Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 56-2 (2023) 2152–2157

Real-time distributed trajectory planning for mobile robots

Binh Nguyen *, **Truong Nghiem** **, **Linh Nguyen** ***, **Anh Tung Nguyen** ****, **Thang Nguyen** *

* *College of Engineering, Texas A&M University-Corpus Christi, Texas 78412, USA, (e-mail: binh.nguyen@tamucc.edu, thang.nguyen@tamucc.edu, corresponding author: Thang Nguyen)*

** *School of Informatics, Computing, and Cyber Systems, Northern Arizona University, Flagstaff, AZ 86011, USA*

*** *Institute of Innovation, Science and Sustainability, Federation University Australia, Churchill 3842, VIC, Australia*

**** *Department of Information Technology, Uppsala University, PO Box 337, SE-75105, Uppsala, Sweden*

Abstract: Efficiently planning trajectories for nonholonomic mobile robots in formation tracking is a fundamental yet challenging problem. Nonholonomic constraints, complexity in collision avoidance, and limited computing resources prevent the robots from being practically deployed in realistic applications. This paper addresses these difficulties by modeling each mobile platform as a nonholonomic motion and formulating trajectory planning as an optimization problem using model predictive control (MPC). That is, the optimization problem is subject to both nonholonomic motions and collision avoidance. To reduce computing costs in real time, the nonholonomic constraints are convexified by finding the closest nominal points to the nonholonomic motion, which are then incorporated into a convex optimization problem. Additionally, the predicted values from the previous MPC step are utilized to form linear avoidance conditions for the next step, preventing collisions among robots. The formulated optimization problem is solved by the alternating direction method of multiplier (ADMM) in a distributed manner, which makes the proposed trajectory planning method scalable. More importantly, the convergence of the proposed planning algorithm is theoretically proved while its effectiveness is validated in a synthetic environment.

Keywords: Multi-robot Systems, Distributed Model Predictive Control, Nonholonomic Trajectory Planning, Real-time Optimization, Convexification.

1. INTRODUCTION

Nowadays, autonomous wheeled mobile robots (WMR) have become ubiquitous in many applications including cooperative transportation (Yufka et al., 2010; Nguyen et al., 2020), search and coverage (Mirzaei et al., 2011) and environmental monitoring (Le et al., 2021b) to name a few. Moreover, collaboration in a group of WMRs opens possibilities to carry out sophisticated tasks that cannot be handled by a single agent (Binh et al., 2021; Nguyen et al., 2016b; Nguyen and La, 2017; Nguyen et al., 2021). However, the interaction between robots exposes challenging problems in collision avoidance, real-time processing, nonholonomic pose, and distributed implementation. Although many studies were devoted to trajectory planning for multiple WMRs (Matouš et al., 2022), due to the complicated nature of collaborations and limited resources on mobile platforms, it is still needed to further enhance the planning strategies for the robots to efficiently deal with computing costs, nonholonomic motions, and scalability. To this end, this paper presents a collision-free and distributed convexified model predictive control (MPC) method for trajectory planning of multiple nonholonomic mobile robots.

MPC has been recognized as an effective approach to determining the control inputs of multiple-input multiple-output and underactuated systems by solving constrained optimization problems that are associated with predictions of state variables (Camacho and Alba, 2013; Nguyen et al., 2022). In recent years, MPC-based methods have also been employed for controlling or planning of multiple nonholonomic mobile robots (Van Parys and Pipeleers, 2017; Alrifaee et al., 2017). For a large number of robots, crucial factors for real-time implementations of MPC-based planning algorithms are their computational resources that are required to solve a highly constrained optimization problem over a specified horizon at each time step (LaValle, 2006). It is worth noting that most kinematic models of mobile robots are nonlinear, which leads to difficulties in developing MPC-based algorithms. In this case, the nonlinear (Katriniok et al., 2019; Viana et al., 2019) and convexified (Le et al., 2021a) MPCs are two common approaches that can be used to deal with the nonlinearities entailed by the nonholonomic motion. Though it is com-

10.1016/j.ifacol.2023.10.1120

putationally expensive, the former is only able to return a local solution for a non-convex optimization problem. Likewise, the later suggests transforming the non-convex nonholonomic constraints into approximate forms in order to convert them into convexified optimization problems, making them solvable.

In the literature, it is well known that a distributed MPC technique can utilize less computational effort and provide better scalability than a centralized counterpart. Instead of solving a complicated nonlinear non-convex optimization problem directly, an optimal solution can be iteratively obtained through local sub-problems. For instance, Van Parys and Pipeleers (2016) provided a distributed MPC paradigm based on ADMM for planning trajectories of multiple linear robotic systems. Recently, Lafmejani and Berman (2021) proposed online centralized nonlinear MPC methods for controlling multiple nonholonomic robots. Nevertheless, it simplifies collision avoidance to only one horizontal direction, which consequently limits the robot maneuver. Therefore, it is still challenging to implement the distributed MPC algorithms for nonholonomic mobile robots.

In this paper, a real-time distributed trajectory planning problem for multiple nonholonomic robots using both MPC and ADMM is further investigated. Specifically, we first formulate the planning problem as an optimization problem with convexified nonholonomic constraints. We then propose to find the linearization point-based distance minimization to the nonholonomic motion before applying them to solve convex optimization problems. The proposed algorithm leads to an improvement in the accuracy of the prediction for the next steps. Differing from the Voronoi portion approach (Nguyen et al., 2016a; Le et al., 2022), our work presents the first attempt at taking advantage of the predicted values in the previous step to form linear avoidance conditions for the next step. The proposed algorithm was validated in two numerical simulations with promising results.

2. PRELIMINARIES

2.1 Description of robots and formation

Consider a group of N_v robots in a two-dimensional (2D) space. Let us define a bounded convex set $\mathcal{S} \in \mathbb{R}^2$ as the space where the robots operate. Each robot is considered a circle with radius R centered at $p_i(t) \in \mathbb{R}^2$, for every $i \in \mathcal{V} = \{1, 2, \ldots, N_v\}$. Here, the variable $t \in \mathbb{N}$ is the discrete-time instant. In the group of robots, each robot must know its destination point and can make bi-directional communication with its neighbors to establish a communication network. The network of the robots is described by a graph $\mathcal{G}(t)$ at all times. This work assumes that the communication graph $\mathcal{G}(t)$ may vary over time t but is always strongly connected. At time t, let $\mathcal{N}_i^{(t)}$ be the set of neighbors of robot i, $|\mathcal{N}_i^{(t)}|$ be the number of elements in $\mathcal{N}_i^{(t)}$, and $\mathcal{E}(t)$ be the set of (i, j) such that robot i has a communication channel with robot j (i.e., $\mathcal{E}(t)$ is the set of edges of $\mathcal{G}(t)$).

2.2 Problem formulation

Let us consider the continuous-time kinematics of wheeled mobile robots (WMRs) in the following form

$$\dot{x}_i = v_i \cos \phi_i, \tag{1a}$$
$$\dot{y}_i = v_i \sin \phi_i, \tag{1b}$$
$$\dot{\phi}_i = \omega_i, \tag{1c}$$

where $\mathbf{p}_i = [x_i, y_i]^\top$, ϕ_i, v_i and ω_i are the position, heading angle, linear velocity, and angular velocity of WMR i, respectively.

This section presents an online distributed trajectory planning problem for multiple robots. The sampling-based motion planning approach (LaValle, 2006, Chapter 5) is used to design the reference trajectory of each robot in the group with a small sampling time $\tau > 0$. Here, the reference trajectories are replanned every sampling step, the discrete time steps $t-1$, t, and $t+1 \in \mathbb{N}$ are consecutive replanning instants, and the reference trajectories are constant between two consecutive time steps t and $t+1$. At each replanning step t, we define the predicted sampled trajectory planning as a sequence $\{\mathbf{r}_{i,h}^{(t)} = \left[x_{i,h+1}^{(t)}, y_{i,h+1}^{(t)}\right]^\top\}_{h=1,2,\ldots,N_h}$, where $\mathbf{r}_{i,1}^{(t)}$ is $\mathbf{p}_i$ at sampling step t, and N_h is the number of predictive steps of the planned trajectory. Based on kinematics (1), the following predicted model is given by

$$x_{i,h+1}^{(t)} = x_{i,h}^{(t)} + \tau\, v_{i,h}^{(t)} \cos \phi_{i,h+1}^{(t)}, \tag{2a}$$
$$y_{i,h+1}^{(t)} = y_{i,h}^{(t)} + \tau\, v_{i,h}^{(t)} \sin \phi_{i,h+1}^{(t)}, \tag{2b}$$
$$\phi_{i,h+1}^{(t)} = \phi_{i,h}^{(t)} + \sigma \omega_{i,h}^{(t)}, \tag{2c}$$

where $x_{i,1}^{(t)} = x_i(t\tau)$, $y_{i,1}^{(t)} = y_i(t\tau)$ and $\phi_{i,1}^{(t)} = \phi_i(t\tau)$; $v_{i,h}^{(t)}$ and $\omega_{i,h}^{(t)}$ are predicted navigation linear and angular velocities; and $\sigma = e^\tau - 1$. In this paper, we assume that N_h is predetermined and constant at all times. A variable with the superscript $^{(t)}$ indicates its predicted values at replanning step t. If each robot can track its reference trajectory perfectly ($\mathbf{r}_{i,1}^{(t)} = \mathbf{p}_i(t+1)$ for all t), the planned trajectories should meet the following requirements: (i) robots i and j do not intersect, i.e, $\|\mathbf{r}_{i,h}^{(t)} - \mathbf{r}_{j,h}^{(t)}\|_2 \geq 2R$ for all $(i,j) \in \mathcal{V} \times (\mathcal{V} \setminus \{i\})$, $h = 0, 1, \ldots, N_h$; (ii) $\mathbf{r}_{i,h}^{(t)}$ is bounded, i.e., $\|\mathbf{r}_{i,h}^{(t)}\|_2$ is bounded by geometrical and physical limitations; and (iii) each robot reaches its destination $\mathbf{p}_{F_i} \in \mathcal{S}$ as t increases.

Furthermore, we consider the following geometrical and physical limitations on the discrete model (2a) as follows:

$$\mathbf{r}_{i,h}^{(t)} \in \mathcal{S},\ |\phi_{i,h+1}^{(t)} - \phi_{i,h}^{(t)}| \leq \tau\bar{\omega}_i,\ |v_{i,h}^{(t)}| \leq \bar{v}_i, \tag{3}$$

where $\bar{v}_i, \bar{\omega}_i > 0$ are maximum allowable values of linear and angular velocities.

3. THE PROPOSED METHOD

This section presents an MPC-based method for designing the trajectory $\mathbf{r}_{i,h}^{(t)}$ for $h = 1, \ldots, N_h$ at each time step t. For each prediction step, the planning trajectories are designed with consideration to collision avoidance among robots.

3.1 Collision avoidance

It should be noted that collision avoidance results in non-convex constraints. Thus, the obtained optimization problem might not be solved effectively. In this paper, each robot makes a replica of its neighbors' positions (virtual position) to conduct collision avoidance. This paper considers a relaxed version of collision avoidance by the following lemma:

Lemma 1. For vector functions $f_1, f_2 : t \to \mathbb{R}^2$, if there exists a unit vector $\mathbf{u} : t \to \mathbb{R}^2$ ($\|\mathbf{u}\|_2 = 1$) such that

$$(f_1 - f_2)^\top \mathbf{u} \geq R, \tag{4}$$

then $\|f_1 - f_2\|_2 \geq R$ for all t.

Proof: The proof of Lemma 1 is obtained by applying the Cauchy-Schwarz inequality to the inequality (4). ∎

Assumption 1. No collision occurs in the group of robots at the initial time.

With the help of Lemma 1, let us propose the collision avoidance condition between robots i and j in the following inequality

$$\left(\mathbf{r}_{i,h}^{(t)} - \mathbf{r}_{j,h}^{(t)}\right)^\top \mathbf{u}_{ij,h}^{(t)} \geq 2R, \tag{5}$$

where the unit vector $\mathbf{u}_{ij,h}^{(t)}$ is calculated from the predicted positions of the previous step:

$$\mathbf{u}_{ij,h}^{(t)} = \begin{cases} \dfrac{\mathbf{r}_{i,h}^{(t-1)} - \mathbf{r}_{j,h}^{(t-1)}}{\left\|\mathbf{r}_{i,h}^{(t-1)} - \mathbf{r}_{j,h}^{(t-1)}\right\|_2}, & h = 2, \ldots, N_h, \\ \dfrac{\mathbf{p}_i(t) - \mathbf{p}_j(t)}{\left\|\mathbf{p}_i(t) - \mathbf{p}_j(t)\right\|_2}, & h = 1. \end{cases} \tag{6}$$

However, the communication graph can change suddenly, the predicted positions might not be available for the new neighbors. For example, no prediction in the initial situation, or robot j does not communicate with robot i in the previous step. Thus, let $\mathbf{u}_{ij,h}^{(t)} = \mathbf{u}_{ij}^{(t)} = (\mathbf{p}_i(t) - \mathbf{p}_j(t))(\left\|\mathbf{p}_i(t) - \mathbf{p}_j(t)\right\|_2)^{-1}$ be constant at the current time step.

In light of the collision avoidance condition (5), we will propose a distributed planning algorithm for each robot. This algorithm not only ensures that each robot can reach its given destination but also guarantees collision-free trajectories in the next prediction steps.

3.2 Planning Algorithm

First of all, we take the local cost function for each robot as the following structure

$$f_i^{(t)} = f_{i,\text{des}}^{(t)} + \alpha f_{i,\text{ener}}^{(t)} + \beta_i^{(t)}, \tag{7}$$

with the weighting factor $\alpha > 0$ and

$$f_{i,\text{des}}^{(t)} = \sum_{h=1}^{N_h} \left\|\mathbf{r}_{i,h}^{(t)} - \mathbf{p}_{F_i}\right\|_2^2,$$

$$f_{i,\text{ener}}^{(t)} = \sum_{h=1}^{N_h} \left(v_{i,h}^{(t)}\right)^2 + \frac{1}{\tau^2}\left(\phi_{i,h}^{(t)} - \phi_{i,h-1}^{(t)}\right)^2,$$

where $f_{\ell,\text{des}}^{(t)}$ stands for the navigation action of the robots; $f_{\ell,\text{ener}}^{(t)}$ represents a control effort to reach the desired position; $\beta_i^{(t)}$ stands for a relaxation term for the convergence condition. We here prefer $\phi_{i,h}$ to $\omega_{i,h}$ to reduce the number of decision variables used in optimization problems afterward, i.e. the cost function $f_i^{(t)}$ does not include the variable $\omega_{i,h}$. At each time step t, let us consider the following distributed optimization problem

$$\underset{\mathbf{r}_{i,h}^{(t)}, \beta_i^{(t)}, v_{i,h}^{(t)}, \phi_{i,h}^{(t)}}{\text{minimize}} \quad \sum_{i=1}^{N_v} f_i^{(t)} \tag{8a}$$

$$\text{s.t.} \quad (2a), (2b) \text{ and } (3),\ i \in \mathcal{V},\ h \in \mathcal{H}, \tag{8b}$$

$$\left(\mathbf{r}_{i,h}^{(t)} - \mathbf{r}_{j,h}^{(t)}\right)^\top \mathbf{u}_{ij,h}^{(t)} \geq 2R + \varepsilon,\ \forall j \in \mathcal{N}_i^{(t)}, \tag{8c}$$

$$f_{i,\text{des}}^{(t)} \leq \gamma f_{i,\text{des}}^{(t-1)} + \beta_i^{(t)} \mu_i^t, \tag{8d}$$

where $\mathbf{u}_{ij,h}^{(t)}$ is given by (6); ε is a safe distance between two robots; $\gamma, \mu_i \in (0, 1)$ are given damping coefficients. The idea of using $\beta_i^{(t)}$ is from safety-critical control (Nguyen and Sreenath, 2022; Garg et al., 2022) to relax the convergence condition (8d) with damping factor γ. That is, $f_{i,\text{des}}^{(t)}$ is not required to decrease after each iteration of the MPC-based algorithm. The feasibility of the optimization problem (8) can be further discussed in Zeng et al. (2021). Based on the optimization problem (8), the path-planning algorithm 1 shows the steps of the online formation trajectory planning for multiple WMRs with collision avoidance.

Algorithm 1 Planning Algorithm

Input: Number of robots N_v, radius R, safe distance ε, predictive horizon N_h and current position $\mathbf{r}_i(t)$.

Output: $\mathbf{r}_{i,h}^{(t)}$, $h = 1, 2, \ldots, N_h$.

1: **Initiate:** at $k = 1$, set $\mathbf{r}_{i,0}^{(t)} = \mathbf{p}_i(t)$.

2: At robot i, at each time instant t, solve the optimization problem (8) to obtain $\mathbf{r}_{i,h}^{(t)}$, $i = 1, 2, \ldots, N_v$.

3: Let $\{\mathbf{r}_{i,1}^{(t)}\}_{t=0,1,\ldots}$ be the planned trajectory.

3.3 Algorithm analysis

To analyze the convergence of Algorithm 1, we assume that each robot can track its reference, i.e. $\mathbf{p}_i(t+1) = \mathbf{r}_{i,1}^{(t)}$.

Proposition 1. For $\mu \in (0, 1)$, suppose that the optimization problem (8) is feasible for all t. Then, the robot i reaches its destination $\mathbf{p}_{F_i}$ as t increases.

Proof: To begin with, let $J_{i,k} = f_{i,\text{des}}^{(k)}$. From $(8d)$, we have $J_{i,k} \leq \gamma J_{i,k-1} + \beta_i^{(k)} \mu_i^k$, where $\beta_i^{(k)}$ is a solution to the optimization problem (8). Taking the summation of both sides of the inequality from 0 to k and noting that $\beta_i^{(k)}$ is upper bounded by $\bar{\beta}$, one has $(1-\gamma)\sum_{t=0}^{k} J_{i,t} \leq J_{i,0} + \bar{\beta}\frac{1-\mu_i^k}{1-\mu_i}$. Through the limits, we have $\sum_{t=0}^{\infty} J_{i,t} \leq \frac{J_{i,0}}{1-\gamma} + \frac{\bar{\beta}}{(1-\gamma)(1-\mu_i)}$, which implies that $\lim_{t\to\infty} J_{i,t} = 0$, $\lim_{t\to\infty} \|\mathbf{r}_{i,0}^{(t)} - \mathbf{p}_{F_i}\|_2 = 0$ for all $i \in \mathcal{V}$. ∎

Unlike the other works on online distributed trajectory planning (Van Parys and Pipeleers, 2017; Park et al., 2022), our paper discusses the convergence analysis of the designed planning algorithm by introducing (8d). The presence of the term $\beta\mu^k$ plays an important role as a relaxation for the constraint $J_{i,k} \leq \gamma J_{i,k-1}$. Moreover, in the case $r_{i,N_h}^{(t-1)} \neq r_{i,0}^{(t)}$, the asymptotic convergence of $J_{i,k}$ is not guaranteed. Hence, we can select $\mu = 1$ to keep $J_{i,k}$ bounded instead of zero-convergence.

Finding a solution to the optimization problem (8) plays the most important role in Algorithm 1. However, different from the local constraints (8b) and (8d), the coupling constraint (8c) points out the difficulties in deploying the optimization (8) parallely. As a result, we will present how to solve such a problem in a distributed manner in the next section.

4. DISTRIBUTED COMPUTATION FRAMEWORK

This section presents an algorithm based on ADMM to solve the optimization problem (8) in a fully distributed manner. For each robot, let $\boldsymbol{\xi}_{ij,h}^{(t)} \in \mathbb{R}^2$ be a predicted (or virtual) position of robot j at a predictive step $h \in \mathcal{H} = \{1, 2, \ldots, N_h\}$ calculated by robot i for all $j \in \mathcal{N}_i^{(t)}$. By taking advantage of the consensus principle, the robot i can create a replica of the predicted trajectory of robot j by $\boldsymbol{\xi}_{ij}^{(t)}$. The superscript ${}^{(t)}$ will be removed to lighten the notation afterward.

At the beginning, the objective function (8a) is rewritten in the consensus form

$$\min_{\mathbf{r}_{i,h}^{(t)}, \beta_i^{(t)}, v_{i,h}^{(t)}, \phi_{i,h}^{(t)}} \sum_{i=1}^{N_v} f_i(\mathbf{r}_{i,h}, \beta_i, v_{i,h}, \omega_{i,h}) \tag{9a}$$

$$\text{s.t.} \quad \mathbf{r}_{i,h} - \boldsymbol{\xi}_{ji,h} = 0, \ \forall j \in \mathcal{N}_i, \tag{9b}$$

$$\left(\mathbf{r}_{i,h} - \boldsymbol{\xi}_{ij,h}\right)^\top \mathbf{u}_{ij,h} \geq 2R + \varepsilon, \forall j \in \mathcal{N}_i, \tag{9c}$$

$$(2a), (2b) \text{ and } (3), i \in \mathcal{V}, \ h \in \mathcal{H}. \tag{9d}$$

Note that $\boldsymbol{\xi}_{ji,h} \neq \boldsymbol{\xi}_{ij,h}$. Now, we establish affine approximations for nonlinear parts of (2a) and (2b) after each replanning step. Consider the first-order Taylor-series expansion of nonlinear terms $v_{i,h} \cos \phi_{i,h+1}$ and $v_{i,h} \sin \phi_{i,h+1}$ around $v_{i,h}^*$ and $\phi_{i,h+1}^*$ as follow

$$\begin{aligned} \boldsymbol{c}_{i,h} = {} & v_{i,h}^* \cos \phi_{i,h+1}^* + \left(v - v_{i,h}^*\right) \cos \phi_{i,h+1}^* \\ & - \left(\phi - \phi_{i,h+1}^*\right) v_{i,h}^* \sin \phi_{i,h+1}^*, \end{aligned} \tag{10a}$$

$$\begin{aligned} \boldsymbol{s}_{i,h} = {} & v_{i,h}^* \sin \phi_{i,h+1}^* + \left(v - v_{i,h}^*\right) \sin \phi_{i,h+1}^* \\ & + \left(\phi - \phi_{i,h+1}^*\right) v_{i,h}^* \cos \phi_{i,h+1}^*, \end{aligned} \tag{10b}$$

in which $v_{i,h}^*$ and $\phi_{i,h+1}^*$ are linearization points for h-predicted step determined by solving the following optimization problem: for $H - 1 \geq h \geq 1$,

$$\begin{aligned} [v_{i,h}^*, \phi_{i,h+1}^*]^\top = \operatorname*{argmin}_{v,\phi} \Big\{ & \left(x_{i,h+1}^{(t-1)} - x_{i,h}^{(t-1)} - \tau v \cos \phi\right)^2 \\ & + (y_{i,h+1}^{(t-1)} - y_{i,h}^{(t-1)} - \tau v \sin \phi)^2 \\ & + \frac{\rho_v}{2}(v - v_{i,h}^{(t-1)})^2 + \frac{\rho_\phi}{2}(\phi - \phi_{i,h+1}^{(t-1)})^2 \Big\}, \end{aligned} \tag{11}$$

with positive constants ρ_v and ρ_ϕ. It should be noted that the linearization points $v_{i,0}^*$ and $\phi_{i,1}^*$ should be the current velocity and heading angle. Differing from previous studies on MPC for multiple nonholonomic motions where the predicted values are directly used in the next linearization, our work here is to find the closest point to nonholonomic motion in prediction $v_{i,h}^*, \phi_{i,h+1}^*$ instead of predicted point $v_{i,h}^{(t-1)}, \phi_{i,h+1}^{(t-1)}$ in the previous step. Then, based on (10), the linearized constraints for (2a) and (2b) are given by

Algorithm 2 ADMM-based algorithm

Input: Number of robots N_v, radius R; primal residual error ϵ_{pri}, penalty parameter ρ, maximum ADMM iteration number $n_{\max}$, and $\xi_{ii,0}$.
Output: $\mathbf{r}_{i,h}$ for $h = 1, 2, \ldots, H$.
Initiate: $\boldsymbol{\lambda}_{ij,h}^0 = 0, \ \boldsymbol{\xi}_{ij,h}^0 = \mathbf{p}(t)$
Determine linearization points by (11)
for $n = 1, 2, \ldots, n_{\max}$ **do**
  Robot i sends $\boldsymbol{\xi}_{ij,h}^{n-1}$ to j and gathers $\boldsymbol{\xi}_{ji,h}^{n-1}$ from its neighbors; solve (15) to obtain $\boldsymbol{r}_i^n$, v_i^n and ϕ_i^n.
  Robot i sends $\frac{1}{\rho}\boldsymbol{\lambda}_{ij}^{n-1} + \mathbf{r}_i^n$ to j and gathers $\frac{1}{\rho}\boldsymbol{\lambda}_{ji}^{n-1} + \mathbf{r}_j^n$ from its neighbors; calculate $\boldsymbol{\xi}_{ij,h}^n$ by (16).
  if $\sum_{j \in \mathcal{N}_i} \sum_{h=1}^{H} \|\boldsymbol{\xi}_{ij,h}^{n+1} - \mathbf{r}_{i,h}^{n+1}\|_2 < \epsilon_{\text{pri}}$ **then**
    return $\mathbf{r}_{i,h}^{n+1}$.
  else
    Calculate $\boldsymbol{\lambda}_{ij,h}^{n+1}$ by (14c).
  end if
end for

$$x_{i,h+1} = x_{i,h} + \tau \boldsymbol{c}_{i,h}\left(v_{i,h}, \phi_{i,h}\right), \tag{12a}$$

$$y_{i,h+1} = y_{i,h} + \tau \boldsymbol{s}_{i,h}\left(v_{i,h}, \phi_{i,h}\right). \tag{12b}$$

On the other side, the augmented Lagrangian of the optimization problem (9a) is given by:

$$\mathcal{L} = \sum_{i=1}^{N_v} \mathcal{L}_i(\mathbf{r}_{i,h}, \beta_i, v_{i,h}, \phi_{i,h}, \boldsymbol{\xi}_{ji,h}, \boldsymbol{\lambda}_{ij,h}), \tag{13}$$

where $\mathcal{L}_i = f_i + \sum_{j \in \mathcal{N}_i} \sum_{h=1}^{N_h} \boldsymbol{\lambda}_{ij,h}^\top \left(\mathbf{r}_{i,h} - \boldsymbol{\xi}_{ji,h}\right) + \frac{\rho}{2} \left\|\mathbf{r}_{i,h} - \boldsymbol{\xi}_{ji,h}\right\|_2^2$.

According to Boyd et al. (2011), the ADMM-based algorithm for the convexified optimization problem (13) is formulated in

$$\begin{aligned} [\mathbf{r}^{n+1}, \boldsymbol{\beta}^{n+1}, \boldsymbol{v}^{n+1}, \boldsymbol{\phi}^{n+1}]^\top & \\ = \operatorname*{argmin}_{\text{s.t. } (3),\ (12)} & \ \mathcal{L}(\mathbf{r}, \boldsymbol{\beta}, \mathbf{v}, \boldsymbol{\phi}, \boldsymbol{\xi}^n, \boldsymbol{\lambda}^n), \end{aligned} \tag{14a}$$

$$\boldsymbol{\xi}^{n+1} = \operatorname*{argmin}_{\text{s.t. } (9c)} \mathcal{L}(\mathbf{r}^{n+1}, \boldsymbol{\beta}^{n+1}, \mathbf{v}^{n+1}, \boldsymbol{\phi}^{n+1}, \boldsymbol{\xi}, \boldsymbol{\lambda}^n), \tag{14b}$$

$$\boldsymbol{\lambda}_{ij,h}^{n+1} = \boldsymbol{\lambda}_{ij,h}^n + \rho\left(\mathbf{r}_{i,h}^{n+1} - \boldsymbol{\xi}_{ji,h}^{n+1}\right), \tag{14c}$$

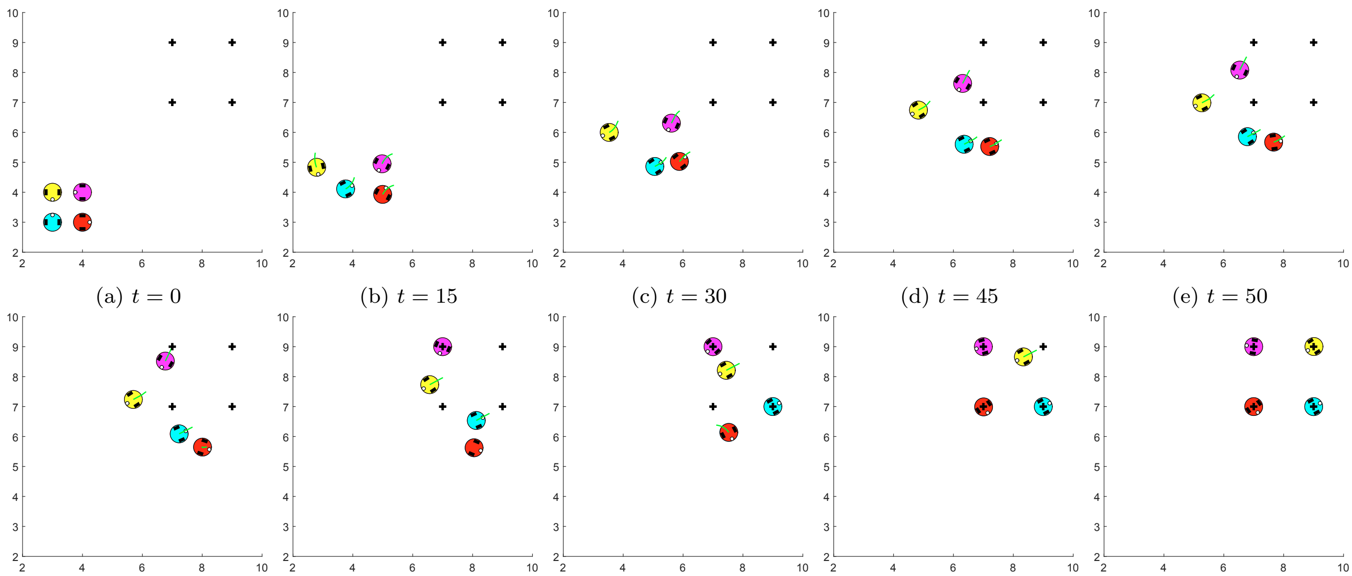
where $\mathbf{r}$, $\boldsymbol{\beta}$, $\boldsymbol{v}$ and $\boldsymbol{\phi}$ stand for combined vectorizations corresponding to $\mathbf{r}_{i,h}$, β_i, $v_{i,h}$, $\phi_{i,h}$, $\boldsymbol{\xi}_{ji,h}$, and $\boldsymbol{\lambda}_{ij,h}$. The sub-optimizations (14a) and (14b) are rewritten as

$$\begin{aligned} & [\mathbf{r}_{i,h}^{n+1}, \beta_{i,h}^{n+1}, v_{i,h}^{n+1}, \phi_{i,h}^{n+1}]^\top = \\ & \operatorname*{argmin}_{\text{s.t. } (3),(12)} f_i + \sum_{j \in \mathcal{N}_i} \sum_{h=1}^{N_h} \frac{\rho}{2} \left\|\mathbf{r}_{i,h} - \boldsymbol{\xi}_{ji,h}^n + \frac{1}{\rho}\boldsymbol{\lambda}_{ij,h}^n\right\|_2^2, \end{aligned} \tag{15}$$

$$\boldsymbol{\xi}_{ij,h}^{n+1} = \operatorname*{argmin}_{\text{s.t. } (9c)} \sum_{j \in \mathcal{N}_i} \sum_{h=1}^{N_h} \left\|\mathbf{r}_{j,h}^{n+1} - \boldsymbol{\xi}_{ij,h} + \frac{1}{\rho}\boldsymbol{\lambda}_{ji,h}^n\right\|_2^2. \tag{16}$$

The solving process is summarized in Algorithm 2. As can be seen from Algorithm 2 that *large-scale* non-convex optimization problem (OP) (8d) is solved approximately by *local* non-convex OP (11) and convex OP (14c).

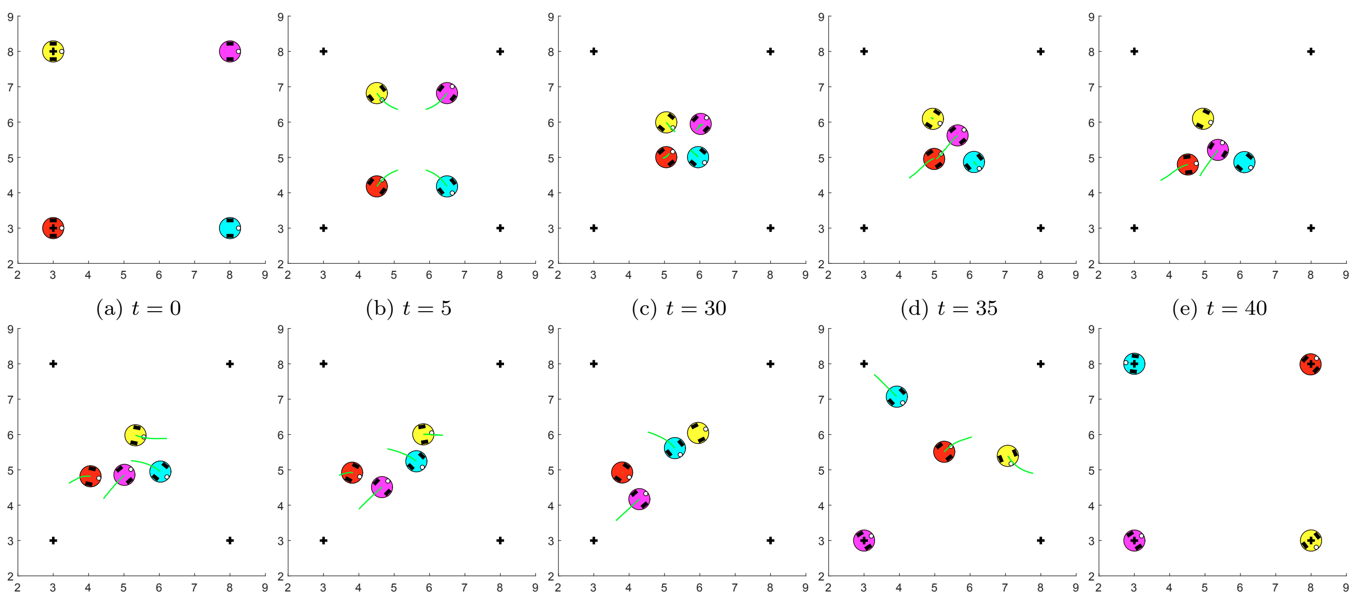
Convergence analysis: After the linearization of nonholonomic constraints (11), the optimization problem (9) has become strictly convex. Then, for both algorithms, their convergences are ensured by preceding work (Boyd et al., 2011) if the optimization problem is feasible.



(a) $t = 0$ (b) $t = 15$ (c) $t = 30$ (d) $t = 45$ (e) $t = 50$

(f) $t = 55$ (g) $t = 65$ (h) $t = 75$ (i) $t = 85$ (j) $t = 95$

Fig. 1. Snapshots of trajectories of all the robots in the first scenario



(a) $t = 0$ (b) $t = 5$ (c) $t = 30$ (d) $t = 35$ (e) $t = 40$

(f) $t = 45$ (g) $t = 50$ (h) $t = 55$ (i) $t = 75$ (j) $t = 115$

Fig. 2. Snapshots of trajectories of all the robots in the second scenario

5. NUMERICAL EXAMPLES

To demonstrate the effectiveness of the proposed algorithm, we conducted two numerical examples of 4 robots moving in an interest space $\mathbf{S} = \{(x, y) \in \mathbb{R}^2 \mid 0 \leq x, y \leq 10\}$. In the first scenario, four robots are located at one corner of the space at time step 1 as shown in Fig. 1a. However, in the second scenario, at the beginning of navigation, each robot is placed at one corner of the environment as illustrated in Fig. 2a. It is noted that the desired positions for the robots in two scenarios were set differently. We set the sampling time $\tau = 0.1[s]$, the radius of a robot $R = 0.3$, the safety distance $\varepsilon = 0.1$, and the prediction horizon $H = 8$ and $H = 10$ in both scenarios, respectively. To run the ADMM algorithm, we set $N_{\max} = 100$ and $\epsilon_{\text{pri}} = 0.01$.

In the first scenario, four robots are colored yellow, pink, red and aquamarine. The desired final positions of the four robots are interchanged, i.e., the yellow (aquamarine) robot swaps its position with that of the pink (red) robot. Furthermore, the final destinations of the four robots are also moved to another places. These swapped positions between the initial and final positions possibly cause collision among the four robots when reaching their desired places. Thanks to Algorithm 1, the four robots successfully completed their mission to reach the desired destination points as illustrated in Fig. 1j. In Figures 1c-1e, they moved very closely to others but they could still find the optimal trajectories to reach the destinations without colliding each other.

In the second scenario, the destinations of the four robots are organized such that the yellow (red) robot swaps its position with that of the aquamarine (red) robot. This setup requires that all the robots meet at the center, where collision among the robots highly possibly occurs as shown in Figures 2c-2d due to the fact that the nonholonomic constraint of each robot limits its motion. However, the robots were still be able to manage to reach the final destinations as demonstrated in Fig. 2j.

6. CONCLUDING REMARKS

The paper has proposed a distributed MPC algorithm in planning trajectories in real time for multiple nonholonomic mobile robots. Our proposed method first calculates linearization points for convexifying the next predictions, which can be used for establishing collision avoidance. It then computes the nominal points, incorporating both linear and angular velocities based on the predicted positions. Finally, a convex distributed optimization approach using ADMM is deployed for each robot, enabling real-time implementations. The simulation results demonstrate the potential of the proposed algorithm in practice.

REFERENCES

Alrifaee, B., Maczijewski, J., and Abel, D. (2017). Sequential convex programming MPC for dynamic vehicle collision avoidance. In *2017 IEEE Conf. Control Technology and Applications (CCTA)*, 2202–2207.

Binh, N.T., Dai, P.D., Quang, N.H., Ty, N.T., and Hung, N.M. (2021). Flocking control for two-dimensional multiple agents with limited communication ranges. *Int. Journal of Control*, 94(9), 2411–2418.

Boyd, S., Parikh, N., and Chu, E. (2011). *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc.

Camacho, E.F. and Alba, C.B. (2013). *Model predictive control*. Springer Science & Business Media.

Garg, K., Arabi, E., and Panagou, D. (2022). Fixed-time control under spatiotemporal and input constraints: A quadratic programming based approach. *Automatica*, 141, 110314.

Katriniok, A., Sopasakis, P., Schuurmans, M., and Patrinos, P. (2019). Nonlinear Model Predictive Control for Distributed Motion Planning in Road Intersections Using PANOC. In *2019 IEEE 58th Conf. Decision and Control (CDC)*, 5272–5278.

Lafmejani, A.S. and Berman, S. (2021). Nonlinear MPC for collision-free and deadlock-free navigation of multiple nonholonomic mobile robots. *Robotics and Autonomous Systems*, 141, 103774.

LaValle, S.M. (2006). *Planning algorithms*. Cambridge University Press.

Le, V.A., Nguyen, L., and Nghiem, T.X. (2021a). ADMM-Based Adaptive Sampling Strategy for Nonholonomic Mobile Robotic Sensor Networks. *IEEE Sensors Journal*, 21(13), 15369–15378.

Le, V.A., Nguyen, L., and Nghiem, T.X. (2021b). An Efficient Adaptive Sampling Approach for Mobile Robotic Sensor Networks using Proximal ADMM. In *2021 American Control Conf. (ACC)*, 1101–1106.

Le, V.A., Nguyen, L., and Nghiem, T.X. (2022). Multistep Predictions for Adaptive Sampling in Mobile Robotic Sensor Networks Using Proximal ADMM. *IEEE Access*, 10, 64850–64861.

Matouš, J., Varagnolo, D., Pettersen, K.Y., and Paliotta, C. (2022). Distributed MPC for Formation Path-Following of Multi-Vehicle Systems. *IFAC-PapersOnLine*, 55(13), 85–90.

Mirzaei, M., Sharifi, F., Gordon, B.W., Rabbath, C.A., and Zhang, Y.M. (2011). Cooperative multi-vehicle search and coverage problem in uncertain environments. In *2011 50th IEEE Conf. Decision and Control and European Control Conf.*, 4140–4145.

Nguyen, A.T., Nguyen, T.B., and Hong, S.K. (2020). Dynamic Event-Triggered Time-Varying Formation Control of Second-Order Dynamic Agents: Application to Multiple Quadcopters Systems. *Applied Sciences*, 10(8), 2814.

Nguyen, L.V., Kodagoda, S., Ranasinghe, R., and Dissanayake, G. (2016a). Information-Driven Adaptive Sampling Strategy for Mobile Robotic Wireless Sensor Network. *IEEE Transactions on Control Systems Technology*, 24(1), 372–379.

Nguyen, Q. and Sreenath, K. (2022). Robust Safety-Critical Control for Dynamic Robotics. *IEEE Transactions on Automatic Control*, 67(3), 1073–1088.

Nguyen, T., Han, T.T., and La, H.M. (2016b). Distributed flocking control of mobile robots by bounded feedback. In *2016 54th Annual Allerton Conf. Communication, Control, and Computing (Allerton)*, 563–568.

Nguyen, T. and La, H.M. (2017). Distributed formation control of nonholonomic mobile robots by bounded feedback in the presence of obstacles. In *2017 IEEE Int. Conf. Real-time Computing & Robotics*, 206–211.

Nguyen, T.B., Nguyen, T., Nghiem, T., Nguyen, L., Baca, J., Rangel, P., and Song, H.K. (2022). Collision-free Minimum-time Trajectory Planning for Multiple Vehicles based on ADMM. In *2022 IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, 13785–13790.

Nguyen, T.T., Trinh, M.H., Nguyen, C.V., Nguyen, N.H., and Dang, M.V. (2021). Coordination of multi-agent systems with arbitrary convergence time. *IET Control Theory & Applications*, 15(6), 900–909.

Park, J., Kim, D., Kim, G.C., Oh, D., and Kim, H.J. (2022). Online Distributed Trajectory Planning for Quadrotor Swarm With Feasibility Guarantee Using Linear Safe Corridor. *IEEE Robotics and Automation Letters*, 7(2), 4869–4876.

Van Parys, R. and Pipeleers, G. (2016). Online distributed motion planning for multi-vehicle systems. In *2016 European Control Conf. (ECC)*, 1580–1585.

Van Parys, R. and Pipeleers, G. (2017). Distributed MPC for multi-vehicle systems moving in formation. *Robotics and Autonomous Systems*, 97, 144–152.

Viana, Í.B., Kanchwala, H., and Aouf, N. (2019). Cooperative Trajectory Planning for Autonomous Driving Using Nonlinear Model Predictive Control. In *2019 IEEE Int. Conf. Connected Vehicles and Expo (ICCVE)*, 1–6.

Yufka, A., Parlaktuna, O., and Ozkan, M. (2010). Formation-based cooperative transportation by a group of non-holonomic mobile robots. In *2010 IEEE Int. Conf. Systems, Man and Cybernetics*, 3300–3307.

Zeng, J., Li, Z., and Sreenath, K. (2021). Enhancing Feasibility and Safety of Nonlinear Model Predictive Control with Discrete-Time Control Barrier Functions. In *2021 60th IEEE Conf. Decision and Control (CDC)*, 6137–6144.