

# Deadlock Resolution and Recursive Feasibility in MPC-based Multi-robot Trajectory Generation

Yuda Chen, Meng Guo and Zhongkui Li, *Senior Member, IEEE*

**Abstract**—Online collision-free trajectory generation within a shared workspace is fundamental for most multi-robot applications. However, many widely-used methods based on model predictive control (MPC) lack theoretical guarantees on the feasibility of underlying optimization. Furthermore, when applied in a distributed manner without a central coordinator, deadlocks often occur where several robots block each other indefinitely. Whereas heuristic methods such as introducing random perturbations exist, no profound analyses are given to validate these measures. Towards this end, we propose a systematic method called infinite-horizon model predictive control with deadlock resolution. The MPC is formulated as a convex optimization over the proposed modified buffered Voronoi with warning band. Based on this formulation, the condition of deadlocks is formally analyzed and proven to be analogous to a force equilibrium. A detection-resolution scheme is proposed, which can effectively detect deadlocks online before they even happen. Once detected, it utilizes an adaptive force scheme to resolve deadlocks, under which no stable deadlocks can exist under minor conditions on robots' target positions. In addition, the proposed planning algorithm ensures recursive feasibility of the underlying optimization at each replanning under both input and model constraints, is concurrent for all robots and requires only local communication. Comprehensive simulation and experiment studies are conducted over large-scale multi-robot systems. Significant improvements on success rate are reported, in comparison with other state-of-the-art methods and especially in crowded and high-speed scenarios.

**Index Terms**—Multi-robot systems, motion planning, trajectory generation, deadlock resolution, recursive feasibility, collision avoidance.

## I. INTRODUCTION

**C**OLLISION-free trajectory generation is essential for multi-robot systems to perform various missions in a shared environment, such as cooperative inspection and transportation [1], [2], [3], [4]. However, it becomes especially challenging when a large number of agile robots navigate in a crowded space with high speed. The commonly-seen multi-robot trajectory generation (MATG) algorithms can be classified into roughly six categories: potential fields [5], [6] that design virtual driving forces induced by artificial potentials; geometric guidance [7], [8] such as reciprocal velocity obstacles (RVO) [8], [9], [10] that analyze the geometric properties based on the position and velocity of the others; conflicts

resolution [11], [12] that designs heuristic rules to resolve potential collisions; learning-based methods [13], [14], [15] that rely heavily on accurate simulators and reward shaping; control-law-based methods [16], [17], [18] that have a strong theoretical guarantee; and optimization-based methods [19], [20], [21] that model the problem as numerical optimizations.

Optimization-based methods have gained increasing popularity recently due to its modeling capability and extensibility. In particular, optimization-based methods construct and solve various optimizations to achieve collision-free navigation, such as the mixed integer quadratic programming in [19], sequential convex programming in [22], and model predictive control (MPC) in [23]. However, most of the aforementioned methods do not ensure explicitly feasibility of the underlying optimization during the whole navigation. In other words, the optimization might be infeasible at one time step, thus the whole system stops. Some works in [24], [25] and [26] propose to tackle the feasibility problem by gradually scaling up the time step, which however requires a centralized coordinator. Another work [27] introduces the notion of control barrier function, which can guarantee collision-free trajectory. Nonetheless, it might be overly conservative due to excessive breaking. The methods in [28] ensure feasibility by utilizing relative-safe corridor among robots. Last but not least, the above methods often impose a fully-connected communication network with high communication burden.

More importantly, another well-known issue in MATG is that deadlocks often occur during navigation in multi-robot systems. Formally, a deadlock happens when multiple robots are blocked by each other indefinitely and cannot make any progress towards their targets [29]. It is often caused by the symmetric configuration of the underlying system [30], and the lack of a central coordinator [31]. A commonly-used resolution scheme is inspired by the right-hand rules. For instance, an artificial perturbation to the right-hand side of each robot is introduced in [27] in order to break the equilibrium of deadlock. This however can lead to unpredictable behavior and even safety issue in practice as the magnitude of such perturbations is hard to determine. Other works in [32], [33], [34] instead propose to select a detour point on the right-hand side of each robot as the temporary target. Nonetheless, the validity and effectiveness of these schemes still lack theoretical analyses and guarantee on the performance, and might lead to livelock problems in practice where the robots oscillate around the deadlock positions indefinitely. Similar clockwise maneuver is adopted in [35] which shows that the tangent motion can resolve deadlocks for some cases. Important advances on deadlock analysis and resolution are recently reported

his work was supported in part by the National Science and Technology Major Project under grant 2022ZD0116401; in part by the National Natural Science Foundation of China under grants U2241214, T2121002, 62373008, and 62203017. Corresponding author: Zhongkui Li.

The authors are with the State Key Laboratory for Turbulence and Complex Systems, Department of Mechanics and Engineering Science, College of Engineering, Peking University, Beijing 100871, China. E-mail: {xyy, meng.guo, zhongkli}@pku.edu.cn.

in [30], [36], [37]. The authors in [30], [36], [37] analyze the condition of deadlocks based on the Karush-Kuhn-Tucker (KKT) formulation for multi-robot systems and present a proportional-derivative control law to resolve deadlock, which can be theoretically shown for the case with no more than three robots. As pointed out in [32], algorithms that can provably avoid deadlock in general cases without a global coordinator still await.

To address the open problem of deadlock resolution with feasibility guarantee in a distributed manner, this work proposes a novel systematic trajectory generation method, called infinite horizon model predictive control with deadlock resolution (IMPC-DR). Firstly, the traditional buffered Voronoi cells (BVC) proposed in [32] is extended by taking into account full planned trajectories of neighboring robots and introducing a velocity-dependent buffer width. Furthermore, an extra warning band is added to the terminal horizon to facilitate the resolution of potential deadlocks. An improved distributed MPC formulation is proposed based on these constraints and a novel cost function that deals with potential deadlocks. Given this formulation, the condition of deadlocks during navigation is formally analyzed and shown to be analogous to a force equilibrium between the attractive forces from the targets and repulsive forces from the neighboring robots. Consequently, a detection mechanism is designed to detect any potential deadlocks online and before they might happen. Once detected, an adaptive resolution scheme is followed to incrementally adjust the inter-robot repulsive force to break the equilibrium. Properties of the proposed algorithm regarding deadlock resolution, recursive feasibility and local communication are all formally analyzed and proven.

To summarize, the main contributions of this paper are as follows:

- The modified buffered Voronoi cells with warning band (MBVC-WB) introduces a novel trajectory-based and velocity-dependent space partition technique. It can avoid potential collision between the consecutive sampling points, which may take place in the traditional point-based BVC in [32] and the sampling-based methods in, e.g., [23], [38].
- The condition of deadlocks for MPC-based MATG problems is formally analyzed and revealed as a force equilibrium. This is more general than an earlier result in [30] for three robots. Based on this condition, a novel online detection scheme is provided to detect potential deadlocks early-on before they appear, i.e., not afterwards as in [27], [32]. Furthermore, in contrast to heuristic methods in [27], [33], [34], the proposed deadlock resolution scheme has a theoretical demonstration that the inter-robot repulsive forces are adapted in a smooth way to falsify the deadlock condition, again before potential deadlocks appear. It is formally proven that under the proposed resolution scheme, no stable deadlocks can exist under minor conditions on the target positions.
- The proposed complete algorithm not only ensures the deadlock resolution, but also guarantees that the optimization at each replanning is recursively feasible. Such an assurance for feasibility is often overlooked and simply

assumed in related works [22], [39], [40], rather than guaranteed explicitly and formally.

- Effectiveness and performance of the proposed algorithm are validated extensively by numerical simulations against other state-of-the-art methods including iSCP [39], DMPC [40] and BVC [32]. Our method shows a significant increase in both success rate and feasibility, especially for large-scale crowded and high-speed scenarios.
- Hardware experiments are carried out using nano quadrotors Crazyflies in a workspace captured by an indoor motion capture system OptiTrack. Up to 20 nano quadrotors successfully perform experiments of antipodal transitions in 2D and 3D, validating the applicability of the proposed algorithms on real-time platforms.

As mentioned earlier, the problem of deadlock resolution considered in this paper is restricted to the MATG area. It should be mentioned that deadlock phenomena also take place in other areas such as computer sciences [41] and discrete event systems [42], [43], [44], [45], [46]. For instance, the works [42], [43], [44] address deadlock resolution in multi-robot path planning on topological graphs in the area of traffic management. Although the deadlock phenomena in different areas are similar, yet the root cause, the resolution schemes and the implemented tools are quite distinct. More specifically, the deadlocks considered here belongs to the category of policy-induced deadlocks, due to the distributed and online trajectory planning scheme. In contrast, the deadlocks from the aforementioned works are often caused by the topological structure and the transition condition of the robots. Thus, the associated resolution schemes and the involved tools are rather distinct. The former requires the real-time adaptation of robot trajectories under dynamical and geometric constraints, while the latter often focuses on designing discrete policies using tools such as finite-state automata or labeled transition systems.

The remaining parts of this paper are organized as follows. The problem statement is described in Section II. The method and the corresponding analysis are presented in Section III and Section IV, respectively. Numerical simulations and hardware experiments are given in Section V. Conclusions and future work can be found in Section VI. The nomenclatures used in this paper are summarized in Table I.

## II. PROBLEM STATEMENT

This section formulates the multi-robot trajectory generation (MATG) problem considered in this paper.

### A. Robot Dynamics

Consider a team of  $N$  robots, where each robot  $i \in \mathcal{N} \triangleq \{1, 2, \dots, N\}$  is modeled as a point mass in  $\mathbb{R}^d$ , and  $d = 2, 3$  is the dimension of the configuration space. Furthermore, its motion is approximated by the double integrator, i.e.,

$$\dot{x}^i(t+h) = \mathbf{A}x^i(t) + \mathbf{B}u^i(t),$$

where  $x^i(t) = [p^i(t), v^i(t)]$  is robot  $i$ 's state, including the position  $p^i(t)$  and velocity  $v^i(t)$ ,  $u^i(t)$  as acceleration

TABLE I  
NOMENCLATURES

$p_{\text{target}}^i$	Target of robot $i$ .
$p^i, p^{ij}$	Position of robot $i$ , $p^{ij} = p^i - p^j$ .
$p_k^i$	Planned position of robot $i$ at the $k$ -th step of horizon.
$\mathcal{P}^i$	Planned trajectory of robot $i$ .
$\bar{\mathcal{P}}^i$	Predetermined trajectory of robot $i$ .
$\bar{p}_k^i$	Position of robot $i$ at the $k$ -th step of horizon in $\bar{\mathcal{P}}^i$ .
$v_k^i$	Velocity of robot $i$ at the $k$ -th step of horizon.
$x_k^i$	$x_k^i = [p_k^i, v_k^i]$ .
$u_k^i$	Control input of robot $i$ at the $k$ -th step of horizon.
$K$	Length of horizon.
$\mathcal{K}, \tilde{\mathcal{K}}$	$\mathcal{K} = \{1, 2, \dots, K\}$ and $\tilde{\mathcal{K}} = \{1, 2, \dots, K-1\}$ .
$w^{ij}$	Warning band of robot $i$ with respect to (w.r.t.) robot $j$ .
$r_{\min}$	The minimum allowable distance between any two robots.
$r'_{\min}$	$r'_{\min} = \sqrt{r_{\min}^2 + h^2 v_{\max}^2}$ .
$\mathcal{N}, \mathcal{N}^i$	The set of all robots, and $\mathcal{N}^i = \{j \mid w^{ij} < \epsilon\}$ .
$\rho^{ij}$	The repulsive coefficient of robot $i$ w.r.t. robot $j$ .
$C^i$	The objective function of robot $i$ .

is the control input,  $\mathbf{A} = \begin{bmatrix} \mathbf{I}_d & h\mathbf{I}_d \\ \mathbf{0}_d & \mathbf{I}_d \end{bmatrix}$ ,  $\mathbf{B} = \begin{bmatrix} \mathbf{0}_d \\ h\mathbf{I}_d \end{bmatrix}$  and  $h$  is the sampling interval. For brevity, the time index “(t)” is omitted wherever ambiguity is not caused in the sequel. As required in practice, both the robot’s velocity and acceleration are subjected to physical constraints. Specifically, it holds that  $\|\Theta_v v^i\| \leq v_{\max}$  and  $\|\Theta_a u^i\| \leq a_{\max}$ , where  $\Theta_v, \Theta_a$  are positive-definite matrices, and  $v_{\max}, a_{\max} > 0$  denote the maximum velocity and acceleration, respectively.

### B. Collision Avoidance

To avoid inter-robot collisions, the minimum distance allowed between any pair of robots is set to  $r_{\min} > 0$ . In other words, collision is avoided, if

$$\|p^{ij}\| = \|p^i - p^j\| \geq r_{\min} \quad (1)$$

holds, for any pair  $(i, j) \in \mathcal{N} \times \mathcal{N}$  and  $i \neq j$ .

### C. MPC-based MATG

The general MATG problem is to design control inputs  $u^i$  for each robot  $i \in \mathcal{N}$  such that it reaches the target position  $p_{\text{target}}^i \in \mathbb{R}^d$ , while avoiding collisions with other robots. As discussed in Section I, many existing work adopts the MPC-based solutions, e.g., [19], [23], where the robot trajectories are calculated by solving an optimization problem at each time step and then executed in a receding horizon fashion. At each time step  $t \geq t_0$ , the planned trajectory of robot  $i$  for the future  $K$  time steps is defined as  $\mathcal{P}^i = [p_1^i, p_2^i, \dots, p_K^i]$  where  $p_k^i$  is the planned position for time instant  $t + kh$ ,  $k \in \mathcal{K}; \mathcal{K} \triangleq \{1, 2, \dots, K\}$  and  $K$  is the length of the planning horizon. As a rule of thumb,  $K > \frac{v_{\max}}{a_{\max} h}$  is chosen such that the robot can reduce its velocity from maximum to zero within the horizon. Similar notations apply to  $v_k^i(t)$ ,  $u_k^i(t)$  as the planned velocity and acceleration. For the ease of notation,  $u^i = [u_0^i, u_1^i, \dots, u_{K-1}^i]$  and analogously for  $x^i$ . Then, the following constrained optimization problem is imposed for the MATG at time  $t > t_0$ .

**Problem 1.** At time  $t > t_0$ , the planned trajectory  $\mathcal{P}^i$  of robot  $i \in \mathcal{N}$  is the solution to the following optimization:

$$\min_{\{u^i, x^i\}} C^i(u^i, x^i) \quad (2a)$$

$$\text{s.t.} \quad \|p_k^i - p_k^j\| \geq r_{\min}, \quad \forall j \neq i, \forall k; \quad (2b)$$

$$x_k^i = \mathbf{A}x_{k-1}^i + \mathbf{B}u_{k-1}^i, \quad \forall k; \quad (2c)$$

$$\|\Theta_a u_{k-1}^i\| \leq a_{\max}, \quad \forall k; \quad (2d)$$

$$\|\Theta_v v_k^i\| \leq v_{\max}, \quad \forall k; \quad (2e)$$

where  $\forall i, k$  is an abbreviation for  $\forall i \in \mathcal{N}$  and  $\forall k \in \mathcal{K}$ ;  $C^i(\cdot)$  is the cost function to be minimized. ■

**Remark 1.** The objective function in (2a) can be of various formats, e.g.,  $C^i = \frac{1}{2} \sum_k (Q_k \|p_k^i - p_{\text{target}}^i\|^2 + R_k \|u_k^i\|^2)$  in [32], where  $Q_k, R_k > 0, k \in \mathcal{K}$ . As discussed in the sequel, a novel objective formulation is proposed in this work, which incorporates the deadlock resolution as an integral part of the optimization process. ■

Once this optimization is solved at time  $t$  and the planned trajectory is derived, the low-level feedback controller of each robot  $i$  tracks  $\mathcal{P}^i(t)$  during the time interval  $[t, t+h)$ . Consequently, it holds that  $x^i(t+h) = x_1^i(t)$  at time  $t+h$ . Afterwards, the above optimization is re-formulated given the updated system state, and solved again for the planned trajectory  $\mathcal{P}^i(t+h)$ . Each robot repeats this procedure locally until all robots reach their respective target positions. This is the common framework for most MPC-based MATG methods.

### D. Distributed Solution, Recursive Feasibility and Deadlock

Notably, constraint (2b) depends on the state of robot  $j$ , meaning that the optimization in (2) cannot be solved by robot  $i$  alone. However, a *distributed* solution without a central coordinator is desired in this work. In other words, the planned trajectories of all robots are calculated parallel and locally by each robot solving the above optimization. Although each robot can only dictate its own trajectory, it can exchange data with other robots via wireless communication, which is assumed to be bidirectional and ideal without drop-outs or delays. Furthermore, as mentioned previously in Section I, there are two aspects of Problem 1 that are of particular interest in this work: recursive feasibility and deadlock.

According to [47], the recursive feasibility of the MPC-based MATG formulated in Problem 1 is defined as follows.

**Definition 1** (Recursive Feasibility). The constrained optimization (2) is called recursive feasible, if it is feasible at time step  $t$ ,  $\forall t > t_0$ , for each robot  $i \in \mathcal{N}$ , then the new optimization at time step  $t+h$  is feasible as well. ■

Recursive feasibility ensures the safety of the resulting trajectories, namely, no inter-robot collisions would happen. However, for certain cases such as symmetric and crowded scenarios, a number of robots block each other and cannot move towards their target positions, also known as deadlocks [27], [29], [37]. Although no collisions happen, this still prohibits a successful navigation. For the MATG problem considered here, the deadlock is formally defined as follows.

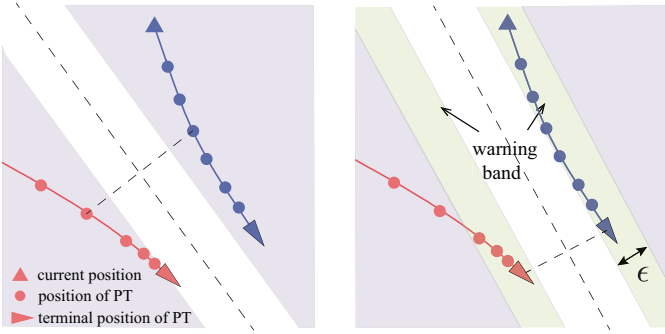


Fig. 1. Illustration of the MBVC-WB. **Left**: the shared space is split at each step of horizon; **Right**: a warning band is added for the terminal step of horizon, i.e.,  $k = K$ .

**Definition 2 (Deadlock).** Robot  $i$  is said to be in a deadlock, if its planned positions  $p_k^i$  remain static at its current position  $p^i$  and it does not reach its target, i.e.,  $p^i(t) = p_k^i(t) = p^i(t_d) \neq p_{\text{target}}^i, \forall k \in \mathcal{K}, \forall t > t_d$ , where  $t_d$  is the time when this deadlock starts. ■

The main objective of this paper is to design an MPC-based trajectory planning method such that i) the constrained optimization (2) is recursive feasible; ii) the underlying robots can detect and resolve potential deadlocks.

### III. PROPOSED SOLUTION

The complete solution is described in this section, which consists of four main steps: re-formulation of the constraint in (2b) for collision avoidance; re-formulation of the complete optimization; formal analyses of the condition for deadlocks; and finally the resolution scheme of potential deadlocks.

#### A. MBVC-WB

The collision avoidance constraint in (2b) is enforced explicitly by requiring the inter-robot distance to be more than  $r_{\min}$  at all time. However, the future states of other robots are not available at the current time. Thus, we propose to replace them with the *predetermined trajectory* of other robots.

**Definition 3 (Predetermined Trajectory (PT)).** The predetermined trajectory for robot  $i$  at time  $t$  is defined as

$$\bar{\mathcal{P}}^i(t) = [\bar{p}_1^i(t), \bar{p}_2^i(t), \dots, \bar{p}_K^i(t)],$$

where  $\bar{p}_k^i(t) = p_{k+1}^i(t-h), k \in \tilde{\mathcal{K}} \triangleq \{1, \dots, K-1\}$ , and  $\bar{p}_K^i(t) = p_K^i(t-h)$  for the end of horizon. ■

Based on the predetermined trajectory, we use the spatial separation method as in [28] to handle inter-robot collisions. As illustrated in Fig. 1, this method forms a separating hyperplane between different robots which restricts their respective motion space. Inspired by [32], we define the following modified buffered Voronoi cell with warning band (MBVC-WB) for any pair of robots  $i$  and  $j$  that  $j \neq i$ :

$$\mathcal{V}_k^{ij} = \left\{ p \mid \left( p - \frac{\bar{p}_k^i + \bar{p}_k^j}{2} \right)^T \frac{\bar{p}_k^{ij}}{\|\bar{p}_k^{ij}\|} \geq r_k^{ij} \right\}, \quad (3)$$

where  $\mathcal{V}_k^{ij}$  is the MBVC-WB for robot  $i$  w.r.t. robot  $j$  over the horizon  $k \in \mathcal{K}$ ;  $\bar{p}_k^{ij} = \bar{p}_k^i - \bar{p}_k^j$ ; the pair-wise buffer width  $r_k^{ij} = \frac{r'_{\min}}{2}$  holds for  $k \in \tilde{\mathcal{K}}$ , where

$$r'_{\min} = \sqrt{r_{\min}^2 + h^2 v_{\max}^2} \quad (4)$$

is the extended buffer width; and  $r_K^{ij} = \frac{r'_{\min}}{2} + w^{ij}$ , where  $w^{ij}$  is an optimization variable as the warning distance between robots  $i$  and  $j$ . Note that  $0 \leq w^{ij} \leq \epsilon$  with  $\epsilon$  being the maximum width of the warning band, which is generally chosen as  $\epsilon \in (\frac{r'_{\min}}{6}, \frac{r'_{\min}}{2})$ .

Like BVC in [32], an important role of MBVC-WB is that it can convert the original non-convex constraint (2b) into two convex constraints. It is not difficult to derive that the collision avoidance constraints between any pair of robots can be decoupled and become  $p_k^i \in \mathcal{V}_k^{ij}, p_k^j \in \mathcal{V}_k^{ji}$ . Via simple re-arrangements, the constraint  $p_k^i \in \mathcal{V}_k^{ij}$  can be rewritten as

$$a_k^{ijT} p_k^i \geq b_k^{ij}, \quad \forall j \neq i, k \in \tilde{\mathcal{K}}, \quad (5a)$$

$$a_K^{ijT} p_K^i \geq b_K^{ij} + w^{ij}, \quad \forall j \neq i; \quad (5b)$$

where the coefficients are given by

$$a_k^{ij} = \frac{\bar{p}_k^{ij}}{\|\bar{p}_k^{ij}\|}, \quad b_k^{ij} = a_k^{ijT} \frac{\bar{p}_k^i + \bar{p}_k^j}{2} + \frac{r'_{\min}}{2}. \quad (6)$$

Note that the above constraints in (5) can be formulated locally by each robot given the planned trajectory  $\mathcal{P}^j(t-h)$  received from its neighbors  $j$  at the previous time step  $t-h$ .

**Remark 2.** The buffered Voronoi cell (BVC) proposed in [32] is defined as follows:

$$\bar{\mathcal{V}}^{ij} = \left\{ p \mid \left( p - \frac{p^i + p^j}{2} \right)^T \frac{p^j - p^i}{\|p^j - p^i\|} \geq \frac{1}{2} r_{\min} \right\}.$$

Compared with the proposed MBVC-WB in (3), there are three main differences: (i) BVC only considers the *current* positions of robots  $i$  and  $j$ , while MBVC-WB takes into account all future positions of both robots according to their predetermined trajectories. This leads to a more accurate space separation and thus a higher utilization rate of the workspace; (ii) the buffer width  $r'_{\min}$  in (4) is velocity dependent to ensure collision avoidance between sampling points; (iii) as explained in the sequel, the warning band  $w^{ij}$  added at the end of the horizon is also included in the objective function, which allows for an effective resolution scheme for potential deadlocks. ■

**Lemma 1.** If  $p_k^i \in \mathcal{V}_k^{ij}$  holds,  $\forall i, j \in \mathcal{N}$  that  $i \neq j$ , and  $\forall k \in \mathcal{K}$ , then

$$\|p_k^{ij}\| \geq r'_{\min} \quad (7)$$

holds for the same set of  $i, j, k$  as above and  $p_k^{ij} = p_k^i - p_k^j$ . Furthermore, assuming that robots  $i, j$  move at constant velocities from  $p_k^i$  to  $p_{k+1}^i$  and from  $p_k^j$  to  $p_{k+1}^j$ , respectively, the planned trajectories  $\mathcal{P}^i(t), \mathcal{P}^j(t)$  are collision-free over the whole trajectory.

*Proof.* For robots  $i$  and  $j$ , if  $p_k^i \in \mathcal{V}_k^{ij}$  and  $p_k^j \in \mathcal{V}_k^{ji}$  hold for  $\forall k \in \mathcal{K}$ , it follows that  $a_k^{ijT} p_k^i \geq b_k^{ij}$  and  $a_k^{jiT} p_k^j \geq b_k^{ji}$ , respectively. Hence,  $a_k^{ijT} p_k^i + a_k^{jiT} p_k^j \geq b_k^{ij} + b_k^{ji}$  holds. Substituting (6) into it, it follows that  $a_k^{ijT} p_k^{ij} \geq r'_{\min}$ . Moreover,

since  $a_k^{ijT} p_k^{ij} \leq \|a_k^{ij}\| \|p_k^{ij}\| \leq \|p_k^{ij}\|$ , (7) can be derived. Furthermore, due to (7), it can be shown that

$$\|p_k^{ij}\| \geq r'_{\min} = \sqrt{r_{\min}^2 + h^2 v_{\max}^2}$$

holds for both  $k$  and  $k-1$ . Since the maximum allowed velocity is  $v_{\max}$ , it follows that  $\|p_k^i - p_{k-1}^i\| \leq h v_{\max}$  and  $\|p_k^j - p_{k-1}^j\| \leq h v_{\max}$ . Consequently, it yields

$$\begin{aligned} \|p_k^{ij}\| &\geq \sqrt{r_{\min}^2 + h^2 v_{\max}^2} \\ &\geq \sqrt{r_{\min}^2 + \frac{1}{4} \|p_k^i - p_{k-1}^i - p_k^j + p_{k-1}^j\|^2}. \end{aligned}$$

Similarly, it holds that

$$\|p_{k-1}^{ij}\| \geq \sqrt{r_{\min}^2 + \frac{1}{4} \|p_k^{ij} - p_{k-1}^{ij}\|^2}.$$

Given these conditions, Appendix A shows that the inter-robot distance during time  $[t + (k-1)h, t + kh]$  is bounded as

$$\|p_{k-1}^i + \beta(p_k^i - p_{k-1}^i) - p_{k-1}^j - \beta(p_k^j - p_{k-1}^j)\| \geq r_{\min} \quad (8)$$

where  $\beta \in [0, 1]$ . In other words, if robot  $i$  moves from  $p_{k-1}^i$  to  $p_k^i$  and robot  $j$  from  $p_{k-1}^j$  to  $p_k^j$  at constant velocities, according to (8), the minimum distance between robots  $i$  and  $j$  is larger than  $r_{\min}$  for all  $k \in \mathcal{K}$ . This completes the proof.  $\square$

### B. Complete Optimization

In addition to the inter-robot collision avoidance, the following terminal constraint is introduced to ensure the feasibility of optimization (2):

$$x_K^i \in X_e, \text{ where } X_e = \{x \mid x = \mathbf{A}x + \mathbf{B}u, u \in \mathbf{U}\}, \quad (9)$$

i.e.,  $X_e$  is a set of invariant states where feasible inputs exist for the system to remain in these states. For the simple model of double integrator, this terminal constraint (9) is equivalent to  $v_K^i = \mathbf{0}_d$ .

**Remark 3.** Note that once the above constraint is enforced, the horizon length  $K$  can be extended to *infinity* as the planned state  $x_k^i$  is identical to  $x_K^i$  if  $u_{k-1}^i = u_e$  for  $k > K$  where  $u_e$  is the control input that satisfies  $x_K^i = \mathbf{A}x_K^i + \mathbf{B}u_e$ . Thus it is also called Infinite-horizon MPC (IMPC). Notably, the infinite horizon ensures that the planned trajectory is remained to be collision-free beyond horizon such that, as the horizon recedes, a feasible trajectory is already prepared.  $\blacksquare$

Furthermore, the objective function in (2a) considered in this work consists of two parts:

$$C^i = C_w^i + C_p^i. \quad (10)$$

The first part

$$C_w^i = \sum_{j \neq i} \rho^{ij} \left( \frac{w^{ij}}{\epsilon} - \ln w^{ij} \right) \quad (11)$$

is related to the warning band in (3),  $\rho^{ij} > 0$  is a designated parameter for deadlock resolution later. Note that  $\lim_{w^{ij} \rightarrow 0^+} C_w^i = +\infty$  and  $\frac{\partial C_w^i}{\partial w^{ij}}|_{w^{ij}=\epsilon} = 0$ . The second part

$$C_p^i = \frac{1}{2} Q_K \|p_K^i - p_{\text{target}}^i\|^2 + \frac{1}{2} \sum_{k=0}^{K-1} Q_k \|p_{k+1}^i - p_k^i\|^2$$

is similar to the commonly-seen quadratic cost to penalize the distance to target and summed velocity, where  $Q_k$ ,  $k = 0, 1, \dots, K$  are the weighting parameters. In particular,  $Q_k > 0$ ,  $k \in \mathcal{K}$  and  $Q_0 = 0$ . Given the above components, the optimization in (2) is rewritten as follows:

$$\begin{aligned} \min_{u^i, x^i, w^{ij}} \quad & C^i \\ \text{s.t.} \quad & a_k^{ijT} p_k^i \geq b_k^{ij}, \forall j \neq i, k \in \tilde{\mathcal{K}}; \end{aligned} \quad (12a)$$

$$a_K^{ijT} p_K^i \geq b_K^{ij} + w^{ij}, \forall j \neq i; \quad (12b)$$

$$\epsilon \geq w^{ij} \geq 0; \quad (12c)$$

$$v_K^i = \mathbf{0}_d; \quad (12d)$$

$$(2c) - (2e).$$

where (12a)-(12c) are the re-formulated constraints for collision avoidance and (12d) is the newly introduced terminal constraint; and the objective function  $C^i$  is defined in (10).

### C. Condition for Deadlocks

The following theorem reveals that deadlock can only happen under specific conditions.

**Theorem 1.** If robot  $i \in \mathcal{N}$  belongs to a deadlock, the following condition holds:

$$Q_K (p_{\text{target}}^i - p_K^i) + \sum_{j \in \mathcal{N}^i} \rho^{ij} \delta_{ij} a_K^{ij} = 0, \quad (13)$$

where  $\mathcal{N}^i \triangleq \{j \mid w^{ij} < \epsilon\}$ ,  $\delta_{ij} = \frac{\epsilon - w^{ij}}{\epsilon w^{ij}}$ ; and  $w^{ij} = w^{ji}$  holds for  $j \in \mathcal{N}^i$ .

*Proof.* To begin with, constraint (2c) in (12) can be directly expanded as  $x_k^i = \mathbf{A}^k x_0^i + \mathbf{A}^{k-1} \mathbf{B} u_0^i + \dots + \mathbf{B} u_{k-1}^i$ ,  $k \in \mathcal{K}$ . Moreover, the domain of  $w^{ij}$  in  $C^i$  is  $w^{ij} > 0$  since  $\lim_{w^{ij} \rightarrow 0^+} C_w^i = +\infty$  and  $C_w^i$  is only convex when  $w^{ij} > 0$ . Therefore, the constraints  $w^{ij} \geq 0$  can be omitted in (12). Then, the Lagrange function of (12) is given as:

$$\begin{aligned} \mathcal{L}^i = C^i &+ \sum_{k=1}^K u \lambda_k^i (\|\Theta_a u_{k-1}^i\| - u_{\max}) \\ &+ \sum_{k=1}^K v \lambda_k^i (\|\Theta_v v_k^i\| - v_{\max}) \\ &+ \sum_{j \neq i} \lambda_k^{ij} (b_K^{ij} + w^{ij} - a_K^{ijT} p_K^i) + \sum_{j \neq i} w \lambda^{ij} (w^{ij} - \epsilon) \\ &+ \sum_{k=1}^{K-1} \sum_{j \neq i} \lambda_k^{ij} (b_k^{ij} - a_k^{ijT} p_k^i) + t \nu^i v_K^i \\ &+ \sum_{k=1}^K \nu_k^i (x_k^i - \mathbf{A}^k x_0^i - \mathbf{A}^{k-1} \mathbf{B} u_0^i - \dots - \mathbf{B} u_{K-1}^i), \end{aligned}$$

where  $u \lambda_k^i$ ,  $v \lambda_k^i$ ,  $\lambda_k^{ij}$ ,  $\nu_k^i = [\nu_k^i, \nu_k^i]$  and  $t \nu^i$  are the Lagrangian multipliers to the corresponding inequality and equality constraints, respectively.

When a deadlock happens, robot  $i$  remain static and thus  $v_{k-1}^i = \mathbf{0}_d$  and  $v_k^i = \mathbf{0}_d$  hold. It implies that both  $\|\Theta_a u_{k-1}^i\| < u_{\max}$  and  $\|\Theta_v v_k^i\| < v_{\max}$  hold. Hence, according

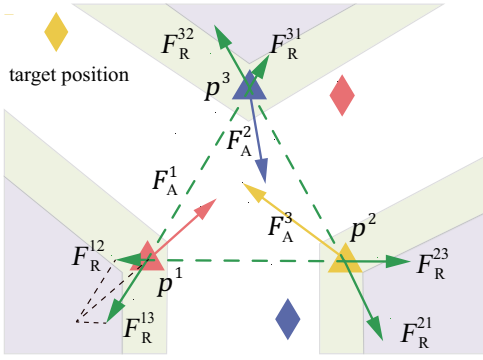


Fig. 2. Deadlock can be treated as a force equilibrium, where the attractive force from the target (in yellow, red, blue) and the repulsive forces from other robots (in green) are balanced.

to the complementary slackness condition of Karush-Kuhn-Tucker (KKT) condition [48],  $u\lambda_k^i = 0$  and  $v\lambda_k^i = 0$  hold. Furthermore, according to the stationary condition of KKT condition, the following equations are satisfied:

$$\frac{\partial \mathcal{L}^i}{\partial p_k^i} = \frac{\partial C^i}{\partial p_k^i} - \sum_{j \neq i} \lambda_K^{ij} a_K^{ij} + p\nu_k^i = 0, \quad (14a)$$

$$\frac{\partial \mathcal{L}^i}{\partial v_k^i} = \begin{cases} v\nu_k^i, & k \in \tilde{\mathcal{K}} \\ v\nu_k^i + t\nu_k^i, & k = K \end{cases} = 0, \quad (14b)$$

$$\frac{\partial \mathcal{L}^i}{\partial u_{k-1}^i} = -\mathbf{B}^T \mathbf{A}^{T^{K-k}} \nu_K^i - \mathbf{B}^T \mathbf{A}^{T^{K-k-1}} \nu_{K-1}^i - \dots - \mathbf{B}^T \nu_k^i = 0. \quad (14c)$$

$$\frac{\partial \mathcal{L}^i}{\partial w^{ij}} = \frac{\partial C^i}{\partial w^{ij}} + \lambda_K^{ij} + w\lambda^{ij} = 0, \quad (14d)$$

Given the actual value of matrices  $\mathbf{A}$ ,  $\mathbf{B}$ , condition (14c) can be rewritten as

$$\nu_K^{iT} \mathbf{A}^{K-k} \mathbf{B} + \nu_{K-1}^{iT} \mathbf{A}^{K-k-1} \mathbf{B} + \dots + \nu_k^{iT} \mathbf{B} = 0, \quad (15)$$

which directly implies that  $\nu_K^{iT} \mathbf{B} = 0$ , i.e.,  $\begin{bmatrix} p\nu_K^{iT}, v\nu_K^{iT} \end{bmatrix}^T \begin{bmatrix} 0 \\ \mathbf{I}_d \end{bmatrix} = 0$  and  $v\nu_K^i = 0$ . Combined with condition (14b),  $v\nu_k^i = 0$ ,  $k \in \mathcal{K}$  holds. Then, (15) can be further simplified as

$$(K-k)h^p \nu_K^i + (K-k-1)h^p \nu_{K-1}^i + \dots + h^p \nu_{k+1}^i = 0.$$

By setting  $k = K-1$ , it follows that  $p\nu_K^i = 0$ . Consequently, (14a) implies that

$$\frac{\partial \mathcal{L}^i}{\partial p_K^i} = \frac{\partial C^i}{\partial p_K^i} - \sum_{j \neq i} \lambda_K^{ij} a_K^{ij} = 0. \quad (16)$$

Moreover, by substituting (10) into (14d), it follows that

$$\rho^{ij} \left( -\frac{1}{w^{ij}} + \frac{1}{\epsilon} \right) + \lambda_K^{ij} + w\lambda^{ij} = 0. \quad (17)$$

Assume that there exists  $j \neq i$  such that  $w\lambda^{ij} > 0$  holds. Due to the complementary slackness,  $w^{ij} = \epsilon$  holds. Thus, (17) implies  $\lambda_K^{ij} + w\lambda^{ij} = 0$ . However, the KKT condition requires that both  $w\lambda^{ij} \geq 0$  and  $\lambda_K^{ij} \geq 0$  must hold. As a result, it follows that  $w\lambda^{ij} = 0$  and  $\lambda_K^{ij} = 0$  hold, which contradicts the

assumption that  $w\lambda^{ij} > 0$ . Thus, it holds that  $w\lambda^{ij} = 0$ ,  $\forall j \neq i$ . Furthermore, combined with (11) and (14d), it follows that

$$\lambda_K^{ij} = -\frac{\partial C^i}{\partial w^{ij}} = -\rho^{ij} \frac{w^{ij} - \epsilon}{\epsilon w^{ij}}. \quad (18)$$

By substituting (18) into (16) and combining  $p_K^i = p_{K-1}^i$ , it can be derived that

$$Q_K(p_{\text{target}}^i - p_K^i) + \sum_{j \neq i} a_K^{ij} \rho^{ij} \frac{\epsilon - w^{ij}}{\epsilon w^{ij}} = 0.$$

Moreover, if robot  $j \notin \mathcal{N}^i$ , then both  $w^{ij} = \epsilon$  and  $\frac{w^{ij} - \epsilon}{\epsilon w^{ij}} = 0$  hold, which leads to equation (13).

For  $j \in \mathcal{N}^i$ , since  $w^{ij} < \epsilon$  and  $w\lambda^{ij} = 0$ , from equation (17), it is clear that  $\lambda_K^{ij} > 0$  which derives

$$a_K^{ijT} p_K^i = b_K^{ij} + w^{ij} \quad (19)$$

via complementary slackness condition.

For robot  $j \in \mathcal{N}^i$ , it will be static as well as robot  $i$ ; otherwise, due to the change of  $\bar{p}_K^j$  and  $a_K^{ij}$ , the equalities (13) and (19) cannot be constantly held. In the following, we will show that  $w^{ji} < \epsilon$  by contradiction. Considering that  $w^{ji} \leq \epsilon$ , assume  $w^{ji} = \epsilon$  such that  $a_K^{jiT} p_K^j \geq b_K^{ji} + \epsilon$  holds. Combining this with equation (19) yields

$$b_K^{ij} + w^{ij} - a_K^{ijT} p_K^i + a_K^{jiT} p_K^j - b_K^{ji} - \epsilon \geq 0. \quad (20)$$

When robots  $i$  and  $j$  remain static, as both  $\bar{p}_K^i = p_K^i(t-h) = p_K^i$  and  $\bar{p}_K^j = p_K^j(t-h) = p_K^j$  hold, it can be derived that  $a_K^{ij} = \frac{p_K^j}{\|p_K^j\|}$  and  $b_K^{ij} = a_K^{ijT} \frac{p_K^i + p_K^j}{2} + \frac{r_{\min}^i}{2}$  also hold. Then,  $a_K^{ji} = \frac{p_K^i}{\|p_K^i\|}$  and  $b_K^{ji} = a_K^{jiT} \frac{p_K^j + p_K^i}{2} + \frac{r_{\min}^j}{2}$  can be similarly derived. Substituting them into inequality (20), we can obtain that  $w^{ij} - \epsilon \geq 0$ , which contradicts with the fact that  $w^{ij} < \epsilon$ . Therefore,  $w^{ji} < \epsilon$  must hold.

Moreover, as robot  $j$  remains static, similar to aforementioned steps,  $w\lambda^{ji} = 0$  can be derived. Combining it with  $w^{ji} < \epsilon$ , we can obtain that

$$a_K^{jiT} p_K^j = b_K^{ji} + w^{ji}. \quad (21)$$

By substituting  $a_K^{ij}$  and  $b_K^{ij}$  in addition to combining (19) and (21), we can get that  $w^{ij} = w^{ji}$ . This completes the proof.  $\square$

Based on Theorem 1,  $F_A^i = Q_K(p_{\text{target}}^i - p_K^i)$  can be regarded as the attractive force from the target and  $F_R^{ij} = \rho^{ij} \delta_{ij} a_K^{ij}$  is the repulsive force from robot  $j$ , where  $a_K^{ij}$  and  $\rho^{ij} \delta_{ij}$  are the direction and magnitude, respectively. As illustrated in Fig. 2, the deadlock condition (13) can be understood as a balance of these forces, i.e.,  $F_A^i + \sum_{j \neq i} F_R^{ij} = 0$ .

**Remark 4.** Without the warning band, i.e.,  $w^{ij} = 0$  and when the objective function  $C^i$  only includes  $C_p^i$  in (10), the necessary condition of deadlocks is rewritten as

$$Q_K(p_{\text{target}}^i - p_K^i) + \sum_{j \neq i} \lambda_K^{ij} a_K^{ij} = 0, \quad (22)$$

which indicates that the magnitude of the repulsive force  $\lambda_K^{ij}$  will be passively determined. In comparison, by introducing the warning band variable  $w^{ij}$  and including it in the objective



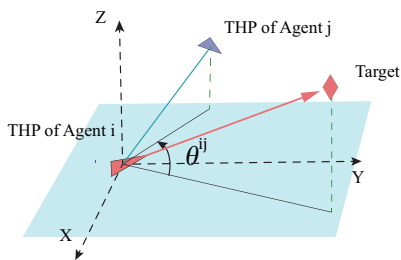


Fig. 3. Illustration of how  $\theta^{ij}$  in (23) is computed, given the terminal horizon position (THP) of robots  $i, j$  and the target.

function (11), the magnitude of the repulsive force satisfies (18) and contains  $\rho^{ij}$  as a parameter. Thus, for deadlock resolution, the repulsive forces can be actively adjusted by changing  $\rho^{ij}$ , rather than being passively determined as in (22). In other words, via adjusting  $\rho^{ij}$  properly, the condition in (13) can be falsified to resolve potential deadlocks. ■

**Remark 5.** It should be mentioned that similar analyses based on KKT conditions can be found in [30], [36], [37], which utilizes a control barrier function method. By contrast, the analysis in this theorem is built upon an MPC with predicted horizon, and many additional constraints such as (12a)-(12d) and a new term (11) related to the warning band is introduced in the optimization. Furthermore, different from [36], [37], the warning band above provides elastic safety margin when possible deadlock happens. ■

#### D. Deadlock Detection and Resolution

A potential deadlock is detected when the following condition termed *terminal overlap* holds.

**Definition 4** (Terminal Overlap). The terminal overlap for robot  $i$  happens when  $p_K^i(t) = p_K^i(t-h)$ ,  $p_K^i(t) \neq p_{\text{target}}^i$ ,  $p_K^i(t) = p_{K-1}^i(t)$ , and  $p_{K-1}^i(t) = p_{K-2}^i(t)$  hold. ■

The above condition can be analyzed as follows. According to the condition of deadlocks in (13), the summed forces at the planned terminal position  $p_K^i$  equals to zero and remains so indefinitely. Consequently, the planned terminal position at two consecutive time steps are identical, i.e.,  $p_K^i(t) = p_K^i(t-h)$  holds. Furthermore, the planned positions at the preceding steps of horizon would approach the same position as time evolves and eventually  $p_k^i$  overlaps for  $k = K, K-1, K-2$ . In other words, the condition of terminal overlap above allows an early detection of potential deadlocks by time  $(K-2)h$ . A duration longer than 2 steps can also be adopted. However, longer duration indicates later detection of deadlocks.

After a positive detection, a resolution scheme is proposed by adapting the coefficient  $\rho^{ij}$  as follows:

$$\rho^{ij} = \rho_0 e^{(\eta^i(t) \sin \theta^{ij})},$$

$$\eta^i(t) = \begin{cases} \eta^i(t-h) + \Delta\eta, & \text{if } b_{\text{TO}}^i = \text{True}; \\ 0, & \text{if } w^{ij} = \epsilon, \forall j \neq i; \\ \eta^i(t-h), & \text{otherwise;} \end{cases} \quad (23)$$

where  $\rho_0 > 0$  and  $\Delta\eta > 0$  are parameters; initially  $\eta^i(t_0) = 0$ ;  $b_{\text{TO}}^i = \text{True}$  holds when a terminal overlap happens;  $\theta^{ij}$  is

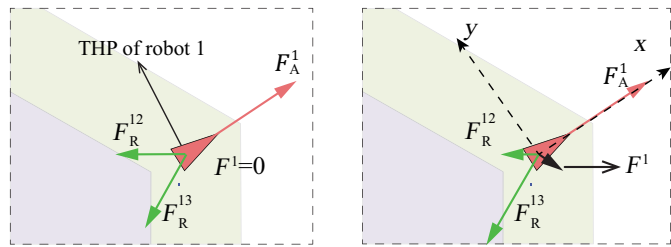


Fig. 4. **Left:** The condition of deadlocks is a force equilibrium for robot 1 in which the resulting force  $F^1 = 0$ . **Right:** After introducing the right-hand rule, the repulsive forces to the left side  $F_R^{13}$  is increased while the force to the right side  $F_R^{12}$  is decreased gradually, yielding the summed force  $F^1$  nonzero. Thus the condition of deadlocks is falsified.

defined as the angle in  $xy$  plane between the projection of line  $\bar{p}_K^i$  to its target position  $p_{\text{target}}^i$  and to  $\bar{p}_K^i$ , as illustrated in Fig. 3.

Consequently,  $\rho^{ij}$  is adapted to modify the repulsive forces by following the proposed right-hand rule. Initially,  $\rho^{ij} = \rho_0$  and no right-hand forces are introduced. If  $b_{\text{TO}}^i = \text{True}$ ,  $\eta^i$  is increased and becomes positive. Consider the following two cases: when robot  $j$  is on the left-hand side of robot  $i$ , i.e.,  $\theta^{ij} > 0$  and thus  $e^{(\eta^i(t) \sin \theta^{ij})} > 1$ . As a result, the repulsive force from robot  $j$  is increased and thus robot  $i$  moves further away from robot  $j$ ; Secondly, when robot  $j$  is on the right-hand side, robot  $i$  approaches to it. Once  $w^{ij} = \epsilon$ ,  $\forall j \neq i$  holds, it is evident that  $p_K^i$  must be outside the warning band of any others and there exists no  $j \in \mathcal{N}^i$ . Then,  $\eta^i$  returns to the initial value 0. Otherwise,  $\eta^i$  remains unchanged.

**Remark 6.** Compared with other methods proposed in [27], [32], deadlocks can only be detected *after* they happen and thus subsequently resolved “on the spot”. Via the detection mechanism described above, deadlocks can be detected and immediately resolved earlier, *before* they actually happen. As also validated by the numerical results in the sequel, this can lead to a safer and much more efficient navigation scheme, especially in crowded and high-speed scenarios. ■

#### E. The Complete Algorithm

The complete method is summarized in Algorithm 1. When the system starts, the predetermined trajectory is initialized in Line 2 as  $\bar{\mathcal{P}}^i(t_0) = [p^i(t_0), \dots, p^i(t_0)]$ . The main loop in Algorithm 1 runs as follows. First, the predetermined trajectory  $\bar{\mathcal{P}}^i$  of each robot as defined in Definition 3 is communicated with its neighboring robots in Line 5. Afterwards, the constraints for collision avoidance  $\text{cons}^i$  in (5) are derived via MBVC-WB in Line 6. Thereafter, the important coefficient  $\rho^{ij}$  adopted to the deadlock resolution is obtained. The optimization (12) is formulated with the updated constraints and solved subsequently in Line 8. Based on the result, the boolean variable related to deadlock detection  $b_{\text{TO}}^i$  is obtained in Line 9. Afterwards, the predetermined trajectory is derived from the planned one via its definition as well. Finally, the planned trajectory is executed by the low-level tracking controller of each robot. The above procedure repeats itself until all robots have reached their target positions.

---

**Algorithm 1: IMPC-DR**


---

**Input** :  $p^i(t_0)$ ,  $p_{\text{target}}^i$

```

1  $b_{\text{TO}}^i \leftarrow \text{False};$ 
2  $\overline{\mathcal{P}}^i(t_0) \leftarrow \text{InitialPT}(p^i(t_0))$ ;
3 while not all robots at target do
4   for  $i \in \mathcal{N}$  concurrently do
5      $\overline{\mathcal{P}}^j(t) \leftarrow \text{Communicate}(\overline{\mathcal{P}}^i(t))$ ;
6      $\text{cons}^i \leftarrow \text{MBVC-WB}(\overline{\mathcal{P}}^i(t), \overline{\mathcal{P}}^j(t))$ ;
7      $\rho^{ij} \leftarrow \text{DeadlockResolve}()$ ;
8      $\mathcal{P}^i(t) \leftarrow \text{Optimization}(\text{cons}^i, \rho^{ij})$ ;
9      $b_{\text{TO}}^i \leftarrow \text{DeadlockDecton}(\mathcal{P}^i(t))$ ;
10     $\overline{\mathcal{P}}^i(t+h) \leftarrow \text{GetPT}(\mathcal{P}^i(t))$ ;
11    ExecuteTrajectory( $\mathcal{P}^i(t)$ );
12   $t \leftarrow t+h$ ;
```

---

The computational complexity of the proposed algorithm is analyzed as follows. Since robot  $i$  communicates with other robots in  $\mathcal{N}$ , its corresponds to  $(K+1) \cdot N + K \cdot d$  constraints in (12) and  $(K \cdot d + N)$  real variables. The convex programming (12) at each time step can be solved in polynomial time w.r.t. the problem size, e.g., using the interior-point method. Moreover, as shown in the sequel, each robot only needs to communicate with others robot within a certain range, this further reduces the computation complexity above. The complexity of other components, such as the derivation of constraints for collision avoidance and deadlock resolution, are neglected compared with solving the optimization above.

#### IV. PROPERTY ANALYSES

This section is devoted to the property analyses of the proposed algorithm, regarding deadlocks, feasibility, and local communication.

Before moving forward, we first define the stability and instability of deadlocks in the sense of Lyapunov [49].

**Definition 5.** The deadlock equilibrium  $x_d^i$  of robot  $i$  is stable, if for any  $R > 0$ , there exists  $r > 0$  such that if  $\|x^i(t_d) - x_d^i\| < r$ , then  $\|x^i(t) - x_d^i\| < R$  for any  $t \geq t_d$ , where  $t_d$  is defined as in Definition 2. Otherwise, the deadlock equilibrium is unstable.

Note that for the cases where the multi-robot system has multiple deadlock equilibria, any deadlock equilibrium cannot be globally stable or unstable.

**Theorem 2.** Assuming that (i)  $\|p_{\text{target}}^i - p_{\text{target}}^j\| > r'_{\min} + 2\epsilon$  holds for  $\forall i \neq j$ ; and (ii) projections of the targets of three or more robots onto the horizontal plane are not collinear, holds, no stable deadlocks exist under Algorithm 1.

*Proof.* Once the condition of terminal overlap in Definition 4 holds,  $\rho^{ij} = \rho_0 e^{(\eta^i(t-h) \sin \theta^{ij})}$  is replaced by  $\rho^{ij} = \rho_0 e^{(\eta^i(t) \sin \theta^{ij})}$  where  $\eta^i(t) = \eta^i(t-h) + \Delta\eta$ . Now consider the direction from robot  $i$  to its target position as the  $x$ -axis and its orthogonal line as the  $y$ -axis, as illustrated in Fig. 4. Assume that this deadlock holds indefinitely. By Theorem 1,

the summed forces acting on robot  $i$  in  $y$ -direction are given by

$$F_y^i = \sum_{j \in \mathcal{N}^i} (-\sin \theta^{ij}) \rho_0 e^{(\eta^i(t) \sin \theta^{ij})} \delta^{ij} = 0. \quad (24)$$

Meanwhile, since this deadlock persists and  $\theta^{ij}$ ,  $\delta^{ij}$  do not change, the equilibrium condition similar to (24) already holds at time  $t-h$ , i.e.,

$$\sum_{j \in \mathcal{N}^i} (-\sin \theta^{ij}) \rho_0 e^{(\eta^i(t-h) \sin \theta^{ij})} \delta^{ij} = 0. \quad (25)$$

Combining (24) and (25), it follows that

$$F_y^i = \sum_{j \in \mathcal{N}^i} \sin \theta^{ij} (1 - e^{(\Delta\eta \sin \theta^{ij})}) \rho_0 e^{(\eta^i(t-h) \sin \theta^{ij})} \delta^{ij}.$$

Since  $\sin \theta^{ij} (1 - e^{(\Delta\eta \sin \theta^{ij})}) \leq 0$  holds for any  $\theta^{ij} \in (-\pi, \pi]$ , it follows that  $F_y^i \leq 0$ . Moreover, the equality  $F_y^i = 0$  holds if and only if  $\theta^{ij} = 0$  or  $\theta^{ij} = \pi$  holds,  $\forall j \in \mathcal{N}^i$ .

Denote by  $\mathbb{D}^i = \{j \mid j = i \text{ or } j \in \mathcal{N}^i \text{ or } \exists k \in \mathcal{N}^j, k \in \mathbb{D}^i\}$  the set of robots within the same deadlock of robot  $i$ . For robot  $j \in \mathbb{D}^i$ , it also holds that  $\theta^{jk} = 0$  or  $\theta^{jk} = \pi$ ,  $\forall k \in \mathcal{N}^j$ . In other words, the projections of  $p_K^j$  and  $p_{\text{target}}^j$  onto the  $xy$ -plane are collinear,  $\forall j \in \mathbb{D}^i$ . Since the second condition in Theorem 2 states that the projections of three or more robots' target are not collinear, the analyses can be focused on the case of two robots. Without loss of generality, these two robots are indexed by 1 and 2. By Lemma 3 in the Appendix B, their positions and targets are collinear. Therefore, as depicted in Fig. 5, a new coordinate can be constructed with the line from  $p_{\text{target}}^1$  to  $p_{\text{target}}^2$  being the  $x$ -axis and  $p_{\text{target}}^1$  being the origin. In this new coordinate, their positions are denoted by  $\hat{p}^1$  and  $\hat{p}^2$ , and their targets by  $\hat{p}_{\text{target}}^1$  and  $\hat{p}_{\text{target}}^2$ , respectively.

Following Theorem 1,  $Q_K(p_{\text{target}}^1 - p_K^1) + \rho^{12} \delta_{12} a_K^{12} = 0$  holds for robot 1; and  $Q_K(p_{\text{target}}^2 - p_K^2) + \rho^{21} \delta_{21} a_K^{21} = 0$  holds for robot 2. Additionally,  $w^{12} = w^{21}$  and  $\delta_{12} = \delta_{21}$  hold. Moreover, since both  $\theta^{12}$  and  $\theta^{21}$  are either 0 or  $\pi$ , it holds that  $\rho^{12} = \rho_0 = \rho^{21}$ . Then, since  $a_K^{21} = -a_K^{12}$  holds,  $(p_{\text{target}}^1 - p_K^1) + (p_{\text{target}}^2 - p_K^2) = 0$  holds, which leads to the following three cases as shown in Fig. 5.

Case (a): As proven in Lemma 3, the underlying deadlock is unstable for robot 1 and 2.

Case (b): Since  $\|p_K^1 - p_K^2\| = \hat{p}^1 - \hat{p}^2 \geq \hat{p}_{\text{target}}^2 - \hat{p}_{\text{target}}^1 > r'_{\min} + 2\epsilon$  holds and  $w^{12}, w^{21} \leq \epsilon$ , it can be verified that  $a_K^{12T} p_K^1 > b_K^{12} + w^{12}$  and  $a_K^{21T} p_K^2 > b_K^{21} + w^{21}$  are satisfied. According to complementary slackness condition, this yields that  $\lambda_K^{12} = \lambda_K^{21} = 0$  holds. Via (18), this further indicates the repulsive force between robots 1 and 2 are zero, i.e., the robots are attracted by their targets only.

Case (c): Clearly,  $\hat{p}^1 \leq \hat{p}_{\text{target}}^1$  and  $\hat{p}^2 \geq \hat{p}_{\text{target}}^2$  hold. Thus, similar to the previous case, it implies that  $\hat{p}^2 - \hat{p}^1 > r'_{\min} + 2\epsilon$  and the condition for deadlocks cannot hold.

To summarize, no stable deadlocks exists under the given two conditions.  $\square$

**Remark 7.** These two conditions in Theorem 2 are not restrictive as the first condition requires that the targets are at least separated by the minimum safety distance; and the



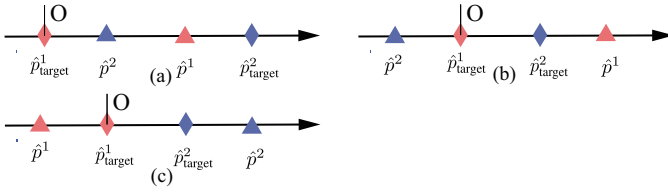


Fig. 5. Illustration of three possible cases when two robots form a deadlock.

second condition that more than three targets are strictly collinear is quite rare to happen and can be easily avoided by slightly adjusting the target positions. Moreover, as shown in Theorem 2, the only possible deadlocks are unstable, meaning that any slight deviations would drive the system away from the deadlock equilibrium. ■

**Remark 8.** The above theorem provides a theoretical guarantee for the proposed deadlock resolution scheme. In comparison, the artificial right-hand perturbation introduced in [27] and the right-hand detour points proposed in [33], [34] are heuristic and thus lacking theoretical analyses. In addition, the above methods in general introduce an instantaneous change of the control inputs whenever deadlocks happen, while the proposed scheme adapts a smooth and gradual adaptation of repulsive forces, before the potential deadlocks actually happen. Moreover, a decentralized control strategy is proposed in [36], [37], which can provably drive the robots out of any possible deadlock. It unfortunately can only be applied to less than three robots and thereby lacks the generality of the proposed deadlock resolution in this paper. ■

**Theorem 3.** *The optimization in (12) is recursively feasible under Algorithm 1.*

*Proof.* Given a feasible solution  $u_{k-1}^i(t-h)$  and  $x_k^i(t-h)$  at time  $t-h$ ,  $k \in \mathcal{K}$ , it is proven in the sequel that  $x_k^i(t) = x_{k+1}^i(t-h)$ ,  $u_{k-1}^i(t) = u_k^i(t-h)$ ,  $k \in \mathcal{K}$  and  $w^{ij}(t) = \min\{\epsilon, a_K^{ijT}(t)p_K^i(t-h) - b_K^{ij}(t)\}$  is a feasible solution at time  $t$ . As mentioned in Remark 3, it is enforced that  $x_{K+1}^i(t-h) = x_K^i(t-h)$  and  $u_K^i(t-h) = u_e$ .

First, as the result of optimization at time step  $t-h$ ,  $x_{k+1}^i(t-h)$  and  $u_k^i(t-h)$  with  $k \in \tilde{\mathcal{K}}$ , satisfy the constraints in (2c)-(2e) naturally. In addition, since  $x_K^i(t) = x_{K+1}^i(t-h) = x_K^i(t-h)$  and  $u_{K-1}^i(t) = u_K^i(t-h) = u_e$  hold,  $x_K^i(t)$  and  $u_{K-1}^i(t)$  also satisfy these constraints. In the meantime, as  $x_K^i(t) = x_{K+1}^i(t-h) = x_K^i(t-h) = x_{K-1}^i(t)$  holds, it is evident that the constraint (12d) holds as well. Second, it has been shown that, by substituting (6), (12a) is equal to

$$\|p_{k+1}^i(t-h) - p_{k+1}^j(t-h)\| \geq r'_{\min}. \quad (26)$$

As a feasible solution at time  $t-h$ ,  $p_{k+1}^i(t-h)$  satisfies  $a_{k+1}^{ijT}(t-h)p_{k+1}^i(t-h) \geq b_{k+1}^{ij}(t-h)$ ,  $\forall j \neq i, k \in \tilde{\mathcal{K}}$ . Combined with Lemma 1, the inequality in (26) holds as well as constraint (12a). Lastly, it remains to be shown that the constraints in (12b), (12c) and  $w^{ij} > 0$  hold. As a feasible solution at time  $t-h$ ,  $p_K^i(t-h)$  satisfies  $a_K^{ijT}(t-h)p_K^i(t-h) \geq$

$b_K^{ij}(t-h) + w^{ij}$ , for  $\forall j \neq i$ . Since  $w^{ij}(t-h) > 0$  holds, it implies that  $a_K^{ijT}(t-h)p_K^i(t-h) > b_K^{ij}(t-h)$  and further

$$\|p_K^i(t-h) - p_K^j(t-h)\| > r'_{\min}. \quad (27)$$

In addition, since  $p_K^i(t-h) = p_{K+1}^i(t-h) = p_K^i(t)$  holds, it yields  $\|p_K^{ij}(t)\| > r'_{\min}$ . In combination with  $a_K^{ij}(t)$  and  $b_K^{ij}(t)$  from (6), it is clear that

$$a_K^{ijT}(t)p_K^i(t) > b_K^{ij}(t). \quad (28)$$

Since  $w^{ij}(t) = \min\{\epsilon, a_K^{ijT}(t)p_K^i(t) - b_K^{ij}(t)\}$ , it implies that  $w^{ij}(t) > 0$ . Thus, the proposed solution  $x_k^i(t) = x_{k+1}^i(t-h)$ ,  $u_{k-1}^i(t) = u_k^i(t-h)$ ,  $k \in \mathcal{K}$  and  $w^{ij}(t) = \min\{\epsilon, a_K^{ijT}(t)p_K^i(t) - b_K^{ij}(t)\}$  is a feasible solution at time  $t$ .

Consequently, if the initial optimizations are feasible for all agents, then they will be feasible in a recursive way and the recursive feasibility of optimization (12) is ensured for all robots. □

**Remark 9.** It should be mentioned that the infeasible problem in motion planning was considered in previous related work, e.g., [28]. Different from [28] where the planned trajectory is represented by a Bezier-spline, this work considers a more general representation as a sequence of sampled points. Thus, the feasibility results above can be extended to more complex dynamics such as unicycles. More importantly, Theorem 2 together with Theorem 3 demonstrate that the proposed Algorithm 1 indeed solves the deadlock resolution issue with feasibility guarantee. ■

**Corollary 1.** *If the optimization (12) is initially feasible, the proposed algorithm ensures that all robots remain collision-free at all time.*

*Proof.* Since optimization (12) is evidently feasible at  $t = t_0$ , Theorem 3 shows that it remains feasible for all future time. As an important constraint (12a), the inter-robot collision avoidance is satisfied by any feasible solution. In other words, inter-robot collisions are avoided for all robots at all time. □

**Theorem 4.** *Even if robot  $i \in \mathcal{N}$  only communicates with robot  $j \in \mathcal{N}$  that satisfies  $\|p^{ij}\| \leq 2v_{\max}Kh + r'_{\min} + 2\epsilon$ , the properties mentioned in Theorems 2 and 3 still hold.*

*Proof.* To begin with, since  $\|p^{ij}\| \geq 2v_{\max}Kh + r'_{\min} + 2\epsilon$  holds,  $\|p_k^{ij}\| \geq r'_{\min} + 2\epsilon$  holds,  $\forall k \in \mathcal{K}$ . Thus, there are no repulsive forces between robots  $i, j$ , i.e., no deadlocks can appear. Second, constraints might be added or removed in (12) if robots enter or leave the above range, respectively. More specifically, when an existing constraint is removed, its feasibility remains unchanged as the problem is less constrained. On the other hand, a new constraint is added when robot  $j$  enters the communication range of robot  $i$ . Since  $\|p^{ij}\| \geq 2v_{\max}Kh + r'_{\min} + 2\epsilon$  holds, it follows that  $\|p_k^{ij}\| \geq r'_{\min}$  holds,  $k \in \mathcal{K}$ , and in turn (26) holds. Following a similar proof of Theorem 3, it can be shown that  $x_k^i(t) = x_{k+1}^i(t-h)$  and  $u_{k-1}^i(t) = u_k^i(t-h)$  is a feasible solution as well. Consequently, enforcing the local communication range above does not effect the theoretical guarantee of the complete algorithm on the deadlock-free property and recursive feasibility. □

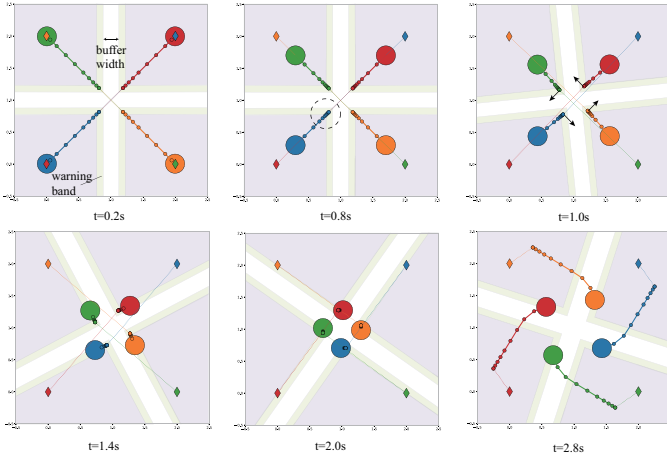


Fig. 6. The process of adopting the proposed right-hand rule where four robots want to swap their positions.

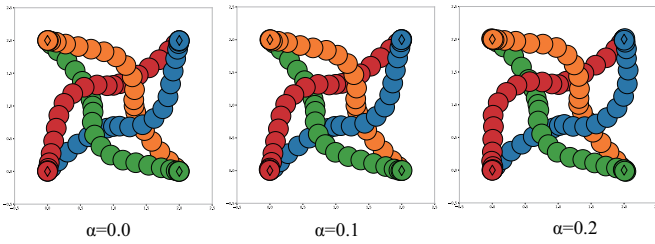


Fig. 7. Trajectories under different disturbances in a symmetric scenario.

**Remark 10.** Many related work [28], [50] requires a fully-connected communication network. By contrast, the above theorem shows that the proposed algorithm requires a local communication strategy, i.e., each robot only communicates with other robots that lie within a communication range. ■

## V. SIMULATION AND EXPERIMENT

In this section, the proposed algorithm is validated via numerical simulations and hardware experiments of large-scale multi-robot systems. The algorithms are implemented in Python3, and publicly available at <https://github.com/PKU-MACDLab/IMPC-DR>. Simulation and experiment videos can be found at <https://youtu.be/IDcXoEer068>. The convex optimizations are formulated by CVXPY [51] and solved by MOSEK [52]. The numerical simulations include some typical scenarios such as symmetric scenarios, narrow passage, position swapping, and random transitions. For the performance evaluation, our method IMPC-DR is compared with three other state-of-the-art methods: iSCP [39], DMPC [40] and BVC [32]. The implementation of iSCP and DMPC is based on [53].

### A. Typical Scenarios

To begin with, some typical scenarios in MATG are considered first. To be consistent with the model of UAVs in the experiments, the maximum velocity is set to  $v_{\max} = 1.0\text{m/s}$  and the maximum acceleration  $a_{\max} = 1.5\text{m/s}^2$ . The sampling time  $h$  is chosen as  $0.2\text{s}$  and the horizon length is set to

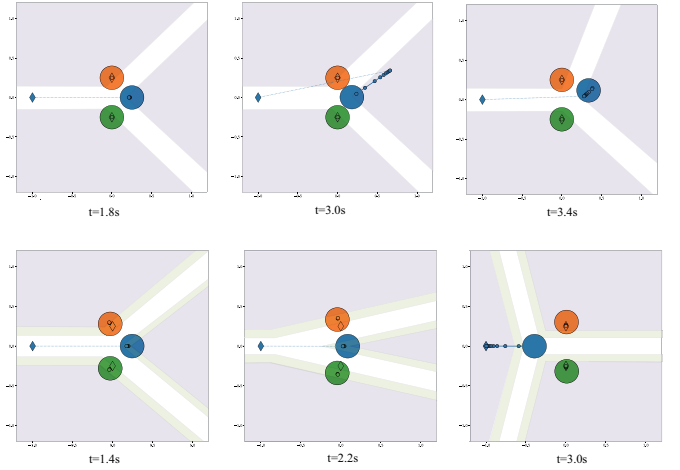


Fig. 8. The typical scenario of narrow passage where one robot passes through other two robots at target positions. **Top:** the traditional right-hand rule suffers from livelock problem. **Bottom:** the proposed method enables the robots on the two sides to temporarily move away and leave space for the middle robot to pass through.

$K = 10$  for all scenarios. The minimum inter-robot distance  $r_{\min}$  is chosen as  $0.3\text{m}$  and the width of warning band  $\epsilon$  is set as  $0.1\text{m}$ . In addition, the position penalty is set to  $Q_K = 30.0$ ,  $\rho_0 = 2.0$  and  $\eta_0 = 2.0$ .

1) *Symmetric Scenarios:* Symmetric scenarios are designed such that the initial and target positions of all robots are symmetric, which is the most common cause for deadlocks [30]. As shown in Fig. 6, four robots located in a  $2\text{m} \times 2\text{m}$  square transit to their antipodal positions. The robots approach the center point initially at  $t = 0.2\text{s}$  and the terminal positions have entered the warning band. Then at  $t = 0.8\text{s}$ , the condition for a terminal overlap holds as the terminal planned positions overlap within the warning band and consequently the proposed resolution scheme is activated. In other words, potential deadlocks are detected at this time instance, before they might happen. As a result, from time  $t = 1.0\text{s}$ , the robots begin to approach to their right-hand sides, as illustrated in Fig. 6. This adaptive process continues at  $t = 1.4\text{s}$ ,  $2.0\text{s}$  until the terminal positions leave the warning band and robots have escaped from the deadlock at time  $t = 2.8\text{s}$ . It is worth noting that without this resolution scheme, a deadlock eventually happens due to the symmetric configuration.

Furthermore, to validate the robustness of our deadlock resolution scheme, artificial forces are applied to all robots as external disturbances, e.g., to mimic wind gust. The forces follow a uniform Gaussian distribution with mean zero and standard deviation  $\sigma$ . As shown in Fig. 7, when  $\alpha = \sigma/a_{\max}$  is changed from 0 to 0.2 with the symmetric scenario described above, the proposed navigation and deadlock resolution scheme achieves the same level of deadlock resolution and similar completion time.

2) *Narrow Passage:* The second typical scenario is where a robot needs to pass through another two robots that are already at their respective target positions. This scenario emphasizes the necessity of adding the warning band to the objective function of optimization (12). As shown in Fig. 8, via the proposed algorithm, when the robot in the middle approaches

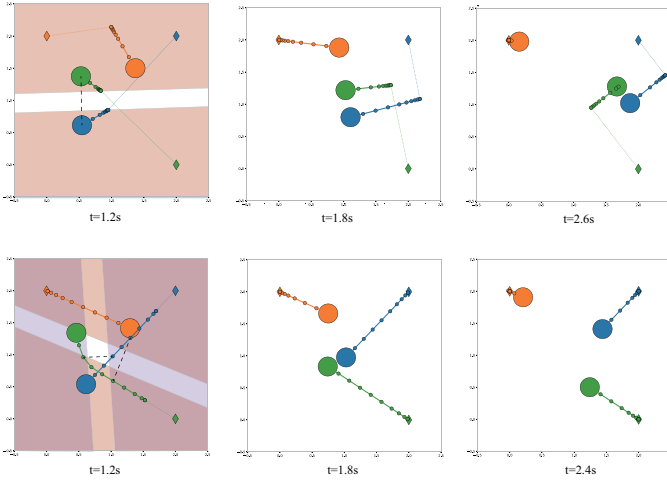


Fig. 9. The typical scenario of swapping positions. **Top:** The traditional BVC method may generate zigzag motions as BVC partitions the workspace only based on the current positions of all robots, as illustrated at  $t = 1.2s$ . **Bottom:** The proposed MBVC-WB however divides the workspace based on all future planned trajectories as shown at  $t = 1.2s$ . Consequently, much smoother trajectories are generated.

the other two robots, these two robots slowly move away from its target position to leave enough space for the middle robot to pass. Concretely, when the middle robot approach to intersection position, another two robots are compelled to enter their warning band related to this robot as shown at  $t = 1.4s$ . Intuitively, the penalty term (11) in the objective function (10) can be decreased when both robots on the side move away from their targets until they are not located within the warning band anymore at around  $t = 3.0s$ . In contrast, the heuristic approach of choosing a detour point during deadlock suffers from the livelock problem. Specifically, at time  $t = 3.0s$ , the robot in the middle chooses a temporary target position and moves away once the deadlock is detected as all robots are static. However, at around  $t = 3.4s$ , it judges that it has escaped from this deadlock, and the target position is reset to the actual one. Consequently, it comes back to the same deadlock equilibrium and this process repeats indefinitely. Similar phenomena can be found if artificial right-hand perturbations [27] are used as the heuristic method.

3) *Position Swapping:* The last scenario is where the robots swap their positions, as shown in Fig. 9. This scenario is designed to emphasize that the modified space partition constraint in (5) leads to a more accurate separation among the robots and thus a higher utility rate of the workspace. A comparison between the proposed method and the traditional BVC [32] is shown in Fig. 9 for a particular setup. Specifically, since the BVC only considers the current positions of all robots for space partition, all future positions in the planned trajectory are limited to this partition. Thus, it often leads to an overly conservative navigation structure with excessive breaking and low efficiency. In contrast, the proposed method enforces multiple space partitions based on the planned positions at each time step of horizon, i.e., the planned trajectories can be extended as how other robots plan to move. This yields a much smoother and significantly more efficient navigation

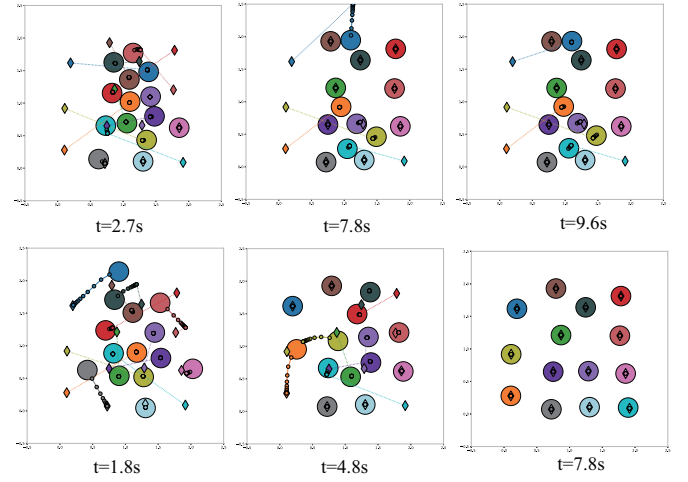


Fig. 10. Snapshots of the random transition in crowded navigation tasks in 2D by 14 robots. **Top:** The traditional right-hand rule that chooses a detour point suffers from the livelock problem, where 4 robots oscillate around the deadlock positions from time  $t = 7.8s$ . **Bottom:** The proposed method is applied to the same scenario, where potential deadlocks are resolved easily.

strategy. This difference is apparent in Fig. 9, where the robots accomplish the navigation task via the proposed method at  $t = 3.0s$  while it takes  $4.0s$  for the traditional BVC method. More detailed comparisons w.r.t. efficiency and success rate can be found in the next part.

## B. Random Transitions

To systematically compare our method with other baselines including the incremental sequential convex programming (iSCP) [39], the distributed model predictive control (DMPC) [40] and the buffered Voronoi cell (BVC) [32], the scenario of random transitions is designed, where the initial and target positions are randomly chosen in crowded 2D workspace and high-speed 3D space. In total, 100 random tests are generated for each case. In each test, a navigation task is successful if all robots arrive at their target positions within considerable long time 50.0s. Note that for optimization-based methods, if the underlying optimization is infeasible and thus no solution exists, then the task fails.

1) *Crowded 2D Workspace:* The 2D workspace is set to  $2m \times 2m$  and the number of robots ranges from 2 to 14. The safety diameter, maximum velocity, maximum acceleration, the warning band width and other parameters are selected the same as before. The only difference is that the sampling time  $h = 0.15s$  and the horizon length is set to  $K = 12$ . An example of 14 robots is shown in Fig. 10. The results in terms of success rate and infeasibility are summarized in Table II, which shows that IMPC-DR achieves 100% success rate and does not suffer from infeasibility at all. Especially for the highly crowded case of 14 robots, the success rate is much higher than other baselines. More specifically, BVC [32] with simple right-hand heuristic often suffers from the livelock problem described earlier in Section IV with a success rate only at 41%, of which an example is shown in Fig. 10. It is worth noting that when  $\epsilon$  is set to 0.1m, 14 robots within the finite space of  $2m \times 2m$  is almost at the *highest* capacity.

TABLE II  
RANDOM TRANSITIONS IN CROWDED 2D SCENARIOS

Metric	Method	Number of robots						
		2	4	6	8	10	12	14
Success	IMPC-DR	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>
	BVC [32]	100	100	97	94	83	65	41
	iSCP [39]	100	99	94	81	58	39	12
	DMPC [40]	100	99	98	91	95	73	63
Infeas.	IMPC-DR	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>
	BVC [32]	0	0	1	0	0	2	0
	iSCP [39]	0	1	6	19	20	61	88
	DMPC [40]	0	1	2	9	5	27	37

TABLE III  
TIME OF RANDOM TRANSITIONS IN CROWDED 2D SPACE

Method	Completion Time [s]						
	2	4	6	8	10	12	14
IMPC-DR	1.98	2.28	2.72	3.16	4.31	5.06	6.20
BVC [32]	2.01	2.75	3.26	4.51	6.21	7.97	10.85

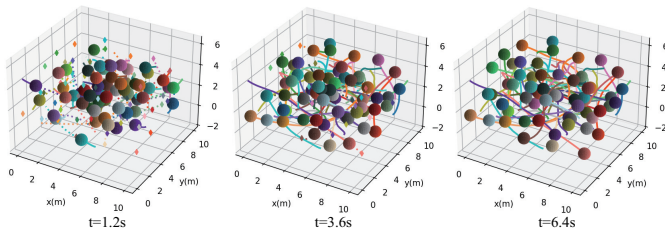


Fig. 11. Snapshots of high-speed random transitions within 3D via the proposed method by 60 robots.

One thousand random tests are conducted with a success rate of 100% and zero deadlocks as well as no livelock. This implies that the proposed method in this work has a good performance of deadlock resolution and livelock avoidance. Furthermore, the completion time is evaluated for BVC [32] and the proposed method to illustrate the efficiency of MBVC-WB. As provided in Table III, the proposed method has a significant decrease in transition time especially in a more crowded scenario.

2) *High-Speed 3D Workspaces*: In this case, the maximum velocity and acceleration are increased to 3m/s and 2m/s<sup>2</sup>, respectively; the 3D workspace is extended to 10m × 10m × 5m. The safety diameter of all robots is set to 1.0m in addition to the warning band width is extended to 0.2m. As shown in Fig. 11, 60 robots can transit at a high speed with safety guarantee by IMPC-DR in this space. Tests are performed for system sizes from 8 to 60, of which the comparisons w.r.t. the success and infeasibility rate are summarized in Table IV. It can be seen that the performance remains almost the same as the 2D case in Table II. In contrast, the performance of other baselines degraded *significantly*, mostly due to the overaggressive trajectories and slow reaction to deadlocks. This highlights the effectiveness of the proposed terminal constraints in (9).

It is worth pointing out that for the above evaluation, the

TABLE IV  
RANDOM TRANSITIONS IN HIGH-SPEED 3D SCENARIOS

Metric	Method	Number of robots						
		8	16	24	32	40	50	60
Success	IMPC-DR	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>
	BVC [32]	95	58	43	35	27	21	17
	iSCP [39]	49	5	0	0	0	0	0
	DMPC [40]	83	36	12	0	0	0	0
Infeas.	IMPC-DR	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>
	BVC [32]	5	42	57	65	73	79	83
	iSCP [39]	51	95	100	100	100	100	100
	DMPC [40]	17	64	88	100	100	100	100

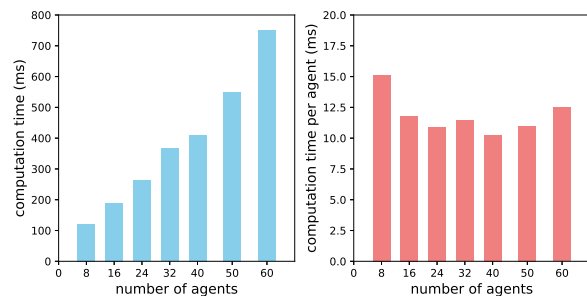


Fig. 12. **Left**: Computation time over different system sizes. **Right**: Computation time per robot.

computation of all robots is performed on one common computer. However, the planning of each robot is an independent process. Despite of being implemented in Python3 instead of C++, it can still achieve a cycle time of 0.75s for the most complex case of 60 robots in 3D. Fig. 12 illustrates how the computation time changes as the number of robots increases. It can be seen that the local computation time of each robot remains almost unchanged as the system size increases. Thus, the proposed distributed method scales well with the number of robots.

### C. Experiments

To further validate the proposed method, several experiments are performed on a nano quadrotor platform.

1) *Hardware Setup*: As shown in Fig. 13, the platform consists of several nano quadrotors named Bitcraze Crazyflie 2.1. Their states in the workspace are captured by an indoor motion capture system *OptiTrack*, of which the update frequency is 120Hz. This information is sent to the main control computer where the proposed trajectory generation algorithm is carried out for all quadrotors. The trajectory of quadrotor is fitted to a 7<sup>th</sup>-order polynomial and then sent to other quadrotors along with its state information via high-frequency radio. After receiving the position information, Kalman filter is used to estimate the current velocity and position. A feedback controller based on [54] is used to track the updated trajectory.

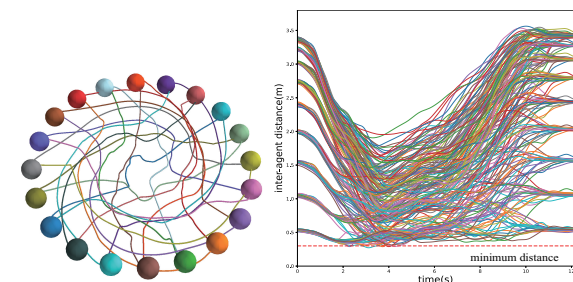
Furthermore, to avoid the inter-quadrotor air turbulence, the minimum distance between quadrotors  $r_{\min}$  is chosen as 0.3m and the width of warning band  $\epsilon$  is chosen as 0.1m. The maximum velocity and acceleration of Crazyflies are set



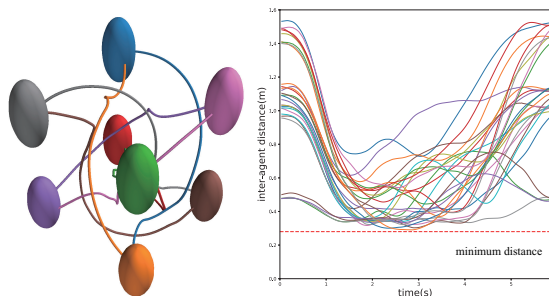


Fig. 13. Hardware platform consists of a team of Crazyflies nano quadrotors, a motion capture system, and a control computer.

to  $1\text{m/s}$  and  $1\text{m/s}^2$ , respectively, to ensure safety. Lastly, the sampling time  $h$  is set to  $0.2\text{s}$  and the horizon length to 15, to balance the control performance and the computation burden.



(a) Twenty robots transit to their antipodal positions in a circle. **Left:** Robot Trajectories. **Right:** Inter-robot distances.



(b) 8 crazyflies in a cube fly to their antipodal positions. **Left:** Robot Trajectories. **Right:** Inter-robot distances.

Fig. 14. Hardware experiments of several scenarios, which can be compared with the numerical simulations in Section V-A.

2) *Results:* The first experiment is the symmetry scenario where 20 quadrotors transit to their antipodal position in a 2D circle with radius  $1.7\text{m}$ . As shown in Fig. 14(a), the navigation task is accomplished within  $11.3\text{s}$  with smooth and safe trajectories. Due to the symmetric setup, each agent is potentially blocked by its left and right neighbors at about  $t = 2.0\text{s}$ . Via the proposed resolution scheme, the repulsive force from the left hand is increased and thus larger than the right-hand force, yielding a right-hand rotation. This pattern continues for about  $7\text{s}$  such that most of the underlying agents are close to their targets. A slight violation of the safety constraint (less than  $0.03\text{m}$ ) happens around  $2.6\text{s}$  due to the strong air turbulence in the confined space, which however can be compensated by enlarging the safety margin, as did in [40],

[24]. In the second experiment, 8 quadrotors transit to their antipodal corners in a cube of  $1\text{m}^3$ . To consider the downwash effect caused by flying quadrotors, the robots are encapsulated within an ellipsoid of diameter  $0.3\text{m}$  in  $xy$ -axis and  $0.7\text{m}$  in  $z$ -axis. The proposed IMPC-DR formulation in (12) can be easily adapted to this configuration. The final trajectories are shown in Fig. 14(b), where the right-hand rotation is also visible when the robots avoid each other.

## VI. CONCLUSION

This work has proposed a novel and effective navigation algorithm for multi-robot systems. Its core part is called infinite-horizon model predictive control with deadlock resolution. Different from many heuristic methods, it can effectively detect and resolve deadlocks online before they even happen. In addition, it provably ensures recursive feasibility of the underlying optimization at each replanning. It is a fully distributed method that requires only local communication and scales well with the number of robots. Compared with other state-of-the-art baselines, its advantages especially in crowded and high-speed scenarios are significant, as demonstrated both in simulations and hardware experiments. Future work includes extensions to the obstacle-rich environment.

## APPENDIX A

### LEMMA 2

**Lemma 2.** Consider two line segments: the line segment from  $p_1$  to  $p_2$  and the line segment from  $q_1$  to  $q_2$ , where  $p_1, p_2, q_1, q_2 \in \mathbb{R}^d$ ,  $d = 2, 3$ . If

$$\|r_1\|, \|r_2\| \geq \sqrt{r_{\min}^2 + \frac{1}{4}\|l_2 - l_1\|^2} \quad (29)$$

is satisfied for  $r_1 = q_1 - p_1$ ,  $r_2 = q_2 - p_2$ ,  $l_1 = p_2 - p_1$  and  $l_2 = q_2 - q_1$ , then

$$\|p_1 + t(p_2 - p_1) - q_1 - t(q_2 - q_1)\| \geq r_{\min} \quad (30)$$

holds,  $\forall t \in [0, 1]$ .

*Proof of Lemma 2.* It is trivial to show that the left-hand side of (30) is equivalent to

$$p_1 + t(p_2 - p_1) - q_1 - t(q_2 - q_1) = r_1 + t(r_2 - r_1).$$

Hence, it suffices to prove that

$$\|r_1 + t(r_2 - r_1)\| \geq r_{\min}, \quad \forall t \in [0, 1].$$

Introduce a function  $F(t) = \|r_1 + t(r_2 - r_1)\|^2$ ,  $t \in [0, 1]$ . Moreover, set  $F(t_{\min}) = \min_{t \in [0, 1]} F(t)$ . Consider the following three cases:

- (a)  $t_{\min} = 0$ :  $F(0) = r_1^T r_1 \geq r_{\min}^2$ .
- (b)  $t_{\min} = 1$ :  $F(1) = r_2^T r_2 \geq r_{\min}^2$ .
- (c)  $0 < t_{\min} < 1$ :  $F(t_{\min})$  is given by

$$F(t_{\min}) = \frac{r_1^T r_1 r_2^T r_2 - r_1^T r_2 r_1^T r_2}{(r_2 - r_1)^T (r_2 - r_1)},$$

where

$$t_{\min} = -\frac{r_1^T (r_2 - r_1)}{(r_2 - r_1)^T (r_2 - r_1)}. \quad (31)$$

Note that  $F(t_{\min}) \geq r_{\min}^2$  is equivalent to

$$r_1^T r_1 r_2^T r_2 - r_1^T r_2 r_1^T r_2 \geq r_{\min}^2 (r_2 - r_1)^T (r_2 - r_1),$$

which can be rewritten as

$$\begin{aligned} & (r_1^T r_1 - r_{\min}^2) (r_2^T r_2 - r_{\min}^2) \\ & \geq (r_1^T r_2 - r_{\min}^2) (r_1^T r_2 - r_{\min}^2). \end{aligned} \quad (32)$$

Now, consider the following two cases:

- (i)  $r_1^T r_2 - r_{\min}^2 \geq 0$ . Since  $t_{\min} \in (0, 1)$ , it follows from (31) that  $r_1^T r_1 \geq r_1^T r_2$  and  $r_2^T r_2 \geq r_1^T r_2$ . Thus,

$$\begin{aligned} r_1^T r_1 - r_{\min}^2 & \geq r_1^T r_2 - r_{\min}^2 \geq 0, \\ r_2^T r_2 - r_{\min}^2 & \geq r_1^T r_2 - r_{\min}^2 \geq 0, \end{aligned}$$

which induces (32).

- (ii)  $r_1^T r_2 - r_{\min}^2 < 0$ . Without loss of generality, assume that  $r_1^T r_1 \leq r_2^T r_2$ . Then, it is easy to show that  $r_1^T r_1 \geq r_{\min}^2 + \frac{1}{4} \|l_2 - l_1\|^2$ , given (29). Combining with the simple fact that  $l_2 - l_1 = q_2 - q_1 - p_2 + p_1 = r_2 - r_1$  leads to

$$\begin{aligned} r_1^T r_1 & \geq r_{\min}^2 + \frac{1}{4} \|r_2 - r_1\|^2 \\ & = r_{\min}^2 + \frac{1}{4} (r_1^T r_1 + r_2^T r_2 - 2r_1^T r_2) \\ & \geq r_{\min}^2 + \frac{1}{4} (2r_1^T r_1 - 2r_1^T r_2). \end{aligned}$$

After re-organizing the terms, we have

$$r_2^T r_2 \geq r_1^T r_1 \geq 2r_{\min}^2 - r_1^T r_2,$$

and

$$\begin{aligned} r_1^T r_1 - r_{\min}^2 & \geq r_{\min}^2 - r_1^T r_2 \geq 0, \\ r_2^T r_2 - r_{\min}^2 & \geq r_{\min}^2 - r_1^T r_2 \geq 0, \end{aligned}$$

which implies (32).

Now, the proof is completed.  $\square$

## APPENDIX B

### LEMMA 3

**Lemma 3.** Via the proposed navigation scheme, if two robots 1 and 2 are struck in deadlocks, the following properties hold:

- 1)  $p^1, p_{\text{target}}^1, p^2$  and  $p_{\text{target}}^2$  are collinear.
- 2) the deadlocks shown in Fig. 5(a) are unstable.

*Proof.* Since it is proven in Theorem 1 that the summed forces equal to zero, it implies that  $a_K^{12} = -\frac{p_{\text{target}}^1 - p_K^1}{\|p_{\text{target}}^1 - p_K^1\|} = -\frac{p_{\text{target}}^1 - p^1}{\|p_{\text{target}}^1 - p^1\|}$ . Furthermore, via (6), it holds that  $a_K^{12} = \frac{p_K^1 - p_K^2}{\|p_K^1 - p_K^2\|} = \frac{p^1 - p^2}{\|p^1 - p^2\|}$ . Thus,  $p_{\text{target}}^1, p^1$  and  $p^2$  are collinear. Via similar analyses,  $p_{\text{target}}^2, p^1$  and  $p^2$  are also collinear. Consequently,  $p^1, p_{\text{target}}^1, p^2$  and  $p_{\text{target}}^2$  are all collinear.

In the following, we will show that the deadlocks in Fig. 5(a) are unstable. As shown in Fig. 15(a), define  $\mathcal{B}_d^1(r_d) \triangleq \{p \mid \|p - p^1\| \leq r_d\}$  as the ball of radius  $r_d$  around the deadlock position of robot 1. The radius  $r_d$  is chosen such that 1) if  $\|v^1\| < \frac{2r_d}{h}$  holds, then  $\|v_k^1\| < \frac{2r_d}{h}$  and  $\|u_k^1\| < \frac{4r_d}{h^2}$  also hold, yielding that  $\|\Theta_v v_k^1\| < v_{\max}$  and  $\|\Theta_a u_{k-1}^1\| < a_{\max}$  where  $k \in \mathcal{K}$ ; 2) if  $p_k^i, p_k^i(t-h) \in \mathcal{B}_d^i(r_d)$  holds for  $i = 1, 2$  and  $k \in \mathcal{K}$ , then the constraints  $a_K^{12} p_K^1 = b_K^{12} + w^{12}$  and  $a_k^{12} p_k^1 > b_k^{12}$  hold for  $k \in \tilde{\mathcal{K}}$ .

Thereafter, the motion tendency of the underlying robots are analyzed when they are bounded by  $\mathcal{B}_d^1(r_d)$  and  $\mathcal{B}_d^2(r_d)$ . At time  $t-h$ , consider the vector

$$\begin{aligned} r_\tau &= ((p_K^2(t-h) - p_{\text{target}}^1) \times (p_K^1(t-h) - p_K^2(t-h))) \\ &\quad \times (p_K^1(t-h) - p_K^2(t-h)), \end{aligned}$$

along which the unit vector is defined as  $\tau = \frac{r_\tau}{\|r_\tau\|}$ . Consider the planning process at time  $t$ , it can be shown that for robot 1, the following holds

$$\begin{aligned} & \tau^T \frac{\partial C^1}{\partial p_K^1} \Big|_{p_K^1 = p_K^1(t-h)} \\ &= \tau^T \frac{\partial C_p^1}{\partial p_K^1} \Big|_{p_K^1 = p_K^1(t-h)} - \tau^T \frac{\partial C_w^1}{\partial p_K^1} \Big|_{p_K^1 = p_K^1(t-h)} \quad (33) \\ &= \tau^T \frac{\partial C_p^1}{\partial p_K^1} \Big|_{p_K^1 = p_K^1(t-h)}, \end{aligned}$$

where  $\tau^T \frac{\partial w^{12}}{\partial p_K^1} = \tau^T \frac{\partial \{a_K^{12} p_K^1 - b_K^{12}\}}{\partial p_K^1} = \tau^T a_K^{12} = 0$  leads to  $\tau^T \frac{\partial C_w^1}{\partial p_K^1} = \frac{\partial C_w^1}{\partial w^{12}} \tau^T \frac{\partial w^{12}}{\partial p_K^1} = 0$ . Meanwhile, since both  $\|\Theta_a u_{k-1}^i\| < u_{\max}$  and  $\|\Theta_v v_k^i\| < v_{\max}$  hold for  $k \in \mathcal{K}$ , (14a) and  $p_{v_k}^i = 0, \forall k \in \tilde{\mathcal{K}}$  can be derived similarly as in the proof of Theorem 1. Furthermore, owing to  $a_k^{12} p_k^1 > b_k^{12}$ , it can be obtained that  $\lambda_k^{12} = 0$ . Combining this with (14a), the equality  $\frac{\partial \mathcal{L}^1}{\partial p_k^1} = \frac{\partial C^1}{\partial p_k^1} = 0$  and  $Q_k(p_k^1 - p_{k+1}^1) + Q_{k-1}(p_k^1 - p_{k-1}^1) = 0$  hold,  $k \in \tilde{\mathcal{K}}$ . As a result,  $p_k^1, k = 0, 1, \dots, K$  are collinear, which induces  $Q_k \|p_k^1 - p_{k+1}^1\| = Q_{k-1} \|p_k^1 - p_{k-1}^1\|$ . Since it has been shown that  $\sum_{k=0}^{K-1} \|p_k^1 - p_{k+1}^1\| = \|p_K^1 - p_0^1\|$ , it follows that  $Q_k \|p_k^1 - p_{k+1}^1\| = 1/\sum_{k=0}^{K-1} \frac{1}{Q_k} \|p_K^1 - p_0^1\|$ . Since  $\lim_{Q_0 \rightarrow 0^+} 1/\sum_{k=0}^{K-1} \frac{1}{Q_k} = 0$ , it further implies that  $\|Q_k (p_k^1 - p_{k-1}^1)\| = 0$  and  $Q_k (p_{k+1}^1 - p_k^1) = 0$  hold for  $k = 0, 1, \dots, K-1$ . Then, since  $Q_k > 0$  exists for  $k \in \tilde{\mathcal{K}}$ , it is clear that  $p_{k+1}^1 = p_k^1$  holds for  $k \in \tilde{\mathcal{K}}$ . According to  $p_K^1 = p_{K-1}^1$ , it can be derived that

$$\begin{aligned} \frac{\partial C_p^1}{\partial p_K^1} &= Q_{K-1}(p_K^1 - p_{K-1}^1) + Q_K(p_K^1 - p_{\text{target}}^1) \\ &= Q_K(p_K^1 - p_{\text{target}}^1). \end{aligned}$$

Along with (33), the following relation holds

$$\tau^T \frac{\partial C^1}{\partial p_K^1} \Big|_{p_K^1 = p_K^1(t-h)} = Q_K \tau^T (p_K^1(t-h) - p_{\text{target}}^1).$$

As depicted in Fig. 15(a), it is evident that the angle between  $p_K^1(t-h) - p_{\text{target}}^1$  and  $\tau$  are larger than  $\frac{\pi}{2}$ , i.e., both  $\tau^T (p_K^1(t-h) - p_{\text{target}}^1) < 0$  and  $\tau^T \frac{\partial C^1}{\partial p_K^1} \Big|_{p_K^1 = p_K^1(t-h)} < 0$  hold. In addition,  $\tau^T a_K^{12} = 0$  holds by definition. Following (16), it can be derived that

$$\tau^T \frac{\partial \mathcal{L}^1}{\partial p_K^1} \Big|_{p_K^1 = p_K^1(t-h)} = \tau^T \frac{\partial C^1}{\partial p_K^1} \Big|_{p_K^1 = p_K^1(t-h)} < 0.$$

After convex programming, the Lagrange function  $\mathcal{L}^1$  will be decreased and  $\tau^T (p_K^1(t-h) - p_K^1(t-h)) > 0$  is enforced, i.e.,  $p_K^1$  has the motion tendency along the direction of  $\tau$ .

As shown in Fig. 15(b), both  $\tau_t^T p_K^1 \geq \tau_t^T p_{\text{target}}^1$  and  $\tau_t^T p_K^2 \leq \tau_t^T p_{\text{target}}^1$  hold at time  $t$ , where  $\tau_t$  is the unit vector that is perpendicular to  $p_{\text{target}}^1 - p_{\text{target}}^2$  at time  $t$ . Due to the



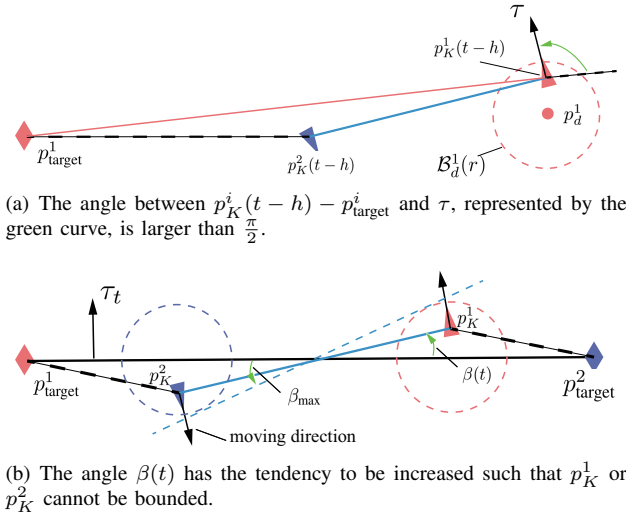


Fig. 15. The deadlock shown in Fig. 5(a) is unstable.

motion tendency described above, the angle between  $p_K^1 - p_K^2$  and  $p_{\text{target}}^2 - p_{\text{target}}^1$ , denoted as  $\beta(t)$ , is monotonically increased, i.e.,  $\beta(t+h) > \beta(t)$  when  $\beta(t) \neq 0$ . Note that  $\beta(t)$  is bounded by the maximum angle  $\beta_{\max}$ , which is the one associated with the common tangent line of these two spheres  $\mathcal{B}_d^1(r_d)$  and  $\mathcal{B}_d^2(r_d)$ .

As already proven,  $p_{k+1}^i = p_k^i$  holds for  $k \in \mathcal{K}$ , which implies  $p_1^i = p_K^i$  for  $i = 1, 2$ . Since  $p^i(t+h) = p_1^i$ , it is evident that the aforementioned property for  $p_K^i$  also holds for  $p^i$ ,  $i = 1, 2$ . Consider a situation that robots 1 and 2 are initially centrosymmetric around the point  $\frac{p_{\text{target}}^1 + p_{\text{target}}^2}{2}$  with  $p^i(0) = p_K^i(0)$ ,  $k \in \mathcal{K}$ ,  $i = 1, 2$ . Then, it is evident that these two robots will always remain centrosymmetric. Thus, given a bound arbitrarily close to the deadlock equilibrium and  $\beta(t_0)$  arbitrarily close to zero, we can find an initial state stated as above for each robot such that robots 1 and 2 will leave their corresponding bounds  $\mathcal{B}_d^1(r_d)$  and  $\mathcal{B}_d^2(r_d)$  simultaneously. According to Definition 5, the deadlocks for both robots 1 and 2 are unstable. This completes the proof.  $\square$

## REFERENCES

- [1] S.-J. Chung, A. A. Paranjape, P. Dames, S. Shen, and V. Kumar, "A survey on aerial swarm robotics," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 837–855, 2018.
- [2] Z. Zhou, J. Liu, and J. Yu, "A survey of underwater multi-robot systems," *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 1, pp. 1–18, 2022.
- [3] T. Zhang, J. Xiao, L. Li, C. Wang, and G. Xie, "Toward coordination control of multiple fish-like robots: Real-time vision-based pose estimation and tracking via deep neural networks," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 12, pp. 1964–1976, 2021.
- [4] N. Gafur, V. Yfantis, and M. Ruskowski, "Optimal scheduling and non-cooperative distributed model predictive control for multiple robotic manipulators," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 390–397.
- [5] J. Borenstein and Y. Koren, "Real-time obstacle avoidance for fast mobile robots," *IEEE Transactions on systems, Man, and Cybernetics*, vol. 19, no. 5, pp. 1179–1187, 1989.
- [6] Y. Koren and J. Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 1991, pp. 1398–1404.

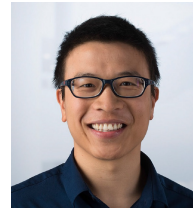
- [7] J. Alonso-Mora, A. Breitenmoser, P. Beardsley, and R. Siegwart, "Reciprocal collision avoidance for multiple car-like robots," in *IEEE International Conference on Robotics and Automation*, 2012, pp. 360–366.
- [8] J. van den Berg, M. Lin, and D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2008, pp. 1928–1935.
- [9] J. van den Berg, S. J. Guy, M. Lin, and D. Manocha, "Reciprocal n-body collision avoidance," in *14th International Symposium on Robotics Research (ISSR)*, vol. 70, 2011, pp. 3–19.
- [10] J. Huang, J. Zeng, X. Chi, K. Sreenath, Z. Liu, and H. Su, "Velocity obstacle for polytopic collision avoidance for distributed multi-robot systems," *IEEE Robotics and Automation Letters*, vol. 8, no. 6, pp. 3502–3509, 2023.
- [11] C. Tomlin, G. Pappas, and S. Sastry, "Conflict resolution for air traffic management: a study in multiagent hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 509–521, 1998.
- [12] K. Cole and A. M. Wickenheiser, "Reactive trajectory generation for multiple vehicles in unknown environments with wind disturbances," *IEEE Transactions on Robotics*, vol. 34, no. 5, pp. 1333–1348, 2018.
- [13] T. Fan, P. Long, W. Liu, and J. Pan, "Distributed multi-robot collision avoidance via deep reinforcement learning for navigation in complex scenarios," *The International Journal of Robotics Research*, vol. 39, no. 7, pp. 856–892, 2020.
- [14] G. Shi, W. Hönig, X. Shi, Y. Yue, and S.-J. Chung, "Neural-swarm2: Planning and control of heterogeneous multirobot swarms using learned interactions," *IEEE Transactions on Robotics*, pp. 1–17, 2021.
- [15] Y. F. Chen, M. Liu, M. Everett, and J. P. How, "Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 285–292.
- [16] S. Zhao, D. V. Dimarogonas, Z. Sun, and D. Bauso, "A general approach to coordination control of mobile agents with motion constraints," *IEEE Transactions on Automatic Control*, vol. 63, no. 5, pp. 1509–1516, 2018.
- [17] D. Panagou, D. M. Stipanović, and P. G. Voulgaris, "Distributed coordination control for multi-robot networks using lyapunov-like barrier functions," *IEEE Transactions on Automatic Control*, vol. 61, no. 3, pp. 617–632, 2016.
- [18] D. Panagou, M. Turpin, and V. Kumar, "Decentralized goal assignment and safe trajectory generation in multirobot networks via multiple lyapunov functions," *IEEE Transactions on Automatic Control*, vol. 65, no. 8, pp. 3365–3380, 2020.
- [19] A. Richards, J. How, and IEEE, "Decentralized model predictive control of cooperating uavs," in *IEEE Conference on Decision and Control (CDC)*, 2004, pp. 4286–4291.
- [20] S. El-Ferik, B. A. Siddiqui, and F. L. Lewis, "Distributed nonlinear mpc of multi-agent systems with data compression and random delays," *IEEE Transactions on Automatic Control*, vol. 61, no. 3, pp. 817–822, 2016.
- [21] V. Cichella, I. Kaminer, C. Walton, N. Hovakimyan, and A. M. Pascoal, "Optimal multivehicle motion planning using bernstein approximants," *IEEE Transactions on Automatic Control*, vol. 66, no. 4, pp. 1453–1467, 2021.
- [22] F. Augugliaro, A. P. Schoellig, and R. D'Andrea, "Generation of collision-free trajectories for a quadcopter fleet: A sequential convex programming approach," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 1917–1922.
- [23] D. Morgan, G. P. Subramanian, S.-J. Chung, and F. Y. Hadaegh, "Swarm assignment and trajectory optimization using variable-swarm, distributed auction assignment and sequential convex programming," *The International Journal of Robotics Research*, vol. 35, no. 10, pp. 1261–1285, 2016.
- [24] J. Park and H. J. Kim, "Online trajectory planning for multiple quadrotors in dynamic environments using relative safe flight corridor," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 659–666, 2021.
- [25] W. Hönig, J. A. Preiss, T. K. S. Kumar, G. S. Sukhatme, and N. Ayanian, "Trajectory planning for quadrotor swarms," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 856–869, 2018.
- [26] L. Campos-Macias, D. Gomez-Gutierrez, R. Aldana-Lopez, R. de la Guardia, and J. I. Parra-Vilchis, "A hybrid method for online trajectory planning of mobile robots in cluttered environments," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 935–942, 2017.
- [27] L. Wang, A. D. Ames, and M. Egerstedt, "Safety barrier certificates for collisions-free multirobot systems," *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 661–674, 2017.
- [28] J. Park, D. Kim, G. C. Kim, D. Oh, and H. J. Kim, "Online distributed trajectory planning for quadrotor swarm with feasibility guarantee using

- linear safe corridor,” *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4869–4876, 2022.
- [29] J. Alonso-Mora, J. A. DeCastro, V. Raman, D. Rus, and H. Kress-Gazit, “Reactive mission and motion planning with deadlock resolution avoiding dynamic obstacles,” *Autonomous Robots*, vol. 42, no. 4, pp. 801–824, 2018.
- [30] J. Grover, C. Liu, and K. Sycara, “Why does symmetry cause deadlocks?” *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 9746–9753, 2020.
- [31] J. Alonso-Mora, P. Beardsley, and R. Siegwart, “Cooperative collision avoidance for nonholonomic robots,” *IEEE Transactions on Robotics*, vol. 34, no. 2, pp. 404–420, 2018.
- [32] D. Zhou, Z. Wang, S. Bandyopadhyay, and M. Schwager, “Fast, on-line collision avoidance for dynamic vehicles using buffered Voronoi cells,” *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1047–1054, 2017.
- [33] A. Pierson, W. Schwarting, S. Karaman, and D. Rus, “Weighted buffered voronoi cells for distributed semi-cooperative behavior,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 5611–5617.
- [34] M. Abdullhak and A. Vardy, “Deadlock prediction and recovery for distributed collision avoidance with buffered voronoi cells,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 429–436.
- [35] K. Garg and D. Panagou, “Finite-time estimation and control for multi-aircraft systems under wind and dynamic obstacles,” *Journal of Guidance, Control, and Dynamics*, vol. 42, no. 7, pp. 1489–1505, 2019.
- [36] J. S. Grover, C. Liu, and K. Sycara, “Deadlock analysis and resolution for multi-robot systems,” in *International Workshop on the Algorithmic Foundations of Robotics*. Springer, 2020, pp. 294–312.
- [37] J. Grover, C. Liu, and K. Sycara, “The before, during, and after of multi-robot deadlock,” *The International Journal of Robotics Research*, 2022.
- [38] H. Zhu and J. Alonso-Mora, “Chance-constrained collision avoidance for mavs in dynamic environments,” *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 776–783, 2019.
- [39] Y. Chen, M. Cutler, and J. P. How, “Decoupled multiagent path planning via incremental sequential convex programming,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 5954–5961.
- [40] C. E. Luis and A. P. Schoellig, “Trajectory generation for multiagent point-to-point transitions via distributed model predictive control,” *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 375–382, 2019.
- [41] M. Zhou and M. P. Fanti, *Deadlock Resolution in Computer-Integrated Systems*. CRC Press, 2005.
- [42] J. Park and S. Reveliotis, “Deadlock avoidance in sequential resource allocation systems with multiple resource acquisitions and flexible routings,” *IEEE Transactions on Automatic Control*, vol. 46, no. 10, pp. 1572–1583, 2001.
- [43] S. Reveliotis, “Assessing and restoring “traffic-state order” in open, irreversible, dynamically routed, zone-controlled guidelpath-based transport systems,” *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 2, pp. 1286–1295, 2022.
- [44] E. Roszkowska and S. Reveliotis, “A distributed protocol for motion coordination in free-range vehicular systems,” *Automatica*, vol. 49, no. 6, pp. 1639–1653, 2013.
- [45] Y. Zhou, H. Hu, Y. Liu, S.-W. Lin, and Z. Ding, “A distributed method to avoid higher-order deadlocks in multi-robot systems,” *Automatica*, vol. 112, p. 108706, 2020.
- [46] Z. Xing, X. Chen, X. Wang, W. Wu, and R. Hu, “Collision and deadlock avoidance in multi-robot systems based on glued nodes,” *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 7, pp. 1327–1330, 2022.
- [47] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model Predictive Control: Theory, Computation, and Design*. Nob Hill Publishing, 2017.
- [48] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.
- [49] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*, Prentice Hall, 1991.
- [50] J. Tordesillas and J. P. How, “Mader: Trajectory planner in multiagent and dynamic environments,” *IEEE Transactions on Robotics*, pp. 1–14, 2021.
- [51] S. Diamond and S. Boyd, “CVXPY: A Python-embedded modeling language for convex optimization,” *Journal of Machine Learning Research*, vol. 17, no. 83, pp. 1–5, 2016.
- [52] E. D. Andersen and K. D. Andersen, *The Mosek Interior Point Optimizer for Linear Programming: An Implementation of the Homogeneous Algorithm*. Springer, 2000, pp. 197–232.
- [53] C. E. Luis, “Multiagent planning,” Website, [https://github.com/carlosluis/multiagent\\_planning](https://github.com/carlosluis/multiagent_planning).

- [54] D. Mellinger and V. Kumar, “Minimum snap trajectory generation and control for quadrotors,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 2520–2525.



**Yuda Chen** received the B.E. degree (2015) in Aerospace Engineering from Beijing Institute of Technology University, Beijing, China. He is currently a PhD candidate at the College of Engineering, Peking University, China. His research interests include multi-robot motion planning and coordination.



**Meng Guo** (M’16, SM’21) received the M.Sc. degree (2011) in System, Control, and Robotics and the Ph.D. degree (2016) in Electrical Engineering from KTH Royal Institute of Technology, Sweden. He was a postdoctoral associate with the Department of Mechanical Engineering and Materials Science, Duke University, USA. During 2018–2021, he worked as a senior research scientist on Reinforcement Learning and Planning at the Bosch Center for Artificial Intelligence (BCAI), Germany. Since 2022, he is an assistant professor at the Department of Mechanics and Engineering Science, College of Engineering, Peking University, China. His main research interests include task and motion planning for robotic systems.



**Zhongkui Li** (M’11, SM’21) received the B.S. degree in space engineering from the National University of Defense Technology, China, in 2005, and his Ph.D. degree in dynamics and control from Peking University, China, in 2010. Since 2013, Dr. Li has been with the Department of Mechanics and Engineering Science, College of Engineering, Peking University, China, where he is currently a tenured Associate Professor. His current research interests include cooperative control and planning of multi-agent systems.

Dr. Li was the recipient of the State Natural Science Award of China in 2015, the Natural Science Award of the Ministry of Education of China in 2022 and 2011, and the National Excellent Doctoral Thesis Award of China in 2012. His coauthored papers received the IET Control Theory & Applications Premium Award in 2013 and the Best Paper Award of Journal of Systems Science & Complexity in 2012. He serves as an Associate Editor of IEEE Transactions on Automatic Control, and several other journals.