathcal{S}_i$, we have $x_i^+ - z_i^d \in \bar{e}_i^+(0) \oplus \mathcal{S}_i$. For all $k \in \mathbb{N}_{N-2}$, $\bar{e}_i^+(k+1) = A_i \bar{e}_i^+(k) + B_i \overline{\pi}_i^+(k)$. Due to $\bar{e}_i^+(N-1) = e_i^0(N)$, $\overline{\pi}_i^+(N-1) = K_{if} e_i^0(N)$ and $\bar{e}_i^+(N) = (A_i + B_i K_{if}) e_i^0(N)$, it holds that $\bar{e}_i^+(N) = A_i \bar{e}_i^+(N-1) + B_i \overline{\pi}_i^+(N-1)$, implying that constraint (41c) is satisfied. For all $k \in \mathbb{N}_{N-2}$, $\bar{e}_i^+(k) + z_i^d \in \mathcal{X}_i \ominus \mathcal{S}_i$ and $\overline{\pi}_i^+(k) + v_i^d \in \mathcal{U}_i \ominus K_{iS} \mathcal{S}_i$. Combined with the terminal conditions in Assumption 4(ii), $\bar{e}_i^+(N-1) + z_i^d \in \mathcal{X}_i \ominus \mathcal{S}_i$ and $\overline{\pi}_i^+(N-1) + v_i^d \in \mathcal{U}_i \ominus K_{iS} \mathcal{S}_i$ hold. Thus, the state constraint (41d) and the input constraint (41e) are satisfied for all $k \in \mathbb{N}_{N-1}$. Due to the positive invariance of $\mathcal{E}_{if}$ and $\mathcal{Z}_{if}$, $\bar{e}_i^+(N) + z_i^d \in \mathcal{Z}_{if}$, satisfying constraint (41i). Furthermore, for $k \in \mathbb{N}_{N-2}$, we have $\mathbf{relint}(\overline{z}_i^+(k) \oplus C_i \cap \widetilde{\mathcal{O}}) = \emptyset$ with $\overline{z}_i^+(k) = \bar{e}_i^+(k) + z_i^d$. It follows from $\bar{e}_i^+(N-1) = e_i^0(N) \in \mathcal{E}_{if}$ and $\overline{z}_i^+(N-1) = \bar{e}_i^+(N-1) + z_i^d \in \mathcal{Z}_{if}$ as well as $\mathbf{relint}(\mathcal{Z}_{if} \oplus C_i \cap \widetilde{\mathcal{O}}) = \emptyset$ in Assumption 4(iv) that $\mathbf{relint}(\overline{z}_i^+(k) \oplus C_i \cap \widetilde{\mathcal{O}}) = \emptyset$ holds for all $k \in \mathbb{N}_{N-1}$. As a result, the separating hyperplane $\mathcal{H}(\overline{z}_i^+(k) \oplus C_i, \widetilde{\mathcal{O}}_h)$ and the safe half-space $\mathcal{F}(\overline{z}_i^+(k) \oplus C_i, \widetilde{\mathcal{O}}_h)$ can be formed between the sets $\overline{z}_i^+(k) \oplus C_i$ and $\widetilde{\mathcal{O}}_h$, thereby generating the safe set $\mathcal{R}(\overline{z}_i^+(k) \oplus \mathcal{C}_i, \widetilde{\mathcal{O}})$ by (30). Therefore, we have $\overline{z}_i^+(k) = \overline{e}_i^+(k) + z_i^d \in \mathcal{R}(\overline{z}_i^+(k) \oplus \mathcal{C}_i, \widetilde{\mathcal{O}})$ for all $k \in \mathbb{N}_{N-1}$, which implies constraint (41f) holds. By letting $\widetilde{\mathcal{C}}_i^+(k) = \overline{z}_i^+(k) \oplus \mathcal{C}_i$ and $\widetilde{\mathcal{S}}_j^+(k) = \hat{z}_j^+(k) \oplus \mathcal{S}_j$ with the constructed $\overline{z}_i^+(k) = z_i^0(k+1) = e_i^0(k+1) + z_i^d$ and $\hat{z}_j^+(k) = z_j^0(k+1) = e_j^0(k+1) + z_j^d$, it follows from (33)–(34) and $\epsilon(i,k) = \min_{j \in \mathbb{N}_a \setminus i} \epsilon(i,j,k)$ that, for $k \in \mathbb{N}_{N-1}$,

$$
\begin{aligned}
\alpha^\top(\widetilde{\mathcal{C}}_i(k), \widetilde{\mathcal{S}}_j(k)) z_i^0(k) &\geq \mu(\widetilde{\mathcal{C}}_i(k), \widetilde{\mathcal{S}}_j(k)) + \epsilon(i,j,k) \\
&\geq \mu(\widetilde{\mathcal{C}}_i(k), \widetilde{\mathcal{S}}_j(k)) + \alpha^\top(\widetilde{\mathcal{C}}_i(k), \widetilde{\mathcal{S}}_j(k))(z_j^0(k) - \hat{z}_j(k)) \\
&= \beta(\widetilde{\mathcal{C}}_i(k), \widetilde{\mathcal{S}}_j(k)) + \zeta(\widetilde{\mathcal{C}}_i(k), \widetilde{\mathcal{S}}_j(k)) + \alpha^\top(\widetilde{\mathcal{C}}_i(k), \widetilde{\mathcal{S}}_j(k))(z_j^0(k) - \hat{z}_j(k)) \\
&= \alpha^\top(\widetilde{\mathcal{C}}_i(k), \widetilde{\mathcal{S}}_j(k))(\hat{z}_j(k) + y_{sj}^0(\widetilde{\mathcal{C}}_i(k), \widetilde{\mathcal{S}}_j(k))) + \zeta(\widetilde{\mathcal{C}}_i(k), \widetilde{\mathcal{S}}_j(k)) + \alpha^\top(\widetilde{\mathcal{C}}_i(k), \widetilde{\mathcal{S}}_j(k))(z_j^0(k) - \hat{z}_j(k)) \\
&= \alpha^\top(\widetilde{\mathcal{C}}_i(k), \widetilde{\mathcal{S}}_j(k))(z_j^0(k) + y_{sj}^0(\widetilde{\mathcal{C}}_i(k), \widetilde{\mathcal{S}}_j(k))) + \zeta(\widetilde{\mathcal{C}}_i(k), \widetilde{\mathcal{S}}_j(k)).
\end{aligned}
$$

Thus, the sets $z_i^0(k) \oplus \mathcal{C}_i$ and $z_j^0(k) \oplus \mathcal{S}_j$ can be separated, that is, $\forall k \in \mathbb{N}_{N-1}$, $\mathbf{relint}(z_i^0(k) \oplus \mathcal{C}_i \cap z_j^0(k) \oplus \mathcal{S}_j) = \emptyset$, which ensures $\alpha^\top(z_i^0(k) \oplus \mathcal{C}_i, z_j^0(k) \oplus \mathcal{S}_j) z_i^0(k) \geq \mu(z_i^0(k) \oplus \mathcal{C}_i, z_j^0(k) \oplus \mathcal{S}_j)$ for $k \in \mathbb{N}_{N-1}$. Because of Assumption 4(iv), $\alpha^\top(z_i^0(N) \oplus \mathcal{C}_i, z_j^0(N) \oplus \mathcal{S}_j) z_i^0(N) \geq \mu(z_i^0(N) \oplus \mathcal{C}_i, z_j^0(N) \oplus \mathcal{S}_j)$ holds. Then, for all $k \in \mathbb{N}_{N-1}$, $\alpha^\top(\widetilde{\mathcal{C}}_i^+(k), \widetilde{\mathcal{S}}_j^+(k)) \overline{z}_i^+(k) \geq \mu(\widetilde{\mathcal{C}}_i^+(k), \widetilde{\mathcal{S}}_j^+(k))$, which is equivalent to $\alpha^\top(\widetilde{\mathcal{C}}_i^+(k), \widetilde{\mathcal{S}}_j^+(k)) \overline{z}_i^+(k) \geq \mu(\widetilde{\mathcal{C}}_i^+(k), \widetilde{\mathcal{S}}_j^+(k)) + \frac{\alpha^\top(\widetilde{\mathcal{C}}_i^+(k), \widetilde{\mathcal{S}}_j^+(k)) \hat{z}_i^+(k) - \mu(\widetilde{\mathcal{C}}_i^+(k), \widetilde{\mathcal{S}}_j^+(k))}{2}$, with $\overline{z}_i^+(k) = \hat{z}_i^+(k)$. This guarantees the satisfaction of constraint (41g). In virtue of the fact that $\overline{e}_i^+(k) = \hat{e}_i^+(k)$, we have $\overline{e}_i^+(k) - \hat{e}_i^+(k) = 0$, implying the satisfaction of compatibility constraint (41h) by $\overline{e}_i^+(k)$. Consequently, $\overline{\xi}_i^+(N) \in \mathbb{D}_{iN}(x_i, z_i^d, v_i^d, \xi_i(N), \overline{\xi}_i(N), \hat{\xi}_j(N))$ is a feasible trajectory at x_i^+ and the recursive feasibility of Algorithm 1 can be concluded by induction. ∎

Theorem 2. *Under Assumptions 1–4, the set $z_i^d \oplus \mathcal{S}_i$ is robustly asymptotically stable for the ith controlled closed-loop system $x_i^+ = A_i x_i + B_i \kappa_N^0 + w_i$, $w_i \in \mathcal{W}_i$ under Algorithm 1.*

Proof. Theorem 2 can be proved by following a similar line to the proof of Theorem 2 in Reference 35. As proved in Reference 35, the desired equilibrium point for the controlled system is at the origin and the set $\mathcal{S}_i$ is robustly asymptotically stable for $x_i^+ - z_i^d = A_i(x_i - z_i^d) + B_i(\kappa_N^0 - v_i^d) + w_i$. While the designated equilibrium position and input are set at z_i^d and v_i^d, naturally, the set $z_i^d \oplus \mathcal{S}_i$ is robustly asymptotically stable for the closed-loop system $x_i^+ = A_i x_i + B_i \kappa_N^0 + w_i$ with respect to the disturbance w_i. ∎

5 NUMERICAL SIMULATIONS

To demonstrate the efficacy of Algorithm 1 in addressing the formation control problem in a multi-agent system within an environment containing obstacles and disturbances, we present an example featuring three agents [41] and two static obstacles. In this example, three multi-agent systems are set the same and their system matrix pairs in (1) are given by

$$
A_1 = A_2 = A_3 = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \text{and } B_1 = B_2 = B_3 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{48}
$$

Each agent is subject to the physical constraints (2), whose representations are given by (3)–(5) in Assumption 1(ii). In our simulation, three agents move within a square area, with a side length of 10 m. Therefore, the position state constraints for all agents are set with $\mathcal{X}_1 = \mathcal{X}_2 = \mathcal{X}_3 = [-10, 10] \times [-10, 10]$. The control input constraints and the state disturbances are given by $\mathcal{U}_1 = \mathcal{U}_2 = \mathcal{U}_3 = [-2, 2]$ and $\mathcal{W}_1 = \mathcal{W}_2 = \mathcal{W}_3 = [-0.1, 0.1]$, respectively, in accordance with Assumption 1(ii). In addition, two static obstacles are considered in the simulation, and both of them satisfy Assumption 3(i). Static obstacles 1 and 2 are represented by $\widetilde{\mathcal{O}}_1 := o_1 \oplus \mathcal{O}_1$ and $\widetilde{\mathcal{O}}_2 := o_2 \oplus \mathcal{O}_2$, respectively, where o_1 and o_2 are the center positions of the obstacles and $\mathcal{O}_1$ and $\mathcal{O}_2$ denote the corresponding shapes of two obstacles, given by

$$
o_1 = \begin{bmatrix} 4.4 \\ 1.1 \end{bmatrix}, \ o_2 = \begin{bmatrix} -6.1 \\ -1.1 \end{bmatrix}, \text{ and } \ O_1 = O_2 = \begin{bmatrix} 2 & 0 \\ -2 & 0 \\ 0 & 2 \\ 0 & -2 \end{bmatrix}.
$$

The goal is to control the three agents to reach their assigned destinations z_1^d, z_2^d, z_3^d without collisions, while satisfying their corresponding state and control input constraints and avoiding the static obstacles efficiently in the presence of additive state disturbances. In this example, the assigned equilibrium positions are $z_1^d = [-2; 0], z_2^d = [0; 0], z_3^d = [2; 0]$. The local state feedback gain matrix K_{iS} for the ith agent is obtained by placing the desired closed-loop poles at $(0.2, 0.4)$. In this case, $K_{1S} = K_{2S} = K_{3S} = [-0.48, -1.40]$. By solving the unconstrained optimal control problem for (A_1, B_1, Q_1, R_1), (A_2, B_2, Q_2, R_2), and (A_3, B_3, Q_3, R_3), with stage cost matrices $Q_1 = Q_2 = Q_3 = I$ and $R_1 = R_2 = R_3 = 0.01I$, the terminal feedback gain matrices K_{1f}, K_{2f}, and K_{3f} satisfying Assumption 4(i), and the terminal cost matrices P_1, P_2, and P_3 are obtained, given by

$$
\begin{aligned}
K_{1f} = K_{2f} = K_{3f} &= \left[-0.61361.6099\right], \\
P_1 = P_2 = P_3 &= \begin{bmatrix} 2.6235 & 1.6296 \\ 1.6296 & 2.6457 \end{bmatrix}.
\end{aligned} \tag{49}
$$

The robust positively invariant set $\mathcal{S}_i$ for the ith agent is determined by the minimal robust positively invariant sets, which can be calculated by using the methods in Reference 40. The irreducible representations of $\mathcal{S}_1, \mathcal{S}_2$, and $\mathcal{S}_3$ are identical, consisting of 16 affine inequalities. The method detailed in Section 3.3 is employed to compute the terminal constraint sets $\mathcal{Z}_{if}$ that satisfy Assumption 4. In this simulation, the irreducible representations of $\mathcal{Z}_{1f}, \mathcal{Z}_{2f}$, and $\mathcal{Z}_{3f}$ include 6, 4, and 6 affine inequalities, respectively. For **OCP** proposed in Section 4, the prediction horizon length is set to $N = 7$. The initial state x_i for three agents are $x_1 = [3; 2], x_2 = [-2; -4]$ and $x_3 = [8; -4]$. Under such a setting, **OCP** can be solved by QP in MATLAB R2019a.

Figure 1 illustrates the position trajectories of all agents controlled by the proposed algorithm. The figure includes visual representations of two static obstacles shown as black rectangles. The solid and dotted black lines represent the nominal and actual states of agent 1, while the solid and dotted blue lines depict the nominal and actual states of agent 2. Similarly, the red solid and dotted lines illustrate the nominal and actual states of agent 3. We can see from Figure 1 that agent 1 departs from the initial state $x_1 = [3; 2]$ and successfully reaches the target position $z_1^d = [-2; 0]$ within the designated white terminal constraint set. Remarkably, agent 1 effectively navigates around static obstacles and the other two agents. Similarly, starting from the initial state $x_2 = [-2; -4]$, agent 2 avoids the collisions to reach its intended target position $z_2^d = [0; 0]$ within the blue terminal constraint set. In the same manner, agent 3 initiates its trajectory from the initial point $x_3 = [8; -4]$ and eventually reaches the target point $z_3^d = [2; 0]$. The red area surrounding the target point z_3^d represents the terminal constraint set specific to the optimal control problem for agent 3.

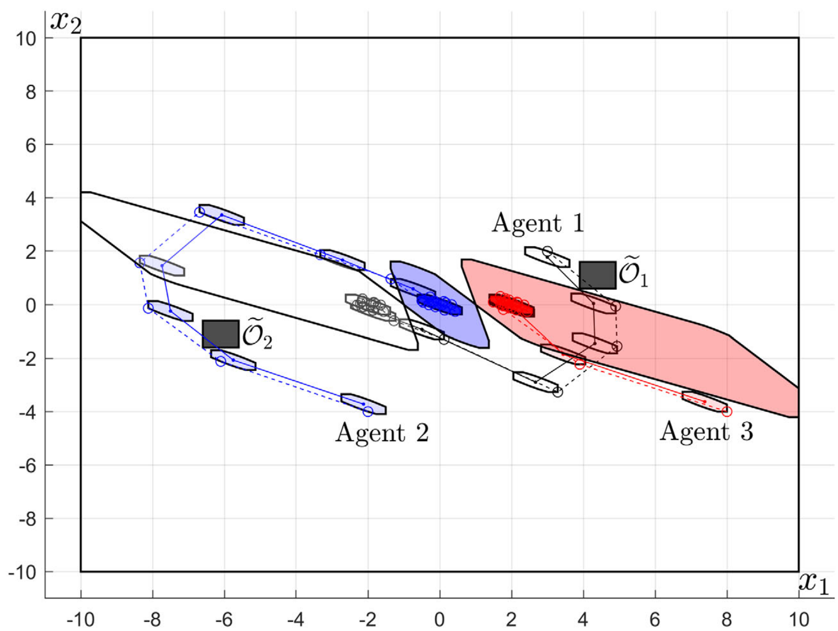

FIGURE 1 Position trajectories of all the agents under Algorithm 1.

Figure 2 presents the relative distances among the three agents. The labels d_{12}, d_{13}, and d_{23} indicate the relative distances between agents 1 and 2, agents 1 and 3, and agents 2 and 3, respectively. The symbol d represents the safe distance, which is set to 0.1 m. As observed in Figure 2, the distances between any two agents consistently exceed the designated safe distance d. This demonstrates that collision avoidance is ensured throughout the movement of the agents. Additionally, Figure 3 depicts the relative distances between each agent and obstacle $\widetilde{\mathcal{O}}_1$, while Figure 4 displays the relative distances between each agent and obstacle $\widetilde{\mathcal{O}}_2$. Based on the information presented in Figures 1 and 3, it is evident that agent 1 initially maintains close proximity to obstacle $\widetilde{\mathcal{O}}_1$, but successfully avoids it. Similarly, agent 2 successfully avoids obstacle $\widetilde{\mathcal{O}}_2$ at time step $t = 1$, as illustrated in Figures 1 and 4. After time step $t = 10$, the relative distances between the three agents, as depicted in Figure 2, and the relative distances between each agent and obstacles, as depicted in Figures 3 and 4, tend

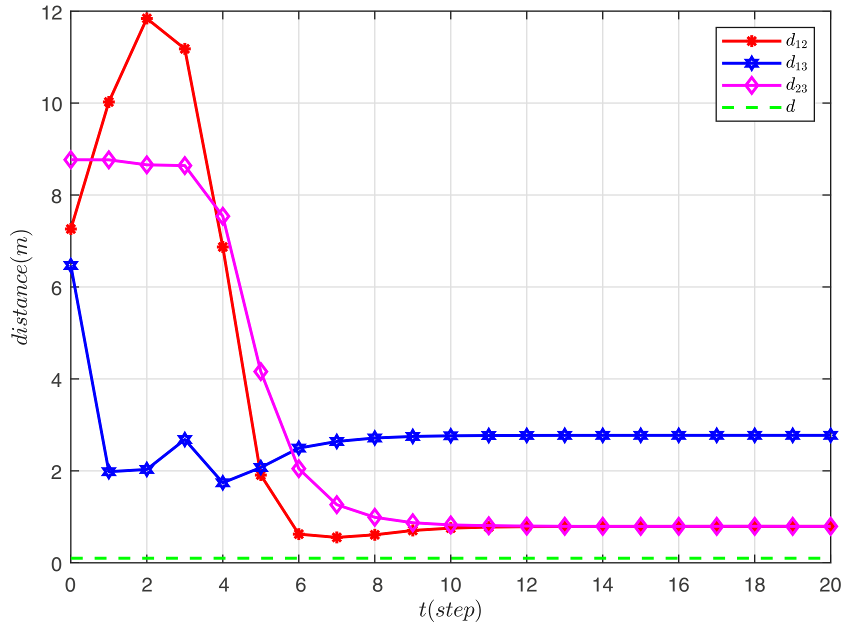

FIGURE 2 The relative distances between three agents under Algorithm 1.

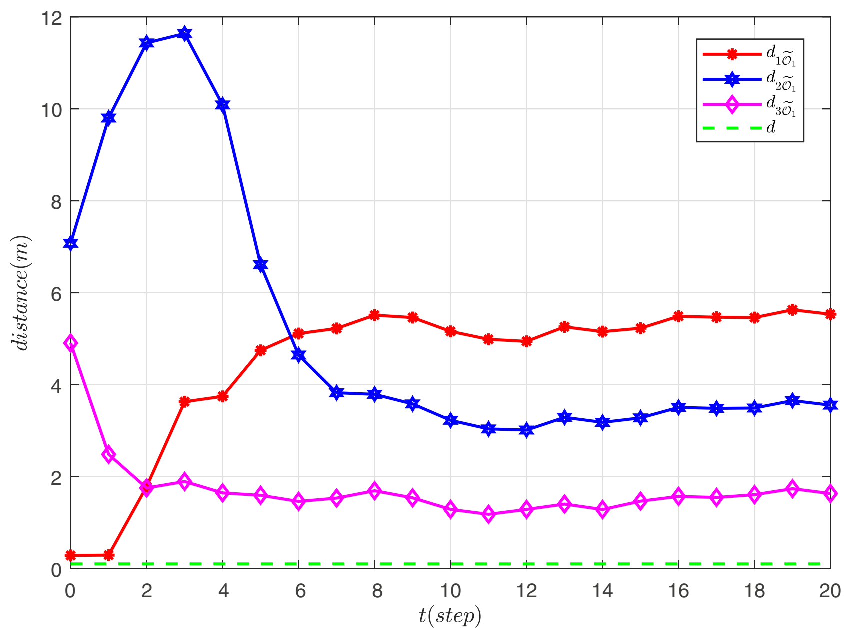

FIGURE 3 The relative distances between each agent and obstacle $\widetilde{\mathcal{O}}_1$ under Algorithm 1.

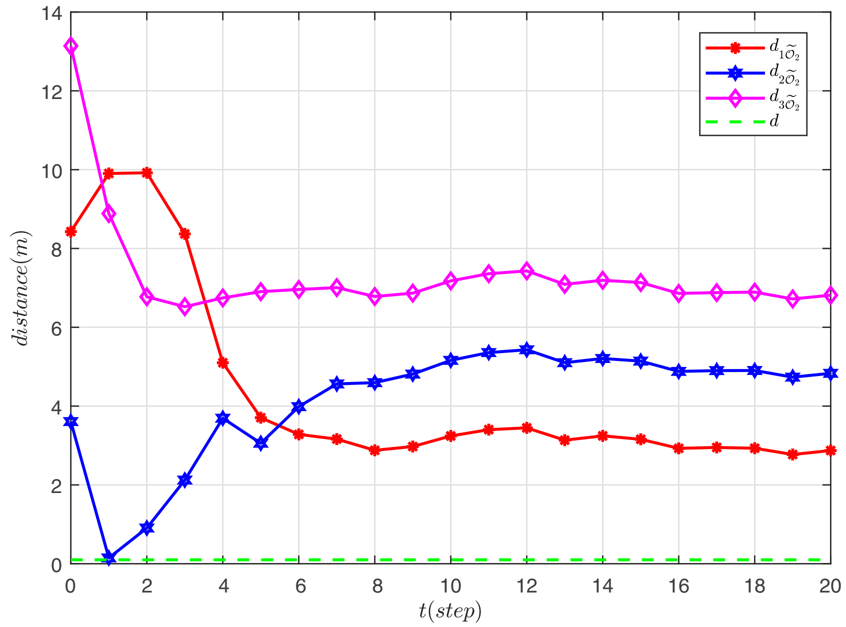

FIGURE 4 The relative distances between each agent and obstacle $\widetilde{\mathcal{O}}_2$ under Algorithm 1.

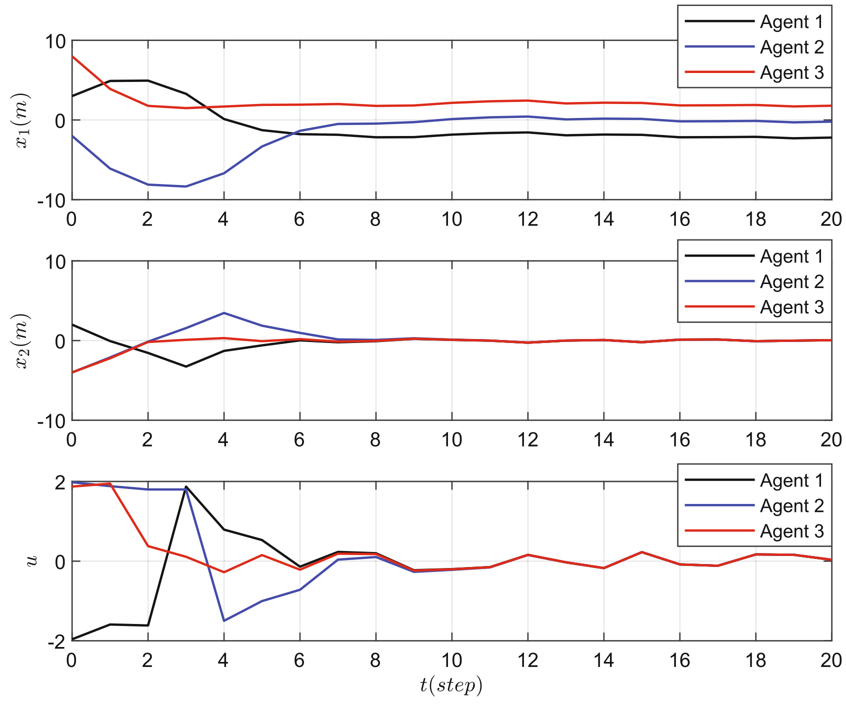

FIGURE 5 Closed-loop position trajectory components x_1, x_2 and control input u of three agents under Algorithm 1.

to stabilize. This indicates that the three agents have successfully achieved the desired formation, characterized by the predefined target positions shown in Figure 1.

The upper and middle subfigures of Figure 5 show the closed-loop position trajectory components of the three agents under Algorithm 1. It is notable that the position components x_1 and x_2 for all three agents remain within the specified state constraint interval of $[-10, 10]$. The lower subfigure of Figure 5 presents the closed-loop control inputs over time, and the control inputs for all three agents satisfy the prescribed input constraints, namely, $u_1 \in [-2, 2]$, $u_2 \in [-2, 2]$, and $u_3 \in [-2, 2]$. These comprehensive illustrations verify the effectiveness of Algorithm 1, demonstrating its capability to

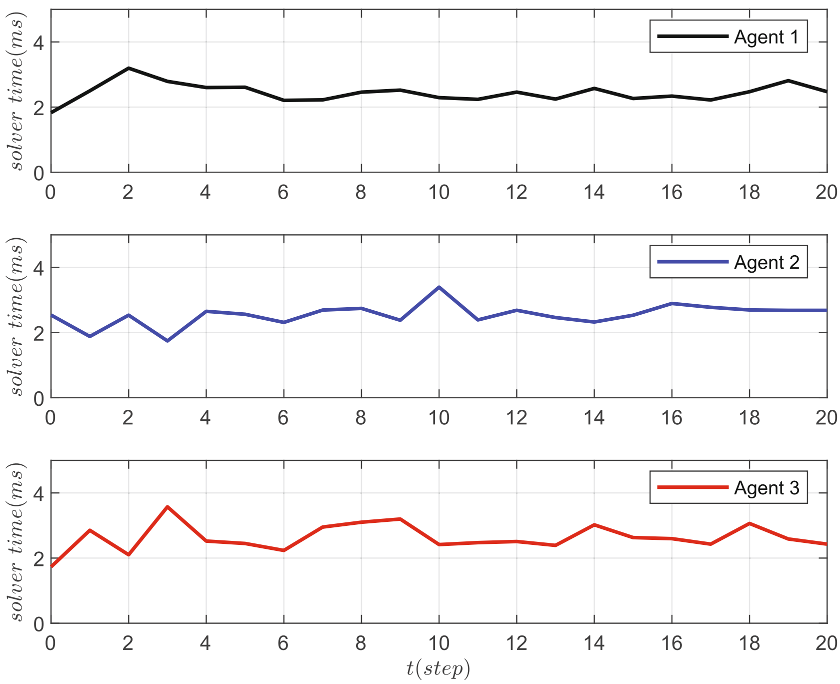

FIGURE 6 Solver times for three agents under Algorithm 1.

guide the agents without collisions efficiently while maintaining adherence to both state and input constraints. Figure 6 illustrates the solver times of the online process in the proposed Algorithm 1 for three agents. The graph shows that at each time step, the solver times for all agents are consistently below 4 ms. This result indicates that the proposed algorithm exhibits high computational efficiency.

6 | CONCLUSIONS

We propose a convex strategy for obstacle and collision avoidance for multi-agent systems in the presence of additive disturbances under a distributed RMPC framework. To address the challenge of simultaneous solutions for all agents, we introduce assumed nominal position and input trajectories, and implement compatibility constraints that consider the residual collision avoidance margin of the optimal states for each agent and its neighbors obtained at the last time step. These constraints aim to decouple the coupled collision avoidance constraints while ensuring agreement among the agents. To handle the inherently nonconvex obstacle avoidance and collision avoidance constraints, we convexify them into time-varying closed polyhedral constraints using the construction of separating hyperplanes and safe sets. Particularly, to ensure the satisfaction of coupled collision avoidance constraints in the presence of disturbances, the residual collision avoidance margin considered in the design of compatibility constraints is introduced into the closed polyhedral constraints for collision avoidance. Consequently, the original nonconvex and intractable collision avoidance optimal control problems in the context of uncertainties can be simultaneously solved using standard QP. A rigorous analysis of the recursive feasibility of the proposed algorithm is guaranteed with the designed compatibility constraints and local terminal constraints, while the closed-loop stability of the disturbed multi-agent systems is ensured using the designed local terminal constraint sets for each agent. In order to validate the effectiveness of our approach, we present a simulation example involving three agents and two static obstacles in the presence of disturbances. The simulation results demonstrate that the agents are able to successfully avoid static obstacles while maintaining collision-free trajectories throughout all time steps in the distributed and disturbed context. Moreover, at each time step, the solver times no more than 4 ms for any agent show a high efficiency of our proposal.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China under Grant 62173036 and Grant 62122014.

CONFLICT OF INTEREST STATEMENT

The authors declare no potential conflict of interests.

DATA AVAILABILITY STATEMENT

Research data are not shared.

ORCID

Li Dai https://orcid.org/0000-0002-7268-7548
Yuanqing Xia https://orcid.org/0000-0002-5977-4911

REFERENCES

1. Oh KK, Park MC, Ahn HS. A survey of multi-agent formation control. *Automatica*. 2015;53:424-440.
2. Liu GP, Zhang S. A survey on formation control of small satellites. *Proc IEEE*. 2018;106(3):440-457.
3. Liu Y, Bucknall R. A survey of formation control and motion planning of multiple unmanned vehicles. *Robotica*. 2018;36(7):1019-1047.
4. Hou Z, Wang W, Zhang G, Han C. A survey on the formation control of multiple quadrotors. *Proceeding of 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*. IEEE; 2017:219-225.
5. Kiattisin K. Formation control of mobile robots: survey. *J Eng Innov*. 2011;4(1):50-64.
6. Egerstedt M, Hu X. Formation constrained multi-agent control. *IEEE Trans Rob Autom*. 2001;17(6):947-951.
7. Yang H, Li Q, Zuo Z, Zhao H. Event-triggered model predictive control for multi-vehicle systems with collision avoidance and obstacle avoidance. *Int J Robust Nonlinear Control*. 2021;31(11):5476-5494.
8. Wu B, Peng Z, Wen G, Huang T, Rahmani A. Distributed time-varying optimization control for multirobot systems with collision avoidance by hierarchical approach. *Int J Robust Nonlinear Control*. 2023;33(6):3928-3946.
9. Raković SV, Levine WS. *Handbook of Model Predictive Control*. Springer; 2018.
10. Mayne DQ. Model predictive control: recent developments and future promise. *Automatica*. 2014;50(12):2967-2986.
11. Rawlings JB, Mayne DQ. *Model Predictive Control: Theory and Design*. Nob Hill Publishing; 2009.
12. Mayne DQ, Seron MM, Raković SV. Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*. 2005;41(2):219-224.
13. Langson W, Chryssochoos L, Raković SV, Mayne DQ. Robust model predictive control using tubes. *Automatica*. 2004;40(1):125-133.
14. Raković SV, Mayne DQ. A simple tube controller for efficient robust model predictive control of constrained linear discrete time systems subject to bounded disturbances. *IFAC Proc Vol*. 2005;38(1):241-246.
15. Raković SV, Kouvaritakis B, Findeisen R, Cannon M. Homothetic tube model predictive control. *Automatica*. 2012;48(8):1631-1638.
16. Raković SV, Kouvaritakis B, Cannon M, Panos C, Findeisen R. Parameterized tube model predictive control. *IEEE Trans Automat Contr*. 2012;57(11):2746-2761.
17. Raković SV, Levine WS, Açikmese B. Elastic tube model predictive control. *Proceeding 2016 of American Control Conference (ACC)*. IEEE; 2016:3594-3599.
18. Negenborn RR, Maestre JM. Distributed model predictive control: an overview and roadmap of future research opportunities. *IEEE Control Syst Mag*. 2014;34(4):87-97.
19. Trodden P, Richards A. Cooperative distributed MPC of linear systems with coupled constraints. *Automatica*. 2013;49(2):479-487.
20. Giselsson P, Rantzer A. On feasibility, stability and performance in distributed model predictive control. *IEEE Trans Automat Contr*. 2013;59(4):1031-1036.
21. Toumieh C, Lambert A. Decentralized multi-agent planning using model predictive control and time-aware safe corridors. *IEEE Robot Autom Lett*. 2022;7(4):11110-11117.
22. Wang Y, Yang Y, Pu Y, Manzie C. Path following by formations of agents with collision avoidance guarantees using distributed model predictive control. *Proceeding of American Control Conference (ACC)*. IEEE; 2021:3352-3357.
23. Vargas S, Becerra HM, Hayet JB. MPC-based distributed formation control of multiple quadcopters with obstacle avoidance and connectivity maintenance. *Control Eng Pract*. 2022;121:105054.
24. Dubay S, Pan YJ. Distributed MPC based collision avoidance approach for consensus of multiple quadcopters. *Proceeding of IEEE 14th International Conference on Control and Automation (ICCA)*. IEEE; 2018:155-160.
25. Ravanshadi I, Boroujeni EA, Pourgholi M. Centralized and distributed model predictive control for consensus of non-linear multi-agent systems with time-varying obstacle avoidance. *ISA Trans*. 2023;133:75-90.
26. Gräfe A, Eickhoff J, Trimpe S. Event-triggered and distributed model predictive control for guaranteed collision avoidance in UAV swarms. *IFAC-PapersOnLine*. 2022;55(13):79-84.
27. Lyu Y, Hu J, Chen BM, Zhao C, Pan Q. Multivehicle flocking with collision avoidance via distributed model predictive control. *IEEE Trans Cybern*. 2019;51(5):2651-2662.
28. Wang P, Ding B. A synthesis approach of distributed model predictive control for homogeneous multi-agent system with collision avoidance. *Int J Control*. 2014;87(1):52-63.
29. Dai L, Cao Q, Xia Y, Gao Y. Distributed MPC for formation of multi-agent systems with collision avoidance and obstacle avoidance. *J Franklin Inst*. 2017;354(4):2068-2085.

30. Wang P, Feng X, Li W, Ping X, Yu W, Qin X. A robust distributed receding horizon control synthesis approach for simultaneous tracking, regulation and formation of multiple perturbed wheeled vehicles with collision avoidance. *Int J Robust Nonlinear Control*. 2021;31(9):4436-4457.
31. Li S, Ye D, Xiao Y, Sun Z. Robust distributed model predictive control for satellite cluster reconfiguration with collision avoidance. *Aerosp Sci Technol*. 2022;130:107917.
32. Zhang X, Liniger A, Borrelli F. Optimization-based collision avoidance. *IEEE Trans Control Syst Technol*. 2020;29(3):972-983.
33. Dai L, Hao Y, Xie H, Sun Z, Xia Y. Distributed robust MPC for nonholonomic robots with obstacle and collision avoidance. *Control Theory Technol*. 2022;20(1):32-45.
34. Raković SV, Zhang S, Hao Y, Dai L, Xia Y. Convex MPC for exclusion constraints. *Automatica*. 2021;127:109502.
35. Sun H, Zhang S, Dai L, Raković SV. Locally convexified rigid tube MPC. *IET Control Theory Appl*. 2023;17(4):446-462.
36. Raković SV, Zhang S, Dai L, Hao Y, Xia Y. Convex model predictive control for collision avoidance. *IET Control Theory Appl*. 2021;15(9):1270-1285.
37. Schneider R. *Convex Bodies: The Brunn–Minkowski Theory*. Cambridge University Press; 2014.
38. Rockafellar RT. *Convex Analysis*. Princeton University Press; 1970.
39. Blanchini F, Miani S. *Set–Theoretic Methods in Control*. Birkhauser; 2008.
40. Raković SV, Kerrigan EC, Kouramas KI, Mayne DQ. Invariant approximations of the minimal robust positively invariant set. *IEEE Trans Automat Contr*. 2005;50(3):406-410.
41. Chen Y, Chen H, Chen B, Zhan J. Leader-following consensus of second-order multi-agent systems: new results based on linear transformation method. *Proceeding of Chinese Automation Congress (CAC)*. IEEE; 2020:6153-6158.

How to cite this article: Sun H, Xie H, Dai L, Xia Y. Convex distributed robust model predictive control for collision and obstacle avoidance. *Int J Robust Nonlinear Control*. 2024;1-21. doi: 10.1002/rnc.7390

APPENDIX A. INITIALIZATION OF ALGORITHM 1

The initialization step aims to get the initial feasible and assumed trajectories which ensure no collisions between agents and avoiding obstacles.

1(a) Solve the finite horizon rigid tube optimal control problem **OCP** without obstacle avoidance constraints (41f), and collision avoidance constraints (41g) and compatibility constraints (41h), which is a strictly convex QP problem. Get the optimized trajectories $\xi_i^0(N) := (\{e_i^0(k)\}_{k=0}^{N}, \{\pi_i^0(k)\}_{k=0}^{N-1})$ for agent $i \in \mathbb{N}_a$ and $\xi_j^0(N) := (\{e_j^0(k)\}_{k=0}^{N}, \{\pi_j^0(k)\}_{k=0}^{N-1})$ for agent $j \in \mathbb{N}_a \setminus i$.

1(b) Construct $\bar{\mathbf{e}}_i(N) := \{\bar{e}_i(k)\}_{k=0}^{N}$ and $\hat{\mathbf{e}}_j(N) := \{\hat{e}_j(k)\}_{k=0}^{N}$ by setting, for $k \in \mathbb{N}_N$, $\bar{e}_i(k) = e_i^0(k)$ and $\hat{e}_j(k) = e_j^0(k)$. Obtain $\bar{\mathbf{z}}_i(N) := \{\bar{z}_i(k)\}_{k=0}^{N}$ with $\bar{z}_i(k) = \bar{z}_i(k) + z_i^d$ and $\hat{\mathbf{z}}_j(N) := \{\hat{z}_j(k)\}_{k=0}^{N}$ with $\hat{z}_j(k) = \hat{e}_i(k) + z_j^d$.

- For each $k \in \mathbb{N}_N$, $i \in \mathbb{N}_a$ and each $j \in \mathbb{N}_a \setminus i$, check if the sets $\bar{z}_i(k) \oplus C_i$ and $\hat{z}_j(k) \oplus \mathcal{S}_j$ can be separated by calculating

$$\phi_{(i,j,k)} := \min_r \{S_j^r \hat{z}_j(k) + 1 - S_j^r \bar{z}_i(k) + \iota_j^r \, : \, r \in \mathbb{N}_{[1:p_{sj}]}\},$$

where $\iota_j^r \in \mathbb{R}$ is a scalar, given by

$$\iota_j^r = \mathrm{h}(C_i, S_j^{r\top}).$$

- For each $k \in \mathbb{N}_N$ and each $h \in \mathbb{N}_o$, check if the sets $\bar{z}_i(k) \oplus C_i$ and $o_h \oplus \mathcal{O}_h$ can be separated by calculating

$$\varphi_{(i,h,k)} := \min_r \{O_h^r o_h + 1 - O_h^r \bar{z}_i(k) + \tau_h^r \, : \, r \in \mathbb{N}_{[1:p_h]}\},$$

where $\tau_h^r \in \mathbb{R}$ is a scalar, given by

$$\tau_h^r = \mathrm{h}(C_i, O_h^{r\top}).$$

- If $\phi_{(i,j,k)} \leq 0$ for all $k \in \mathbb{N}_N$ and all $j \in \mathbb{N}_a \setminus i$ and $\varphi_{(i,h,k)} \leq 0$ for all $k \in \mathbb{N}_N$ and all $h \in \mathbb{N}_o$, set $\overline{\xi}_i(N) = \xi_i^0(N)$ and $\hat{\xi}_j(N) = \xi_j^0(N)$. Otherwise, go to step 1(c).

1(c) – When $\phi_{(i,j,k)} > 0$, for the corresponding j and k, construct the related separating half-space

$$\mathcal{F}_{(i,j,k)} = \{x_i(k) \;:\; S_j^{r^0} x_i(k) \geq S_j^{r^0} \hat{z}_j(k) + 1\} \text{ with}$$
$$r^0 = \arg\min_r \{S_j^r \hat{z}_j(k) + 1 - S_j^r \overline{z}_i(k) + \iota_j^r \;:\; r \in \mathbb{N}_{[1:p_{sj}]}\}.$$

- When $\varphi_{(i,h,k)} > 0$, for the corresponding h and k, construct the related separating half-space

$$\mathcal{F}_{(i,h,k)} = \{x_i(k) \;:\; O_h^{r^0} x_i(k) \geq O_h^{r^0} o_h + 1\} \text{ with}$$
$$r^0 = \arg\min_r \{O_h^r o_h + 1 - O_h^r \overline{z}_i(k) + \tau_h^r \;:\; r \in \mathbb{N}_{[1:p_h]}\}.$$

1(d) Go to step 1(a) but the optimization problem in 1(a) is updated by adding the polyhedral constraints constructed in step 1(c).