Hug and Haul: Nonprehensile Object Transport by Deformable Tube-Connected Multi-Robot Systems

The International Journal of Robotics Research
XX(X):1–20
©The Author(s) 0000
Reprints and permission:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

Bin Zhang[1], Hui Zhi[1], Jose Guadalupe Romero[2], Chenguang Yang[3] and David Navarro-Alarcon[1]

Abstract

This paper presents hug and haul: a complete path-planning and tracking-control framework for object transportation by multi-robot systems (MRS) interconnected with deformable tubes. The transportation task is divided into approaching, enclosing, and transporting stages. Collision-free travelling corridors are first constructed based on the discrete path generated by rapidly-exploring random tree (RRT)*, within which smooth minimum-jerk Bézier paths are generated and reparametrized to maintain suitable inter-robot spacing. Centralized and distributed controllers are then developed to track the planned paths while preserving the tube-connected formation. An adaptive compensation term is further introduced during transport to address unknown object and environmental resistance. Input-to-state stability (ISS) analysis shows that the robot position errors remain bounded while the reference curve evolves and converge to zero after the reference stops changing. Simulations and experiments involving objects with different shapes, quantities, and spatial distributions demonstrate collision-free motion, stable tracking, and successful object transportation under both control schemes.

Keywords

Path planning, tracking control, multiple mobile robots, deformable tubes, object transportation

1 Introduction

Object transportation is a fundamental task in robotics, with applications in logistics (Koung et al., 2021; Wahrmann et al., 2019), manufacturing (Chen et al., 2018; Xuan et al., 2023), and service industries (Flores and Kecskeméthy, 2013; Chung et al., 2013). Recently, we have witnessed significant advances in the ability of robotic manipulators and humanoid robots to grasp and transport objects of different shapes and sizes (Jiao et al., 2025; Zhao et al., 2025; Levé et al., 2025), while the development of object transportation by mobile robots largely falls behind. Object transportation can be performed by a single mobile robot or multiple mobile robots, depending on the size and weight of the object. While a single robot can transport small and lightweight objects, multiple robots are often required to transport large and heavy objects that exceed the capabilities of a single robot, as shown in Fig. 1(a). Existing multi-robot transportation methods generally can be categorized into three classes: grasping, pushing-only, and caging (Tuci et al., 2018). Grasping is a prehensile manipulation that requires the objects to be physically attached to the robots. Although this method can provide stable transportation, it requires the robots to have suitable grippers and the objects to have graspable features, which limits its applicability. Unlike grasping, pushing-only and caging are non-prehensile manipulations that do not require the robots to physically attach to the objects, which allows simpler robot designs and a wider range of objects to be transported. However, existing pushing-only and caging methods are usually designed for transporting single objects, where robots are distributed around the object to push or enclose it. As shown in Fig. 1(b), when multiple objects are present, the robots may not be able to push or enclose all the objects simultaneously due to the inter-robot distances.

Regarding the above limitations, this work proposes a novel approach for object transportation by multiple mobile robots interconnected by deformable tubes, as shown in Fig. 1(c). As shown in Fig. 1(d) and (e), this new configuration not only allows robots to transport single big and heavy objects as in the existing pushing-only and caging methods, but also enables them to transport multiple small and lightweight objects simultaneously. However, the tube-connected configuration introduces new challenges for path planning and tracking control, as the robots need to maintain a certain distance (no longer than the tube length) from each other while avoiding obstacles and ensuring the stability of the tube connections. In this work, we develop a complete hug and haul framework based on the new configuration of robots to address the above challenges and enable efficient object transportation in complex environments. By hugging,

[1]Department of Mechanical Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong
[2]Departamento Académico de Ingeniería Eléctrica y Electrónica, Instituto Tecnológico Autónomo de México, CDMX 01080, Mexico
[3]Department of Computing, The Hong Kong Polytechnic University, Kowloon, Hong Kong

Corresponding author:
David Navarro-Alarcon, Department of Mechanical Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong.
Email: dnavar@polyu.edu.hk

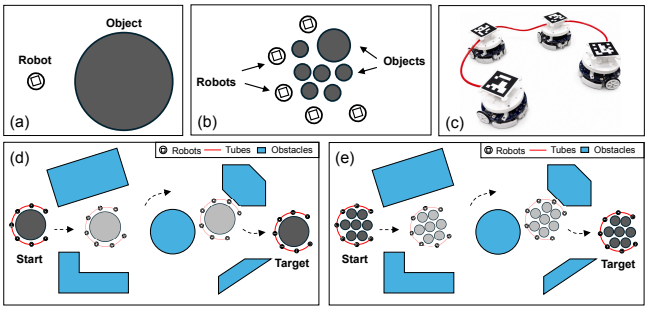


Figure 1. Conceptual illustration of object transportation by the tube-connected robots: (a) Single robot vs. single object; (b) Multiple robots vs. multiple objects; (c) Tube-connected robots; (d) Tube-connected robots transporting single objects; (e) Tube-connected robots transporting multiple objects.

robots can approach and enclose the objects, while by hauling, they can transport the objects to the desired location.

1.1 Related Work

1.1.1 Object Transport by Mobile Robots The simplest configuration for object transport by mobile robots is one robot transporting a single object. Early works can be seen in (Mason, 1986; Lynch, 1992; Lynch and Mason, 1996), where stable pushing and the quasi-static assumption were introduced to analyze the mechanics of pushing a single object by a single robot. These works laid the foundation for the later development of more advanced methods for object transport by mobile robots. Nowadays, more efforts have been devoted to object transport by multiple robots because of their advantages in load capacity, stability, and efficiency. Grasping-based manipulation is a common approach for multiple robots to transport objects. Representative works can be seen in (Wang and Schwager, 2016; Verginis et al., 2023; Sewlia et al., 2022). An observation from these works is that the robots either need extra holding structures or carry manipulators to firmly grasp the objects, which causes the robots to be more complex and expensive. Compared to grasping, non-prehensile methods allow for simpler robot designs and a wider range of objects to be transported. A multi-robot system for nonprehensile object arrangement was proposed in (Talia et al., 2023), where each robot pushes a single object to the desired position without cooperation with other robots. Caging-based methods are studied in (Wang and Kumar, 2002; Wan et al., 2012), where the robots stay enclosed around the object during transportation to confine the object. A pushing-only method optimized the contact points and achieved transportation of objects with convex shapes (Bertoncelli et al., 2022). Regarding objects with complex shapes, a distributed model predictive control (MPC) approach with formation finding was proposed in (Ebel et al., 2024), while a hybrid optimization approach was studied in a series of works (Tang et al., 2024; 2026); both approaches can transport objects with irregular shapes. A recent work proposed a novel method that uses impact interactions between the robots and the object to transport objects (Verhagen and Tumova, 2025). However, this method is limited to space as it requires a microgravity environment to eliminate the friction force.

1.1.2 Object Transport with Deformable Elements Although the above methods can transport objects with different shapes and sizes, they are largely limited to the case of a single object. For multiple objects, the robots may not be able to push or enclose all the objects simultaneously due to the inter-robot distances. One possible solution is to use deformable elements to connect the robots to offer extra constraints to the objects. Object transport by multiple robots with a deformable sheet has also been studied, where robots hold the edge of the sheet and the object is placed on it (Hu et al., 2022; Ma et al., 2026). A configuration that tethers a robot duo with a net, together with a path-planning and control framework for object gathering, was proposed in (Su et al., 2022). The tethered robot duo was also studied for object gathering, but was not validated in real-world experiments (Du et al., 2025). Another approach replaces the cable with a deformable tube for the robot duo and uses an MPC-based controller to transport single objects with different shapes (Zhi et al., 2024). A ball-string-ball structure has also been used to connect the robots, although it was validated only in simulation (Huang and Zhang, 2024).

Checking the above works, we can observe the following limitations: For pushing-only and caging-based methods, they usually focus on single-object transport and the configuration of the MRS determines that they are not suitable for transporting/gathering multiple small objects. Although adding deformable elements can help to overcome these limitations, the existing methods usually focus on the path planning and control of the transport step without considering the approaching and enclosing phases. Therefore, a complete framework for object transport by multiple robots with deformable elements is still missing, which motivates this work.

1.2 Contributions

Seeking to address the limitations of existing methods, this paper develops a complete planning and control framework for object manipulation by tube-connected multiple robots. The main contributions are summarized as follows:

1) A task-oriented and complete offline path planning pipeline to obtain reference paths for tube-connected robots to hug and haul objects. Reference paths include approaching, enclosing, and transporting stages, fulfilling specific task requirements in different stages.
2) Centralized and distributed controllers based on curve representations for robots to hug and haul objects with unknown object and environment resistance while maintaining safe inter-robot spacing.
3) Rigorous input-to-state stability analysis of the proposed method. Extensive validations via both numerical simulations and hardware experiments for single, clustered, and sparse object configurations.

Compared with existing methods, the proposed framework has the following advantages:

1) It offers a complete path planning pipeline covering the approaching, enclosing, and transporting stages. Different stages share a similar path planning

structure, which reduces the complexity of the whole algorithm.
2) Unlike many existing methods formulating control based on a contact model and object dynamics, the proposed control method starts from formation control without requiring knowledge of the contact forces and object properties. This setup frees the method from measuring physical properties that are generally unknown in practical scenarios, such as mass and friction coefficients, thus giving the method more flexibility.
3) The tube connections endow the methods with versatility in handling various object configurations, such as single, clustered, and sparse.

To the best of our knowledge, this is the first work that presents a complete framework for object transport by multiple robots interconnected with deformable tubes.

1.3 Organization

The remainder of this paper is organized as follows: Section 2 introduces the preliminaries and overview of this work; Section 3 presents the details of the path planning for the robots; Section 4 presents the controller design and stability analysis; Section 5 shows the results that validate the proposed method and the comparison and ablation study; Finally, Section 6 concludes this work.

2 Preliminaries

2.1 Notation

This manuscript adopts standard notation. The bold letters and symbols denote vectors and matrices, while the plain letters and symbols denote scalars. We use $\mathbf{I}_n$ to denote an $n \times n$ identity matrix. "$\otimes$" denotes the operation to get the Kronecker product. "diag$(\cdot)$" denotes the diagonalization operation, where "$\cdot$" could be square matrices or scalars.

2.2 Graph Theory

During the tasks, robots interact with each other through some interaction topology that can be described by a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, a_{ij})$, where $\mathcal{V} = \{1, 2, \ldots, n\}$ is the set of nodes (robots), $\mathcal{E} = \{(i, j) \in \mathcal{V} \times \mathcal{V} : a_{ij} \neq 0\}$ is the set of edges (interactions), and a_{ij} are the weights of the edges between the i-th and j-th robots. A graph is called undirected if $a_{ij} = a_{ji}$; otherwise, it is a directed graph (Oh et al., 2015).

For a robot i in the multi-robot system, we denote its neighbors by $\mathcal{N}_i = \{j \in \mathcal{V} : a_{ij} \neq 0\}$. The degree of robot i is defined by $d_i = \sum_{j \in \mathcal{N}_i} a_{ij}$. The interconnection of robots is established by the Laplacian matrix $\mathbf{L} = [l_{ij}] \in \mathbb{R}^{n \times n}$, where

$$l_{ij} = \begin{cases} d_i, & \text{if } i = j \\ -a_{ij}, & \text{if } i \neq j \end{cases} \tag{1}$$

l_{ij} is the (i, j)-th entry of $\mathbf{L}$; $a_{ij} > 0$ if $j \in \mathcal{N}_i$ and $a_{ij} = 0$ otherwise (Mesbahi and Egerstedt, 2010). For an undirected connected graph, the Laplacian matrix $\mathbf{L}$ is symmetric and positive semi-definite (Oh et al., 2015).

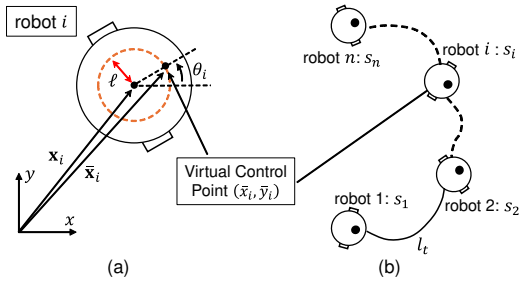


Figure 2. The configuration of the robot: (a) the original coordinates and the shifted coordinates; (b) the tube-connected robots.

2.3 Kinematics of Nonholonomic Robots

In this work, we consider a multi-robot system consisting of n robots that are interconnected by deformable tubes. The tubes are assumed to have equal length l_t, which allows the robots to maintain a certain distance from each other while performing cooperative tasks. The robots considered in this work are assumed to conduct planar motions and have the following nonholonomic dynamics:

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{\theta}_i \end{bmatrix} = \begin{bmatrix} \cos\theta_i & 0 \\ \sin\theta_i & 0 \\ 0 & 1 \end{bmatrix} \mathbf{u}_i \tag{2}$$

where $[\mathbf{x}_i^\mathsf{T}, \theta_i]^\mathsf{T}$ denotes the state of robot i; $\mathbf{x}_i = [x_i, y_i]^\mathsf{T}$ and θ_i are the position and the orientation of the robot, respectively; $\mathbf{u}_i = [v_i, \omega_i]^\mathsf{T}$ represents the control input, with v_i and ω_i as the linear and angular velocities of the robot, respectively.

A common practice for the control of nonholonomic robots is to use the input/output feedback linearization (Siciliano et al., 2009), where a virtual control point is introduced by a change of coordinates to simplify the controller design and stability analysis. The virtual control point is given by:

$$\begin{cases} \bar{x}_i := x + \ell \cos\theta_i \\ \bar{y}_i := y + \ell \sin\theta_i \\ \bar{\theta}_i := \theta_i \end{cases} \tag{3}$$

for an *arbitrary* scalar distance ℓ that translates the agent's position to an arbitrarily close location. The process of the change of coordinates is shown in Fig. 2(a). By using these new coordinates, we can construct a shifted state vector $\bar{\mathbf{x}}_i = [\bar{x}_i, \bar{y}_i]^\mathsf{T}$, whose time derivative yields a shifted dynamic model of the form:

$$\dot{\bar{\mathbf{x}}}_i = \bar{\mathbf{u}}_i, \quad \dot{\bar{\theta}}_i = \omega_i \tag{4}$$

where $\bar{\mathbf{u}}_i$ is the new control input for the position dynamics, which is defined by $\bar{\mathbf{u}}_i = \mathbf{R}_i(\theta_i)\mathbf{u}_i$. $\mathbf{R}_i(\theta_i)$ is a full-rank matrix defined as:

$$\mathbf{R}_i(\theta_i) = \begin{bmatrix} \cos\theta_i & -\ell\sin\theta_i \\ \sin\theta_i & \ell\cos\theta_i \end{bmatrix} \tag{5}$$

This enables us to represent the position dynamics of the multi-robot system as follows:

$$\dot{\bar{\mathbf{x}}} = \bar{\mathbf{u}} \tag{6}$$

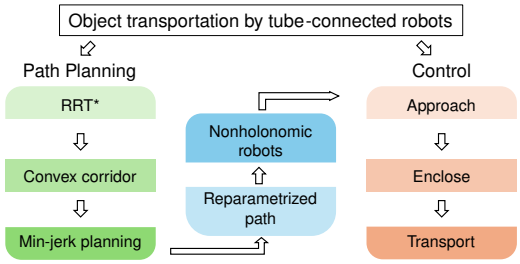


Figure 3. Overview of the proposed method.

for an extended position vector $\bar{\mathbf{x}} = [\bar{\mathbf{x}}_1^\mathsf{T}, \ldots, \bar{\mathbf{x}}_n^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{2n\times 1}$ and an extended input $\bar{\mathbf{u}} = [\bar{\mathbf{u}}_1^\mathsf{T}, \ldots, \bar{\mathbf{u}}_n^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{2n\times 1}$. The control input of the original dynamics can be simply computed as $\mathbf{u} = \mathbf{R}^{-1}\bar{\mathbf{u}}$, with a matrix $\mathbf{R} = \text{diag}(\mathbf{R}_1, \ldots, \mathbf{R}_n)$.

Besides, this work also adopts the following assumptions:

Assumption 1. *Robots and objects stay in a static planar environment with obstacles. Objects are star-shaped. We do not consider dynamic environments in this work.*

Assumption 2. (Quasi-static assumption) (Lynch and Mason, 1996) *During the transport, robots and objects move at such low speeds that inertial forces are negligible.*

2.4 Method Overview

Our goal in this work is to develop a complete hug and haul framework for deformable tube-connected robots to transport objects while avoiding obstacles. To achieve that, we divide the task into three main steps: approaching, enclosing, and transporting. For each step, we will first plan a path offline for the tube-connected robots to follow, and then design a controller to track the planned path. For the offline path planning, we will first use the RRT* algorithm to generate a discrete path for the tube-connected robots, and then generate convex travelling corridors along the discrete path to ensure the safety of the robots. Then, we will formulate a minimum jerk optimization problem to obtain a piecewise smooth path within the convex corridors. Finally, we reparametrize the piecewise path as a single smooth path for the tube-connected robots to follow. After obtaining the planned path, we will design a formation controller based on the parametric curve for the tube-connected robots to track the planned path and perform the cooperative task. The method overview is shown in Fig. 3.

3 Path Planning for Shape Morphing MRS

3.1 Map of the Environment

To plan paths for the tube-connected robots, we first need to construct an environment map that captures the obstacles and free space. We use an occupancy grid map to represent the environment, where each cell in the grid is marked as either occupied or free based on the presence of obstacles. To ensure the safety of the tube-connected robots, we further inflate the occupied cells in the occupancy map by a certain radius that accounts for the size of the robots and/or the objects, as well as a safety margin. This results in an inflated occupancy map that provides a more conservative representation of the environment, which can be used for path planning to avoid collisions with obstacles. An example of the map construction is shown in Fig. 4.

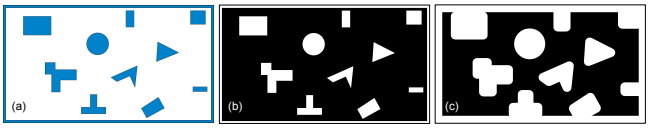


Figure 4. Example map for path planning: (a) the original map; (b) the occupancy map; (c) the inflated occupancy map.

3.2 RRT* Path Planning

With the inflated occupancy map, we can plan paths for the tube-connected robots to follow while avoiding obstacles. In this work, we first use the RRT* algorithm (Karaman and Frazzoli, 2011) to generate a discrete reference path for the tube-connected robots from the initial configuration to the goal configuration. The RRT* algorithm is a sampling-based path planning algorithm that incrementally builds a tree of feasible paths in the state space, and it is asymptotically optimal, meaning that it can find the optimal path as the number of samples approaches infinity. We follow the standard implementation of the RRT* algorithm, which includes sampling random states, extending the tree towards the sampled states, and rewiring the tree to improve the path quality. The discrete reference path generated by RRT* for an example map is shown in Fig. 5(a).

3.3 Convex Corridor Generation

Our next step is to generate convex travelling corridors along the discrete reference path obtained from RRT* for the generation of a smooth reference path for the tube-connected robots. For that, we search a square region for each path section between two adjacent waypoints on the reference path. First, we set the endpoints of the path section as the vertices of the square region. Second, we extend the square region by moving the vertices for a certain distance in the direction perpendicular to and along the path section. Then, we check the occupancy of the cells on the borders of the extended square region. If no occupied cells are detected, we keep the extended square region and repeat the second step; otherwise, we shrink the square region to the previous step and finish the process. Finally, we obtain a square travelling corridor for each path section, represented by a set of linear constraints:

$$a_i x + b_i y \leq c_i, \quad i = 1, \ldots, 4 \tag{7}$$

where a_i, b_i, and c_i are the coefficients of the linear constraints for the i-th border of the square corridor. The above constraints mean that for any point $[x, y]^\mathsf{T}$ satisfying (7), it lies in the corridor. The corridor generation process is shown in Fig. 5(b).

Note that a square is a simple shape for the corridor, which may lead to a small feasible region at the intersection points of the path sections, as shown in Fig. 5(c). To address this issue, we generate an intersection corridor at the intersection points of the path sections, which significantly enlarges the feasible region for smooth path generation, as shown in Fig. 5(d). The process of generating the intersection corridor is the same as that of generating square corridors for path

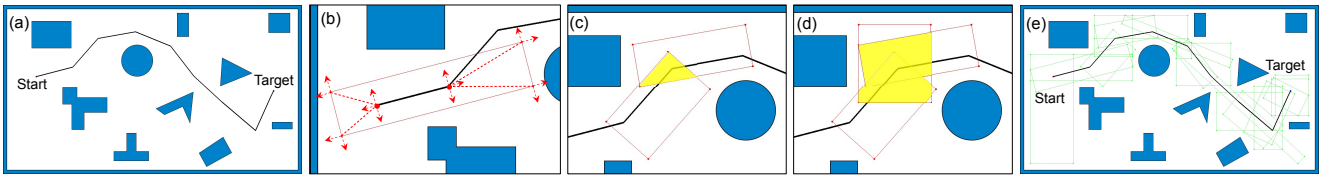


Figure 5. RRT* path planning and corridor generation: (a) the discrete reference path generated by RRT*; (b) corridor generation for path sections. Black lines: the reference path; red lines: the boundaries of the corridor; red dashed arrows: the extension direction of the corridor. (c) the feasible region at the intersection points without intersection corridor; (d) the feasible region at the intersection points with intersection corridor. (e) the generated convex travelling corridor (green squares) for the tube-connected robots.

sections. Finally, we can obtain a convex travelling corridor for the tube-connected robots to follow, which is shown in Fig. 5(e). We can see there are safe regions between the corridors and the obstacles due to the inflated occupancy map, which can ensure the safety of the tube-connected robots during the task.

3.4 Smooth Path Generation

After obtaining the convex travelling corridor, we can generate a smooth reference path for the tube-connected robots. In this work, we use a piecewise Bézier curve to represent the smooth reference path because Bézier curves have many desirable properties. One of the most important properties of Bézier curves is that they are contained within the convex hull of their control points, which allows easy obstacle avoidance by ensuring that the control points of the Bézier curve lie within the generated convex travelling corridor.

Suppose we have generated M sections of the convex travelling corridor, we can represent the m-th section of the smooth reference path as the following p-degree Bézier curve:

$$\mathbf{c}_m(t) = \mathbf{B}_m(t)\boldsymbol{\eta}_m, \quad t \in [0, T_m], m = 1, \dots, M \tag{8}$$

where $\boldsymbol{\eta}_m = [\boldsymbol{\eta}_{m,0}^\mathsf{T}, \cdots, \boldsymbol{\eta}_{m,p}^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{(2p+2)\times 1}$ is the vector of $p+1$ 2D control points for the m-th section of the Bézier curve. T_m is the duration of the m-th section of the Bézier curve, and we denote the whole duration of the curve as $T = \sum_{m=1}^{M} T_m$. $\mathbf{B}_m = \mathbf{b}_m \otimes \mathbf{I}_2 \in \mathbb{R}^{2\times(2p+2)}$ is the basis matrix for the m-th section of the Bézier curve, where $\mathbf{b}_m = [b_m^0, \cdots, b_m^p]$ is the vector of the following Bernstein polynomials:

$$b_m^j(t) = \binom{p}{j}\left(\frac{t}{T_m}\right)^j\left(1-\frac{t}{T_m}\right)^{p-j}, \quad j \in [0, p] \tag{9}$$

where $\binom{p}{j}$ is the binomial coefficient. t is normalized by T_m in the m-th section to map the time instant to $[0, 1]$.

To obtain a smooth curve and avoid sudden changes of paths to facilitate the tracking of the tube-connected robots, we minimize the third time derivative (jerk) of the Bézier curve, inspired by (Mellinger and Kumar, 2011). The cost function of the minimum-jerk optimization is formulated as:

$$J = \sum_{m=1}^{M}\int_0^{T_m}\left\|\frac{d^3\mathbf{c}_m(t)}{dt^3}\right\|^2 dt = \boldsymbol{\eta}^\mathsf{T}\mathbf{Q}\boldsymbol{\eta} \tag{10}$$

where $\boldsymbol{\eta} = [\boldsymbol{\eta}_1^\mathsf{T}, \cdots, \boldsymbol{\eta}_M^\mathsf{T}]^\mathsf{T}$ is the vector of all control points. $\mathbf{Q}$ is a semidefinite Hessian matrix defined as:

$$\mathbf{Q} = \text{diag}\left(\int_0^{T_m}\left(\frac{d^3\mathbf{B}_m(t)}{dt^3}\right)^\mathsf{T}\left(\frac{d^3\mathbf{B}_m(t)}{dt^3}\right)dt\right) \tag{11}$$

To formulate the optimization problem, we need to specify the constraints of the piecewise Bézier curve.

3.4.1 Initial and Goal Configuration Constraints

Throughout the task, the smooth path should start from the initial configuration and end at the goal configuration. We denote the initial and goal configurations as $\mathbf{c}_0$ and $\mathbf{c}_f$, respectively. Then, the initial and goal configuration constraints are formulated as:

$$\begin{cases}\mathbf{c}_0(0) = \mathbf{c}_0, \mathbf{c}_0'(0) = \mathbf{c}_0', \mathbf{c}_0''(0) = \mathbf{c}_0'' \\ \mathbf{c}_M(T_M) = \mathbf{c}_f, \mathbf{c}_M'(T_M) = \mathbf{c}_f', \mathbf{c}_M''(T_M) = \mathbf{c}_f''\end{cases} \tag{12}$$

3.4.2 Smoothness Constraints

To ensure the smoothness of the reference path, we need to constrain the continuity of the piecewise curve up to second order at the intersection points. The smoothness constraints are formulated as:

$$\begin{cases}\mathbf{c}_m(T_m) = \mathbf{c}_{m+1}(0) \\ \mathbf{c}_m'(T_m) = \mathbf{c}_{m+1}'(0) \\ \mathbf{c}_m''(T_m) = \mathbf{c}_{m+1}''(0)\end{cases} \tag{13}$$

for $m = 1, \dots, M-1$.

3.4.3 Obstacle Avoidance Constraints

To ensure the safety of the tube-connected robots, we need to impose constraints to avoid obstacles in the workspace throughout the task. Since we have generated the convex travelling corridors for each section of the reference path and encoded them as some linear constraints, we can directly impose them on the piecewise Bézier curve to avoid obstacles. Considering the convex hull property of Bézier curves, we can ensure that the curve lies within the corridor by ensuring that all control points of the Bézier curve lie within the corridor. Therefore, the obstacle avoidance constraints can be formulated as:

$$a_i\eta_{m,j}^x + b_i\eta_{m,j}^y \le c_i \tag{14}$$

for $i = 1, \dots, 4$, $j = 0, \dots, p$, and $m = 1, \dots, M$, where $\eta_{m,j}^x$ and $\eta_{m,j}^y$ are the x and y coordinates of the j-th control point of the m-th section of the Bézier curve, respectively.

3.4.4 Min-Jerk Optimization It can be seen that constraints (12) and (13) can be written as affine equality constraints of the form $\mathbf{B}_{eq}\boldsymbol{\eta} = \mathbf{b}_{eq}$. Constraint (14) can be written as linear inequality constraints of the form $\mathbf{B}_{ieq}\boldsymbol{\eta} \leq \mathbf{b}_{ieq}$. Therefore, the smooth reference path generation can be formulated as the following quadratic programming (QP) problem:

$$\begin{aligned} \min \quad & J = \boldsymbol{\eta}^T \mathbf{Q} \boldsymbol{\eta} \\ \text{s.t.} \quad & \mathbf{B}_{eq}\boldsymbol{\eta} = \mathbf{b}_{eq} \\ & \mathbf{B}_{ieq}\boldsymbol{\eta} \leq \mathbf{b}_{ieq} \end{aligned} \tag{15}$$

3.5 Curve Reparametrization

After solving the above optimization problem, we can obtain the optimal control points $\boldsymbol{\eta}^*$ for the piecewise Bézier curve, which gives us a smooth reference path. However, the parameterization of the curve may not be suitable for the tracking control of the tube-connected robots, as it may lead to non-uniform motion along the curve. This is because the parameter t in the Bézier curve does not necessarily correspond to the arc length of the curve, which can result in varying inter-robot arc lengths. This problem is crucial for the tube-connected robots, as they are interconnected by tubes that cannot stretch too much, and thus require a more uniform motion along the curve to maintain the integrity of the tubes and ensure the safety of the robots. Also, when there are too many path sections, maintaining the control points of all sections can be computationally expensive for the tracking control of the tube-connected robots. To address this issue, we can reparametrize the piecewise Bézier curve by using the arc length of the curve as the new parameter, which can ensure a more uniform motion along the curve and maintain a consistent inter-robot arc length for the tube-connected robots.

In this work, we reparametrize the piecewise Bézier curve as a p_r-degree Bézier curve. We represent the reparametrized curve in the following linear regression form (Zhang et al., 2023; 2024):

$$\mathbf{c}(s) = \mathbf{G}(s)\boldsymbol{\xi} \tag{16}$$

where $\mathbf{c}(s)$ is the coordinate of a random point on the reparametrized curve; $\boldsymbol{\xi} \in \mathbb{R}^{(2p_r+2)\times 1}$ is the vector containing the control points to be calculated for the reparametrized curve; $\mathbf{G}(s) \in \mathbb{R}^{2\times(2p_r+2)}$ is the matrix containing the Bernstein basis functions for the reparametrized curve, which is defined as $\mathbf{G}(s) = \mathbf{b}_r \otimes \mathbf{I}_2$, where $\mathbf{b} = [b_0, \cdots, b_{p_r}]$ is the vector of the following Bernstein polynomials:

$$b_j(s) = \binom{p_r}{j} s^j (1-s)^{p_r-j}, \quad j \in [0, p_r] \tag{17}$$

To compute the control points $\boldsymbol{\xi}$ for the reparametrized curve, we solve the following optimization problem:

$$\begin{aligned} \min \quad & \int_0^1 \left\| \mathbf{c}_r(s) - \mathbf{c}(s) \right\|^2 ds \\ \text{s.t.} \quad & \mathbf{c}(0) = \mathbf{c}_0, \mathbf{c}(1) = \mathbf{c}_f \end{aligned} \tag{18}$$

where $\mathbf{c}_r(s)$ is the coordinate of a random point on the original piecewise Bézier curve, which can be computed by first mapping the normalized length parameter s to the time parameter t for each section of the piecewise Bézier curve and then computing the coordinate of the point on the curve using the Bézier curve equation. The integral in the cost function usually doesn't have a closed-form solution. To solve the optimization problem, we can use numerical integration to approximate the integral. First, we can sample N points on the original piecewise Bézier curve by uniformly sampling t in $[0, T_m]$ for each path section and computing the corresponding coordinates $\mathbf{c}_r(t)$ for each sampled point. Then, we can compute the corresponding arc length parameter s for each sampled point by calculating the arc length of the curve up to that point and normalizing it by the total arc length of the curve. In this way, we can map the original parameter t to the new parameter s for each sampled point and get the sampled points $\mathbf{c}_r(s)$ parametrized by s. For each sampled point, we can compute the corresponding point on the reparametrized curve $\mathbf{c}(s)$ using the Bézier curve equation. Finally, we can approximate the integral in the cost function and write the optimization problem (18) as:

$$\begin{aligned} \min \quad & \sum_{i=1}^{N} \left\| \mathbf{c}_r(s_i) - \mathbf{c}(s_i) \right\|^2 \\ \text{s.t.} \quad & \mathbf{c}(0) = \mathbf{c}_0, \mathbf{c}(1) = \mathbf{c}_f \end{aligned} \tag{19}$$

The above optimization problem is a least-squares problem with linear equality constraints, which can be efficiently solved using standard optimization techniques.

Remark 1. *By applying this curve reparameterization, the resulting single Bézier curve may violate collision-free conditions as we do not impose the obstacle avoidance constraints in* (19)*. However, this problem can be eased by using a suitable safety margin in the convex corridor generation step and selecting an appropriate degree for the resulting single Bézier curve to reduce the approximation error.*

3.6 Hug and Haul Strategy

In this work, we consider three main stages for the tube-connected robots to hug and haul objects: approaching, enclosing, and transporting, where hugging is achieved by approaching and enclosing, and hauling is achieved by transporting. With the proposed path planning method, we can plan paths for the task, but the specific path planning process may differ for different stages. Now, we will give the details of the path planning process for each stage.

3.6.1 Approaching Path To plan the approaching path for the tube-connected robots, we first need to specify the start and target points for the approaching path. The start point of the approaching path is given by the location of the tube-connected robots in the environment. The target point of the approaching path is determined by the position and the shape of the object to be transported, which can be seen in Fig. 6(c).

At the beginning, we will sample a bunch of points on the edge of the object and represent them in polar coordinates with the origin at the center of the object. Then, for each sampled point (r_o, θ_o) in the polar coordinates, we add a certain offset r_s to the radius r_o to get a bunch of offset points $(r_o + r_s, \theta_o)$. r_s is determined by the size of the robots and the user-defined safety margin for enclosing the object. Next, we map the offset points back to Cartesian coordinates and use them to approximate a closed curve around the

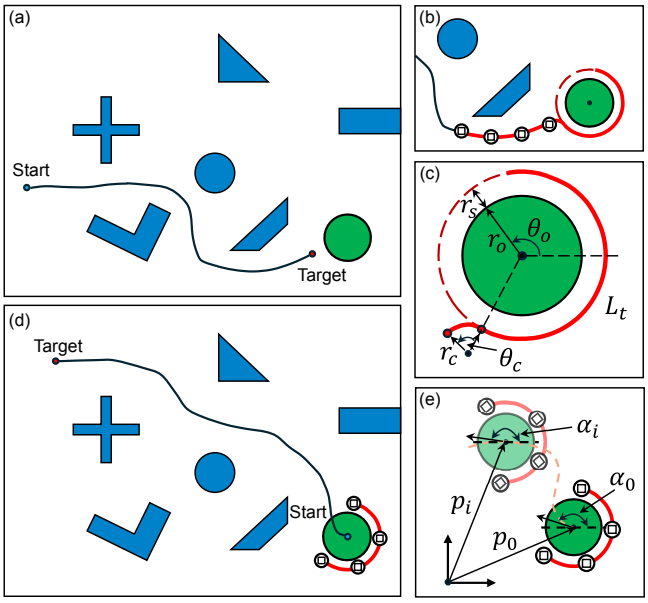


Figure 6. Path planning for different stages: (a) approaching the object; (b) enclosing the object; (c) obtaining the goal point of the approaching path and the section of enclosing the object; (d) transporting the object; (e) computing the shape of the tube-connected robots at each transport step.

object parameterized by the arc length. Then, we extract an enclosing section on the approximated closed curve with length equal to that of the tube-connected robots, given by $L_t = (n-1)l_t$. This enclosing section will be used in the enclosing step for the robots to enclose the object. Finally, we augment an intermediate section with a user-defined radius r_c to the beginning of the enclosing section and use its start point as the goal point of the approaching path. With this intermediate section, we can give more space for the path planning at the goal point. The center of the intermediate section is determined by adding an offset r_c to the radius of the start point of the enclosing section and its length is determined by a predefined angle θ_c. In this way, we can obtain the start and target points for the approaching path. An example can be seen in Fig. 6(a).

3.6.2 Enclosing Path The enclosing path consists of three parts. For the first part, we extract a section at the end of the approaching path with length equal to that of the tube-connected robots. The second and third parts are the intermediate section and the enclosing section obtained in the approaching path planning, respectively. We concatenate these three parts and conduct the curve reparametrization to get the final enclosing path. During the task, when the head robot of the tube-connected robots reaches the goal point of the approaching path, the robots will start to track the enclosing path to enclose the object. Fig. 6(b) is an example of the enclosing path.

3.6.3 Transporting Path For the transporting path, we don't directly plan a path for the tube-connected robots to follow; instead, we plan a path for the center of the object, as shown in Fig. 6(d). The initial and goal configurations for the transporting path are given by the initial position and the desired position of the object, respectively.

After planning the path for the center of the object, we can compute the desired shape of the tube-connected robots at each time instant during the transporting step based on the planned path and the enclosing section, as shown in Fig. 6(e). Specifically, we compute the tangent direction of the planned path at each time instant, then compute the desired shape of the tube-connected robots by aligning the enclosing section with the tangent direction and centering it at the path point. Suppose the enclosing section is represented as a p_o-degree Bézier curve $\mathbf{c}_o(t,s) = \mathbf{G}_o(t,s)\boldsymbol{\xi}_o$; $\mathbf{p}_0$ and $\mathbf{p}_i$ are the start point and a random point on the planned path for the center of the object, respectively; α_0 and α_i are the tangent directions of the planned path at $\mathbf{p}_0$ and $\mathbf{p}_i$, respectively. Then, we can compute the desired shape of the tube-connected robots at the time instant corresponding to $\mathbf{p}_i$ as follows:

$$\mathbf{c}_o(t_i,s) = \mathbf{R}_t(\alpha_i)\left[\mathbf{R}_t^{-1}(\alpha_0)(\mathbf{c}_o(t_0,s)-\mathbf{p}_0)\right] + \mathbf{p}_i \quad (20)$$

where $\mathbf{R}_t(\alpha)$ is the rotation matrix for angle α, given by:

$$\mathbf{R}_t(\alpha) = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix} \quad (21)$$

We can see that the desired path for the tube-connected robots is actually a time-varying Bézier curve. In this way, we can ensure that the tube-connected robots can maintain the enclosure of the object along the transport path.

Remark 2. *Object transport is achieved by controlling robots to maintain a formation enclosing the object and track the transporting path. The enclosing itself does not necessarily indicate the containment or caging of objects as the tube-connected robot system is an open chain.*

4 Curve-Based Controller

4.1 Control Objective

In the previous section, we have planned paths for the tube-connected robots to follow for each step of the cooperative task and represented them as Bézier curves that are parameterized by arc length. Specifically, the reference path for the approaching and enclosing steps is a fixed Bézier curve (computed by (18)), while the reference path for the transporting step is a time-varying Bézier curve (computed by (20)) for the robots. With a slight abuse of notation, we represent reference paths for all steps of the task as a time-dependent Bézier curve, given by $\mathbf{c}(t,s) = \mathbf{G}(t,s)\boldsymbol{\xi}(t)$. The control objective is to design a controller for the tube-connected robots to track the reference path while maintaining the formation of the robots and the safety of the robots and the object during the task.

To achieve this control objective, we equally distribute the tube-connected robots along the reference path by maintaining a constant inter-robot arc length, which is no larger than the tube length l_t. According to the desired inter-robot arc length, we can assign a length parameter s_i for each robot and obtain the corresponding Bernstein basis function matrix $\mathbf{G}_i(t,s_i)$ for each robot. Then, using the shifted position dynamics (6), we can define the robots' position error as:

$$\bar{\mathbf{x}}_e = \bar{\mathbf{x}} - \mathbf{G}(t,s)\boldsymbol{\xi}(t) \quad (22)$$

and the curve-coefficient error as:

$$\boldsymbol{\xi}_e = \mathbf{G}^{+}(t,s)\bar{\mathbf{x}} - \boldsymbol{\xi}(t) \quad (23)$$

where $\mathbf{G}(t,s) = [\mathbf{G}_1(t,s_1)^\mathsf{T}, \cdots, \mathbf{G}_n(t,s_n)^\mathsf{T}]^\mathsf{T}$ is the extended Bernstein basis matrix for all robots and $\mathbf{G}^+(t,s)$ is the pseudo-inverse of $\mathbf{G}(t,s)$. The position error and the curve-coefficient error can be used to design a controller for the tube-connected robots to track the reference path and maintain the formation of the robots. Also, the control objective can be achieved by minimizing the position error, i.e., the convergence of $\bar{\mathbf{x}}_e$ as $t \to \infty$.

4.2 Centralized Controller Design

We first introduce a centralized controller design for the tube-connected robots to track the reference path. The centralized controller has access to the global information of the system, including the positions and orientations of all robots and the curve coefficients. Assume that robots interact with each other through a connected undirected graph with the Laplacian matrix $\mathbf{L}_c$. The centralized formation controller for each robot is designed as follows:

$$\bar{\mathbf{u}}_i = -k_1 \sum_{j \in \mathcal{N}_i} (\mathbf{G}_i - \mathbf{G}_j)\, \boldsymbol{\xi}_e - k_2 \left(\bar{\mathbf{x}}_i - \mathbf{G}_i \boldsymbol{\xi}(t)\right) \quad (24)$$

where k_1 and k_2 are positive control gains. The first term in the controller is a formation-achieving term that drives the robots to the desired formation by leveraging the communication between the robots, while the second term is a tracking term that stabilizes the position error by directly using the position information of each robot and the reference path.

4.2.1 Approaching and Enclosing Control Since the reference paths for the approaching and enclosing steps are fixed Bézier curves, i.e., $\boldsymbol{\xi}(t)$ is constant despite the time variable, we can conduct the same stability analysis for the controllers of these two stages. In the approaching and enclosing control, in order to drive the robots to travel along the reference path while maintaining the formation of the tube-connected robots, we let the length parameter s_i for each robot increase at a constant speed v_s. Therefore, the length parameter s_i for each robot can be computed as

$$s_i(t) = \begin{cases} s_i(0) + v_s t, & t \le T_t \\ s_i(0) + v_s T_t, & t > T_t \end{cases} \quad (25)$$

where $s_i(0)$ is the initial length parameter for the i-th robot. T_t is the total duration of the approaching path or enclosing path. v_s is computed by $v_s = [1 - s_1(0)]/T_t$, ensuring that s_i is no larger than 1.

With a time-varying $s_i(t)$, the Bernstein basis functions $\mathbf{G}_i(s_i(t))$ are time-varying and we have the following proposition for the stability of the approaching and enclosing control.

Proposition 1. *For a multi-robot system with the shifted position dynamics in* (6) *and the centralized controller in* (24)*, if the reference path is a fixed Bézier curve and the length parameter s_i for each robot increases at a constant speed, then the position error $\bar{\mathbf{x}}_e$ is bounded as s_i changes and converges to zero when s_i stops changing; robots' orientations converge to constant values when s_i stops changing.*

Proof. A detailed proof can be found in Appendix A.

4.2.2 Transporting Control In the transporting step, the reference path for the tube-connected robots at each time instant is computed by (20). Considering the affine invariance of the Bézier curve, the transformation of the curve by rotation and translation can be achieved by transforming the control points of the curve. Therefore, the reference path for the tube-connected robots at each time instant can be represented as a Bézier curve with the same Bernstein basis function matrix $\mathbf{G}(t,s)$ but with time-varying curve coefficients $\boldsymbol{\xi}(t)$. In this case, the robots are required to track an evolving curve.

Besides, the robots need to carry the object to the desired position during the transporting step, which means that the robots need extra force to overcome the resistance of the object and the environment. As mentioned in the advantages of the proposed method in Section 1.2, we assume that the mass of the object and the friction coefficient between the object and the environment are *unknown* and that we can measure the global position of the object, denoted by $\mathbf{x}_{obj}$. We design an object compensation term for the controller to compensate for the unknown resistance of the object and the environment, given by:

$$\bar{\mathbf{u}}_{obj} = -\hat{k}(\bar{\mathbf{x}}_i - \mathbf{G}_i \boldsymbol{\xi}(t)), \quad 0 \le \hat{k} \le k_{max} \quad (26)$$

where k_{max} is the maximum value of $\hat{k}$; $\hat{k}$ is an adaptive control gain, updated by the following adaptive law:

$$\dot{\hat{k}} = \begin{cases} \gamma(e_{obj} - \varepsilon)^2 - \sigma \hat{k}, & e_{obj} > \varepsilon \\ -\sigma \hat{k}, & e_{obj} \le \varepsilon \end{cases} \quad (27)$$

where $\gamma > 0$ is the adaptation gain, $\sigma > 0$ is the forgetting factor, and ε is a predefined threshold for the position error of the object. $e_{obj} = \|\mathbf{x}_{obj} - \mathbf{x}^*_{obj}\|$ is the position error of the object, where $\mathbf{x}^*_{obj}$ is the desired position of the object that evolves along the transport path.

With the object compensation term, the centralized transporting controller for each robot is designed as follows:

$$\bar{\mathbf{u}}_i = -k_1 \sum_{j \in \mathcal{N}_i} (\mathbf{G}_i - \mathbf{G}_j)\, \boldsymbol{\xi}_e - k_2 \left(\bar{\mathbf{x}}_i - \mathbf{G}_i \boldsymbol{\xi}(t)\right) + \bar{\mathbf{u}}_{obj} \quad (28)$$

In this case, we have the following proposition for the stability of the transporting control.

Proposition 2. *For a multi-robot system with the shifted position dynamics in* (6) *and the centralized controller in* (28)*, if the reference path is a time-varying Bézier curve with time-varying curve coefficients, then the position error $\bar{\mathbf{x}}_e$ is bounded and converges to zero when curve coefficients stop changing; robots' orientations converge to constant values when curve coefficients stop changing.*

Proof. A detailed proof can be found in Appendix B.

4.3 Distributed Controller Design

For the centralized controller, robots are required to sense their absolute positions with respect to the global reference frame, which may not be feasible in some scenarios due to the limitations of sensing and communication capabilities of the robots. To address this issue, we design a distributed controller for the tube-connected robots to

achieve their formation, which only requires robots to sense relative positions of their neighbors with respect to the global reference frame (Oh et al., 2015). In this way, we can relax the requirement of sensing and communication capabilities for the tube-connected robots, which can expand the applicability of the control strategy to more practical scenarios.

Besides achieving the formation, the robots also need to reach the desired positions and track the reference path to successfully perform the cooperative task. To achieve this control objective, we assume that a subset of the robots, which are called leader robots, can sense their absolute positions with respect to the global reference frame and have access to the reference path, while the other robots, which are called follower robots, can only sense relative positions of their neighbors with respect to the global reference frame.

The robots interact through the same connected undirected graph as in the centralized controller. The distributed formation controller for each robot is designed as follows:

$$\begin{aligned}\bar{\mathbf{u}}_i = &-k_1 \sum_{j\in\mathcal{N}_i} (\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j - \mathbf{G}_i\boldsymbol{\xi}(t) + \mathbf{G}_j\boldsymbol{\xi}(t)) \\ &- \lambda_i k_2 (\bar{\mathbf{x}}_i - \mathbf{G}_i\boldsymbol{\xi}(t))\end{aligned} \tag{29}$$

where k_1 and k_2 are positive control gains. λ_i is an indicator for the role of the i-th robot, defined as:

$$\lambda_i = \begin{cases} 1, \text{if the } i\text{-th robot is a leader robot} \\ 0, \text{if the } i\text{-th robot is a follower robot} \end{cases} \tag{30}$$

The first term in the controller is a formation-achieving term that drives the robots to the desired formation by leveraging the communication between the robots, while the second term is a tracking term that drives the leader robots to track the reference path by directly using the position information of the leader robots and the reference path.

4.3.1 Approaching and Enclosing Control As in the centralized control, we let the length parameter s_i for each robot increase at a constant speed v_s in the approaching and enclosing control. In this case, we have the following proposition for the stability of the distributed approaching and enclosing control.

Proposition 3. *For a multi-robot system with the shifted position dynamics in* (6) *and the distributed controller in* (29)*, if the reference path is a fixed Bézier curve and the length parameter s_i for each robot increases at a constant speed, then the position error $\bar{\mathbf{x}}_e$ is bounded as s_i changes and converges to zero when s_i stops changing; robots' orientations converge to constant values when s_i stops changing.*

Proof. A detailed proof can be found in Appendix C.

4.3.2 Transporting Control As in the centralized control, the reference path for the tube-connected robots is a time-varying Bézier curve for the transporting control. We also add an object compensation term to the distributed controller to compensate for the unknown resistance of the object and the environment. Assuming that the leader robots can sense the global information of the object, the object compensation term is given by:

$$\bar{\mathbf{u}}_{obj} = -\lambda_i \hat{k}(\bar{\mathbf{x}}_i - \mathbf{G}_i\boldsymbol{\xi}(t)), \quad 0 \le \hat{k} \le k_{max} \tag{31}$$

where λ_i is defined in (30) and $\hat{k}$ is updated by the same adaptive law in (27).

With the object compensation term, the distributed transporting controller for each robot is designed as follows:

$$\begin{aligned}\bar{\mathbf{u}}_i = &-k_1 \sum_{j\in\mathcal{N}_i} (\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j - \mathbf{G}_i\boldsymbol{\xi}(t) + \mathbf{G}_j\boldsymbol{\xi}(t)) \\ &- \lambda_i k_2 (\bar{\mathbf{x}}_i - \mathbf{G}_i\boldsymbol{\xi}(t)) + \bar{\mathbf{u}}_{obj}\end{aligned} \tag{32}$$

In this case, we have the following proposition for the stability of the distributed transporting control.

Proposition 4. *For a multi-robot system with the shifted position dynamics in* (6) *and the distributed controller in* (32)*, if the reference path is a time-varying Bézier curve with time-varying curve coefficients, then the position error $\bar{\mathbf{x}}_e$ is bounded and converges to zero when curve coefficients stop changing; robots' orientations converge to constant values when curve coefficients stop changing.*

Proof. A detailed proof can be found in Appendix D.

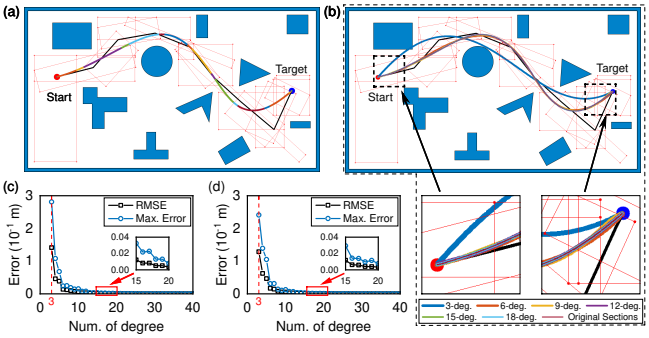


Figure 7. Validity of the path planning and curve reparametrization: (a) Original piecewise Bézier curve planned by the proposed method; (b) The reparametrized Bézier curve with different degrees; (c) The RMSE and maximum error of the reparametrized curve with different degrees; (d) The RMSE and maximum error of the arc length at different length parameters of the reparametrized curve with different degrees.

5 Results

5.1 Validity of the Path Planning and Reparametrization

We first use a simple case to validate the effectiveness of the proposed path planning and curve reparametrization. The environment for this case is the same as the one shown in Fig. 4(a) with a size of 4.32m × 2.43m. In this case, we consider finding a smooth path for a single robot to travel from the start point to the target point while avoiding the obstacles in the environment. The start point and target point are (0.48m, 1.03m) and (3.91m, 1.23m), respectively. We set the inflation radius as 9cm to account for the size of the robot and the safety margin, and then obtain the inflated occupancy map. Next, we implement the RRT* algorithm with a searching radius of 0.7m on the inflated occupancy map to get a discrete path. The resulting discrete path is

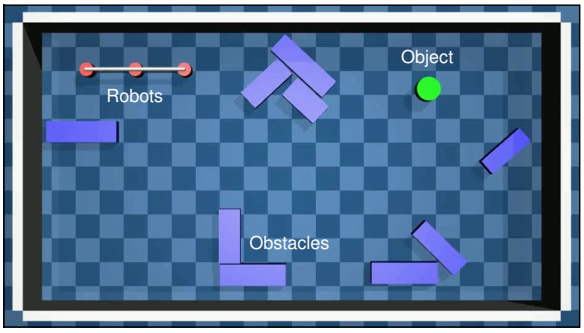


Figure 8. The simulation environment built with MuJoCo.

shown in Fig. 7(a), marked as the black line, which consists of 9 sections and 10 intersection points (including the start and target points). Then, the convex corridors are generated for each section and intersection point, shown in Fig. 7(a) as the red squares. We can see that the convex corridors successfully cover the discrete path and provide safe spaces (displaced from the obstacles) for the robot to travel through. Finally, we implement the min-jerk optimization in (15) to get a smooth piecewise Bézier curve. We set the degree of each section of the piecewise Bézier curve to 3, which means that there are 4 control points for each section. The QP problem is solved using the QP solver of the ALGLIB numerical library*, and the result is shown in Fig. 7(a). We can see that the planned path is smooth and each section of the path lies within the corresponding convex corridor, which means that the planned path can successfully avoid the obstacles in the environment.

After obtaining the piecewise Bézier curve, we implement the curve reparametrization method (19) to get a reparametrized Bézier curve. We test the reparametrization with degrees ranging from 3 to 40. The reparametrization problem is also solved using the QP solver of the ALGLIB numerical library. We select several reparametrized curves with different degrees and show the results in Fig. 7(b). We can see that as the degree of the reparametrized curve increases, the reparametrized curve can better approximate the original piecewise Bézier curve.

To quantitatively evaluate the approximation performance of the reparametrized curve, we compute the root mean square error (RMSE) and the maximum error between the original piecewise Bézier curve and the reparametrized curve for different degrees at the sampled points for the optimization by the following equations:

$$\begin{aligned} \text{RMSE} &= \sqrt{\frac{1}{N}\sum_{i=1}^{N} \|\mathbf{c}(s_i) - \mathbf{c}_r(s_i)\|^2} \\ \text{Max Error} &= \max_{i\in[1,N]} \|\mathbf{c}(s_i) - \mathbf{c}_r(s_i)\| \end{aligned} \tag{33}$$

The results are shown in Fig. 7(c). We can see that both the RMSE and the maximum error decrease as the degree of the reparametrized curve increases, and the RMSE is less than 2mm when the degree of the reparametrized curve is larger than 15. We also evaluate the approximation performance of the reparametrized curve in terms of the arc length, which is crucial for the tracking control of the tube-connected robots. We compute the RMSE and the maximum error of the arc

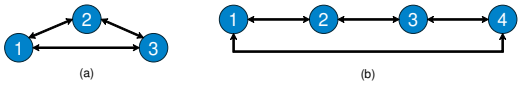


Figure 9. The communication graphs for the robots in the simulations and experiments: (a) graph for simulations; (b) graph for experiments.

length at different length parameters of the reparametrized curve with different degrees by the following equations:

$$\begin{aligned} \text{RMSE}_{len} &= \sqrt{\frac{1}{N}\sum_{i=1}^{N} \|L_r(s_i) - s_i L_r\|^2} \\ \text{Max Error}_{len} &= \max_{i\in[1,N]} \|L_r(s_i) - s_i L_r\| \end{aligned} \tag{34}$$

where $L_r(s_i)$ is the arc length of the reparametrized curve from the start point to the point corresponding to the length parameter s_i; L_r is the total arc length of the reparametrized curve. The results are shown in Fig. 7(d). We can see that both the RMSE and the maximum error of the arc length decrease as the degree increases, and the RMSE of the arc length is less than 2mm when the degree of the reparametrized curve is larger than 15.

These results validate the effectiveness of the proposed path planning and curve reparametrization methods in generating smooth reference paths for the tube-connected robots to follow during the cooperative tasks. Also, compared to the original piecewise Bézier curve, the reparametrized curve largely decreases the number of control points needed for representation with minimal loss in approximation accuracy, which can significantly reduce the computational and data management burden for the tracking control of the tube-connected robots.

5.2 Numerical Simulation

5.2.1 Setup We conduct extensive simulations to validate the effectiveness of the proposed path planning and control methods for the tube-connected robots to perform cooperative tasks. We build a simulation environment with MuJoCo† as shown in Fig. 8. The environment has an arena of size 2.6m $\times$ 1.4m surrounded by walls and contains several obstacles and the object to be transported. All simulations are conducted with three tube-connected robots, which are modeled as cylinders with a radius of 3.5cm and a wheelbase of 5.2cm. Tubes are simulated by small cylinders with a radius of 1cm and a length of 25cm. Throughout the simulations, robots interact with each other through a connected undirected graph shown in Fig. 9(a). For each simulation, we set a 20-second holding time after the enclosing path reaches the end to ensure that the robots can successfully enclose the object before transporting it. We also set a 60-second holding time after the transporting path reaches the end to ensure that the robots can successfully transport the object to the desired position. Parameters of the simulations are listed in Table 1.

*https://www.alglib.net/quadratic-programming/
†https://mujoco.org/

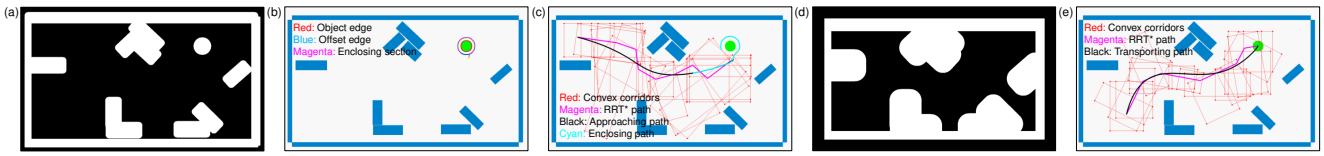


Figure 10. Path planning for the simulation of transporting a single object: (a) The inflated occupancy map for the approaching and enclosing steps; (b) The generation of the object-enclosing section; (c) The path planning results for approaching and enclosing the object; (d) The inflated occupancy map for the transporting step; (e) The path planning results for the transporting step.

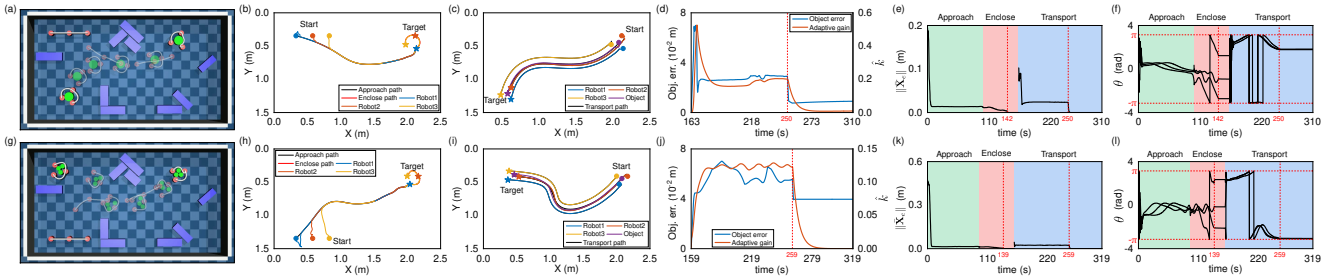


Figure 11. Simulation results for centralized control: (a), (g) Snapshots of the simulations; (b), (h) Robots' trajectories in the approaching and enclosing steps; (c), (i) Robots' and object's trajectories in the transporting step; (d), (j) Position errors of objects and adaptive gain $\hat{k}$; (e), (k) Position errors of the robots; (f), (l) Orientations of the robots. Obj. err.: object's position error.

Table 1. Parameters used in the simulations and experiments

Parameter		Sim.[1]	Exp.[2]
Deg. of piecewise Bézier curve p		3	3
Deg. of reparametrized curve p_r		15	15
Scalar for change of coordinates ℓ		-0.01	-0.01
Leader robot(s) in distributed control		2	1, 4
Control frequency (Hz)		10	10
Forgetting factor σ		0.18	0.2
Error threshold ε (m)		0.0035	0.01
Maximum adaptive gain k_{max}		1.4	1.0
Enclosing holding time (seconds)		20	20
Transport holding time (seconds)		60	30
Length of each tube l_t (cm)	Circle/Square	25	25
	Cloud	–	20
	Multiple objects	25	20
Approach & enclose control gains k_1, k_2	Centralized	1.6, 1.8	0.6, 0.4
	Distributed	1.6, 2.0	0.6, 0.6
Transport control gains k_1, k_2	Centralized	1.25, 1.35	0.6, 0.4
	Distributed	1.25, 1.80	0.6, 0.6
Adaptation gain γ	Centralized	55	5
	Distributed	77	10

[1] Sim. denotes the parameters used in the simulations.
[2] Exp. denotes the parameters used in the experiments.

5.2.2 Centralized Control: Single Object We first conduct a simulation to validate the effectiveness of the proposed path planning and centralized control methods for transporting a single object. A circular object with a radius of 6cm is placed in the environment, and the tube-connected robots are required to approach and enclose the object and then transport it to the desired position. The path planning results for the approaching, enclosing, and transporting steps are shown in Fig. 10. We can see that the planned paths for the tube-connected robots are smooth and can successfully avoid the obstacles.

The simulation results are shown in Fig. 11(a)-(f). From the snapshots and trajectories in Fig. 11(a)-(c), we can see that the tube-connected robots can successfully approach and enclose the object and then transport it to the desired position while avoiding the obstacles in the environment. The position errors of the object and the robots are shown in Fig. 11(d)-(e). We can see that the position errors of robots are bounded when tracking the reference paths. At the beginning of the transporting step, we can see robots' position errors jump up. This is because the ending configuration of the enclosing step is not exactly the same as the starting configuration of the transporting step, which leads to a sudden change in the reference path for the robots. This mismatch comes from the different safety margins of the enclosing section during the enclosing stage and the transport stage. However, this difference is quite small (within 1 cm) and doesn't affect the tasks. When entering the holding time, robots are actually forming a static formation, and then the position errors of robots rapidly converge to zero. Since the robots are connected by tubes and enclose the object, the object's position is confined by the robots. Therefore, the good performance of the robots' position tracking can ensure that the object can be transported to the desired position with a small position error. This can be seen in Fig. 11(d), where the position error of the object is bounded. The ending position error of the circular object is 0.91cm. Orientations of the robots are shown in Fig. 11(f). We can see that robots' orientations converge to some constant values when entering the holding time. All these results follow the theoretical analysis in Section 4.2, which validates the effectiveness of the proposed path planning and centralized control methods for transporting a single object.

5.2.3 Centralized Control: Multiple Objects One of the advantages of the proposed methods over existing approaches is that they are capable of transporting multiple objects. Therefore, we conduct a simulation to validate the effectiveness of the proposed methods for transporting multiple objects. In this simulation, we consider three circular objects with a radius of 3cm and place them as a cluster in the environment.

The simulation results are shown in Fig. 11(g)-(l). From the snapshots and trajectories in Fig. 11(g)-(i), we can see that the tube-connected robots can successfully approach and enclose the cluster of objects and then transport them to the desired position while avoiding the obstacles. Robots' position errors and orientations are shown in Fig. 11(k)-(l). We can see results similar to those in the single-object case: position errors are bounded and then converge to zero during the holding time and orientations converge to constant values during the holding time. Thanks to the tubes, robots can maintain the enclosing of objects even when they are smaller than the gap between robots. We measure the position errors of the objects by using the geometric center (computed by averaging the positions of all objects) of the object cluster, shown in Fig. 11(j). We can see that the position error of the objects is bounded and the ending position error of the cluster of objects is 3.96cm. All the results follow the theoretical analysis in Section 4.2, which validates the effectiveness of the proposed path planning and centralized controller for transporting multiple objects.

5.2.4 Distributed Control In this section, we conduct simulations to validate the effectiveness of the proposed distributed control methods. Similar to the centralized control, we conduct two simulations for transporting a single object and multiple objects, respectively. For these two simulations, we adopt the same initial conditions and the path planning results as the centralized control simulations.

The simulation results are shown in Fig. 12. From the snapshots and trajectories in Fig. 12(a)-(c) and (g)-(i), we can see that the tube-connected robots can successfully approach and enclose the object(s) and then transport it/them to the desired position while avoiding the obstacles in the environment. The position errors and orientations of the robots also show results similar to those in the centralized control simulations, which are shown in Fig. 12(d)-(f) and (j)-(l). We can see that the position errors of robots are bounded and converge to zero during the holding time and orientations converge to constant values during the holding time. The difference is that the position errors of robots in the distributed control simulations are larger than those in the centralized control simulations. This is because only leader robots have access to the global information under the distributed control framework, which leads to a slower convergence of the position errors of robots. The position errors of the object(s) are also bounded. The ending position error of the circular object is 1.28cm and the ending position error of the multiple objects is 5.75cm. All the results follow the theoretical analysis in Section 4.3, which validates the effectiveness of the distributed controllers for transporting a single object and multiple objects.

5.2.5 Comparison and Ablation Study In this section, we perform a comparison study to further validate the effectiveness of our proposed methods. We consider three existing approaches, which are Bertoncelli's method (Bertoncelli et al., 2022), Ebel's method (Ebel et al., 2024), and Zhi's method (Zhi et al., 2024), for comparison. We conduct both qualitative and quantitative comparisons between our proposed methods and the existing approaches. For the quantitative comparison, we conduct simulations with two types of objects: circular and square. For the circular object, we use the same initial conditions and path planning results as in Section 5.2.2. For the square object, we consider a square object with a side length of 11cm. The simulation results and the comparison are shown in Fig. 13 and Table 2.

In terms of qualitative comparison, the approaches in (Bertoncelli et al., 2022; Ebel et al., 2024) only consider the transporting of a single object without path planning, which means that they cannot handle the approaching and enclosing of the object, the transporting of multiple objects, and the obstacle avoidance during the transporting step. The tube-connected approach considers the enveloping of objects, but does not include global path planning or multiple-object transport, which limits its applications (Zhi et al., 2024). Our proposed method offers a complete solution for the cooperative transportation of objects by tube-connected robots from the path planning to the control, which can handle the approaching and enclosing of the object, the transporting of multiple objects, and the obstacle avoidance during the whole process.

For the quantitative comparison, we first report the results of (Bertoncelli et al., 2022) and (Ebel et al., 2024). Since they do not consider the approaching and enclosing of the object, we only compare the transporting step of these two existing approaches with our proposed methods. We also don't take the holding time into account for the comparison since (Bertoncelli et al., 2022) and (Ebel et al., 2024) do not consider this phase. We record the position errors of the circular and square objects, which can be seen in Fig. 13(j) and (l). It can be seen that our proposed methods can achieve smaller position errors for both circular and square objects compared to the two implemented approaches. As shown in Fig. 13(b), at the ending stage of Ebel's method with a circular object, robots even lost contact with the object. We compute the root mean square error (RMSE), maximum error, and ending error of the position errors of the objects, which are shown in Table 2. Both the centralized and distributed controllers give smaller reported errors in most entries, except for the maximum error of the circular object. From Fig. 13(j), we can see that the maximum error of the circular object for our proposed methods appears at the beginning of the transporting step. This is the stage when the robots are adjusting their positions to the transporting path after the enclosing step, which is a transient stage and can be improved by better tuning the ending conditions of the enclosing step.

For the comparison with (Zhi et al., 2024), we conduct the full-stage simulations, i.e., including approaching, enclosing, and transporting, since both our method and Zhi's method consider all stages. For the simulation, we keep the two-robot configuration of (Zhi et al., 2024) and set the tube length to twice that of the three-robot configuration so that the total length of tube-connected robots is the same throughout all simulations. Robots depart from the same initial positions and objects have the same initial and target positions as in the previous simulations. The simulation results are shown in Fig. 13(c), (f), and (i). It can be seen in Fig. 13(c) that robots are trapped by the obstacles when approaching the circular objects, which leads to the failure of the task. When dealing with a square object, Zhi's method can successfully complete the task. Since robots are not tracking planned paths for

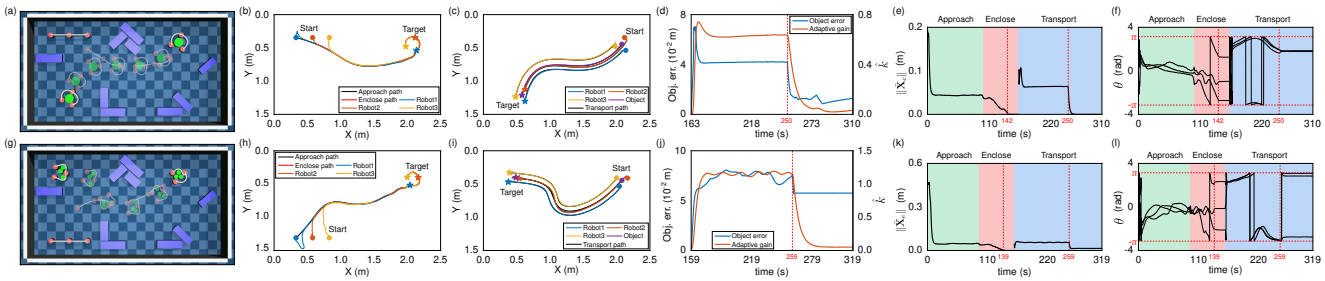


Figure 12. Simulation results for distributed control: (a), (g) Snapshots of simulations; (b), (h) Robots' trajectories in approaching and enclosing steps; (c), (i) Robots' and object's trajectories in the transporting step; (d), (j) Position errors of objects and adaptive gain $\hat{k}$; (e), (k) Robots' position errors; (f), (l) Robots' orientations.

Table 2. Results of comparison and ablation study

Comparison items		Our methods				(Bertoncelli et al., 2022)	(Ebel et al., 2024)	(Zhi et al., 2024)
		Centralized	Cent. w/o. ad.[1]	Distributed	Dist. w/o. ad.[2]			
Qualitative	Global path planning	√	√	√	√	×	×	×
	Approaching and enclosing	√	√	√	√	×	×	√
	Nonconvex object	√	√	√	√	×	√	√
	Multiple objects	√	√	√	√	×	×	×
	Obstacle avoidance	√	√	√	√	×	×	√
Quantitative	Circular RMSE	**2.95cm**	3.11cm	4.26cm	5.11cm	5.55cm	5.38cm	–
	Circular max error	7.00cm	6.76cm	7.05cm	7.05cm	**5.99cm**	19.1cm	–
	Circular ending error	**2.93cm**	3.12cm	4.22cm	7.94cm	5.58cm	19.1cm	failed
	Circular robot RMSE	**3.02cm**	3.27cm	6.62cm	8.05cm	–	–	–
	Circular robot max error	**10.3cm**	10.3cm	10.7cm	10.7cm	–	–	–
	Circular robot ending error	**2.36cm**	2.71cm	6.38cm	7.94cm	–	–	–
	Square RMSE	1.56cm	–	**0.980cm**	–	6.66cm	1.62cm	–
	Square max error	**2.43cm**	–	3.06cm	–	7.74cm	2.60cm	–
	Square ending error	1.93cm	–	**1.10cm**	–	7.56cm	2.25cm	4.12cm

[1] Cent. w/o. ad. denotes centralized control without adaptive gain.
[2] Dist. w/o. ad. denotes distributed control without adaptive gain.

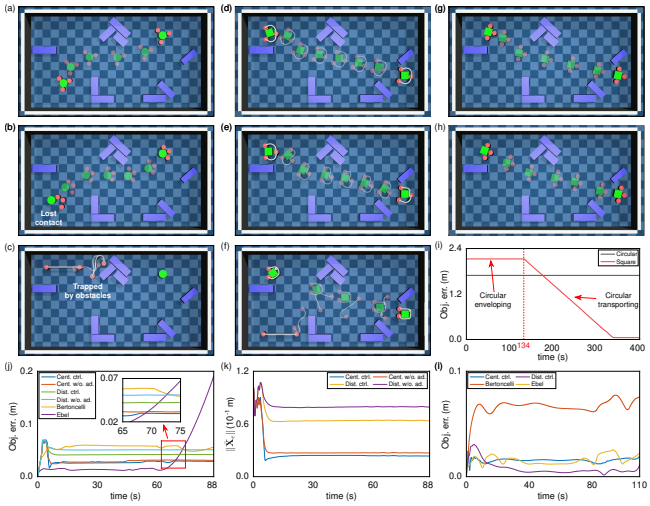


Figure 13. Comparison and ablation study of the simulation results: (a), (b) Bertoncelli's/Ebel's method with a circular object; (c) Zhi's method with a circular object; (d), (e) The proposed centralized/distributed control with a square object; (f) Zhi's method with a square object; (g), (h) Bertoncelli's/Ebel's method with a square object; (i) Object's error of Zhi's method; (j) Position errors of circular objects; (k) Robots' position errors with different proposed methods; (l) Position errors of square objects. Cent. ctrl.: centralized control; w/o. ad.: without adaptive gain; Dist.: distributed.

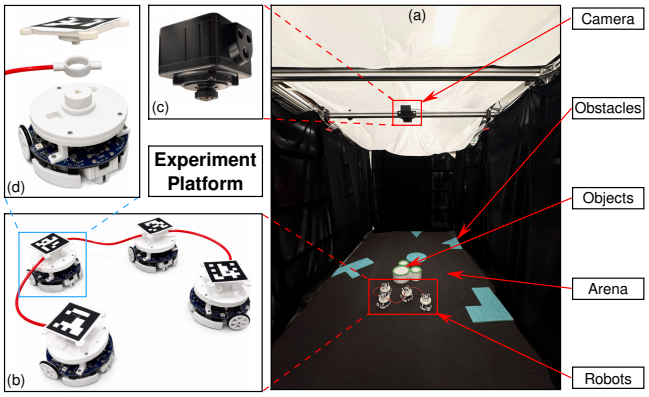


Figure 14. The platform for the experiments: (a) Overview of the platform; (b) The tube-connected robots; (c) The top-down camera; (d) The customized Mona robot.

Zhi's method, we only measure the ending position error of objects, and Table 2 shows that Zhi's method ends up with a higher number.

To validate the effectiveness of the object compensation term in the proposed methods, we also conduct an ablation study with the circular object by removing the object compensation term from both the centralized and distributed control methods. The position errors of the objects and robots are shown in Fig. 13(c) and (f). We can see that the position errors of the objects and robots are larger than those with the object compensation term, which indicates that the object compensation term can improve the performance of the proposed methods. This can also be seen in Table 2, where the RMSE, maximum error, and ending error of the position errors of the objects and robots are larger than those with the object compensation term.

The comparison and ablation study results validate the effectiveness of our proposed methods in achieving better performance than the existing approaches and the

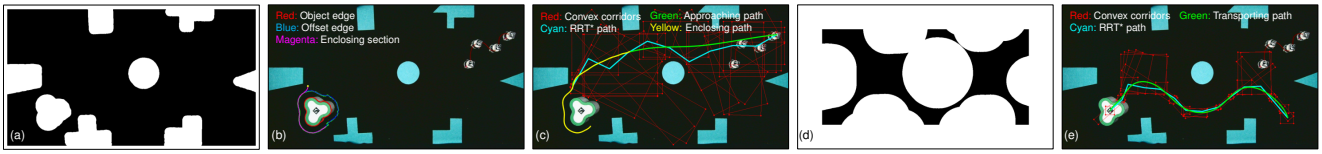


Figure 15. Path planning for transporting a single object: (a) The inflated occupancy map for approaching and enclosing steps; (b) The object-enclosing section; (c) The path planning results for approaching and enclosing steps; (d) The inflated occupancy map for the transporting step; (e) The path planning results for the transporting step.

effectiveness of the object compensation term in improving the performance of our proposed methods.

5.3 Hardware Experiments

5.3.1 Setup In this work, we conduct extensive experiments to validate the effectiveness of the proposed methods for tube-connected robots in performing cooperative tasks. We consider a 2D environment with obstacles and one or more objects to be transported, where the objects are made of foam boards. The platform for the experiments is shown in Fig. 14(a), which includes an arena with a size of 2.78m $\times$ 1.40m to conduct the experiments. A top-down camera (see Fig. 14(c)) is used to track the positions and orientations of the robots and detect the obstacles and the object(s) in the environment. A control PC is used to process the camera data, plan paths for the tube-connected robots, and compute the control inputs for the robots. Communication between the robots and the PC is achieved through Wi-Fi and runs at a frequency of 10Hz. The Wi-Fi communication is configured based on the Robot Operating System (ROS).

The tube-connected robots used in the experiments are customized Mona robots (Arvin et al., 2019) shown in Fig. 14(b). The Mona robot has a diameter of 8cm and nonholonomic dynamics. For each robot, we design a customized 3D-printed structure shown in Fig. 14(d), which consists of a ring structure to connect tubes and a cap to carry the ArUco Markers. Tubes are made of silicone and have a customized length according to the size of the object to be transported. Throughout the experiments, we employ four Mona robots connected by three tubes to perform the cooperative tasks, which means that the length of the tube-connected robots is approximately equal to three times the tube length. Robots are placed in the environment with random initial orientations and proper random initial positions. Robots interact with each other through a connected undirected graph, which is shown in Fig. 9(b). The parameters used in the experiments are listed in Table 1.

We conduct extensive experiments to validate the effectiveness of the proposed methods for different scenarios, including transporting a single object and multiple objects in two different environments. For conciseness, we only present part of the results in this section, and the complete results can be found in the supplementary video.

Remark 3. *Due to the limited onboard capability of the robots, we actually conduct the distributed control experiments with centralized perception and computation.*

5.3.2 Centralized Control: Single Object We first validate the effectiveness of the proposed path planning and centralized controller for tube-connected robots in transporting a single object. A cloud-shaped object is first tested in an environment with several convex and non-convex obstacles. The tube-connected robots are required to approach and enclose the object and then transport it to the desired position. The path planning results for the approaching, enclosing, and transporting steps are shown in Fig. 15. We can see that the planned paths for the tube-connected robots are smooth and can successfully avoid the obstacles in the environment.

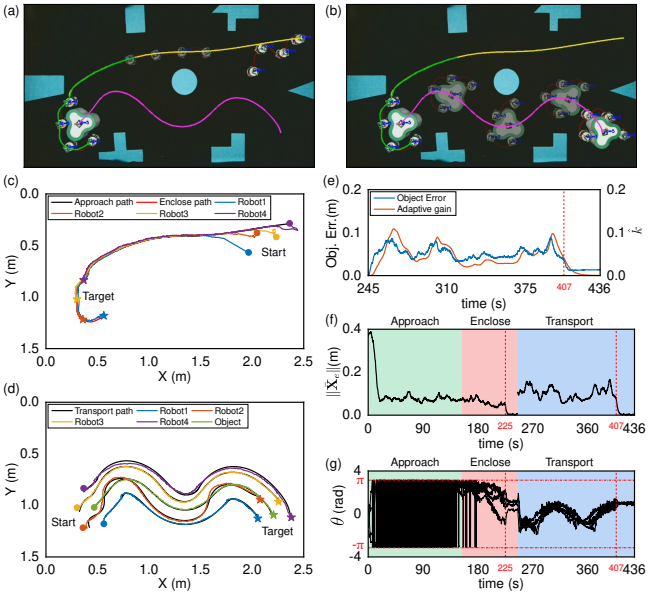


Figure 16. Experiments with a cloud-shaped object: (a), (b) Snapshots of the experiment; (c), (d) Robots' trajectories in the experiment; (e) Position error of the object and the adaptive gain $\hat{k}$; (f), (g) Position errors and orientations of the robots during the experiments.

The experimental results are shown in Fig. 16. From the snapshots and trajectories in Fig. 16(a)-(d), we can see that the tube-connected robots can successfully approach and enclose the cloud-shaped object and then transport it to the desired position while avoiding the obstacles in the environment. The position errors of the object and the robots are shown in Fig. 16(e) and Fig. 16(f), respectively. We can see that the position error of the robots is bounded when tracking the planned paths and rapidly converges to zero when the desired positions stop changing. Similar to the simulation results, at the beginning of the transporting step, we can see robots' position errors jump up due to the sudden change in the reference path for the robots. Confined by the tube-connected robots, the position error of the cloud-shaped object is also bounded and ends with a value of 1.34cm. The orientations of the robots during the experiments are shown in Fig. 16(f). We can see that the orientations of the robots reach some constant values when entering the holding time.

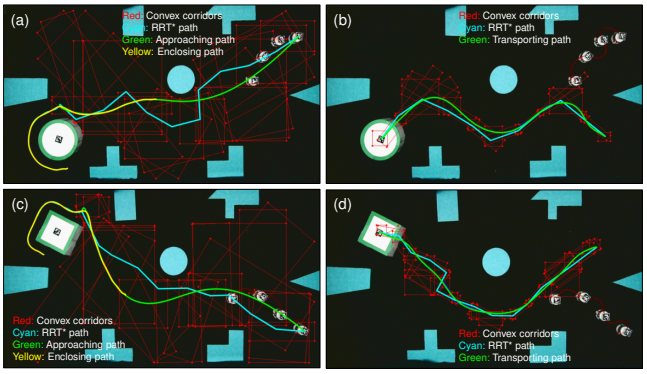


Figure 17. The path planning results for square and circular objects: (a), (b) The path planning results for the circular object; (c), (d) The path planning results for the square object.

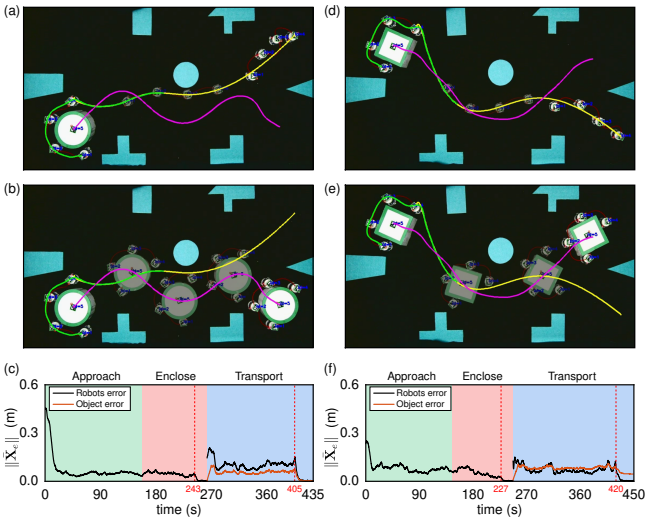


Figure 18. Experiments of square and circular objects: (a), (b) Snapshots of the experiments with the circular object; (c) Position errors of robots and the circular object; (d), (e) Snapshots of the transporting step for the square-shaped object; (f) Position errors of robots and the square object.

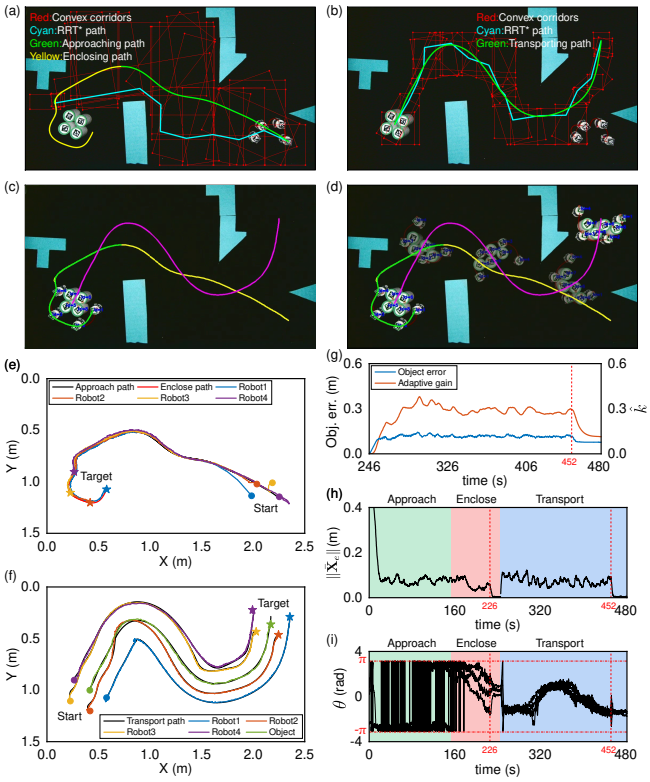


Figure 19. Experiment of multiple objects: (a), (b) The path planning results for multiple objects; (c), (d) Snapshots of the experiment with multiple objects; (e), (f) Robots' and objects' trajectories during the experiment; (g) The position error of objects and the adaptive gain $\hat{k}$; (h), (i) Position errors and orientations of the robots during the experiments.

To validate the effectiveness of the proposed methods for objects with different shapes, we also conduct experiments for a circular object and a square object. The results of the path planning and centralized control for the circular object and the square object are shown in Fig. 17 and Fig. 18, respectively. We can observe similar performance of the tube-connected robots in approaching, enclosing, and transporting the square object and the circular object to that for the cloud-shaped object. Robots' position errors are bounded when tracking the planned paths and rapidly converge to zero when the desired positions stop changing. Objects' position errors are also bounded and end with 0.37cm and 4.49cm for the circular object and the square object, respectively.

These results follow the expected behaviors in the stability analysis of the proposed centralized controller, which validates the effectiveness of the proposed path planning and centralized control methods for tube-connected robots in performing cooperative tasks with a single object to be transported.

5.3.3 Centralized Control: Multiple Objects In this section, we validate the effectiveness of the proposed path planning method and centralized controller with multiple objects. The experiments for the multiple objects can be seen in Fig. 19. We consider four small circular objects to be transported in an environment with several convex and non-convex obstacles. In the path planning, we treat the four objects as a whole and generate a single enclosing section for the four objects. The path planning results for the approaching and enclosing steps and the transporting step are shown in Fig. 19(a) and Fig. 19(b), respectively. We can see that the planned paths for the approaching and enclosing steps and the transporting step are smooth and can successfully avoid the obstacles in the environment. This validates the effectiveness of the proposed path planning method in dealing with multiple obstacles.

From the snapshots and trajectories in Fig. 19(c)-(f), we can see that the tube-connected robots can successfully approach and enclose the four circular objects and then transport them to the desired position while avoiding the obstacles. The position errors and orientations of the robots during the experiments are shown in Fig. 19(h) and Fig. 19(i), respectively. We can observe performance similar to that in the single-object case: the position errors of the robots are bounded and then converge to zero during the holding time and orientations converge to constant values during the holding time. Thanks to the existence of tubes, the robots can maintain the enclosing of the small objects during the transporting step. As in the simulations, we measure the position errors of the objects by using the geometric center of the four circular objects, which is shown in Fig. 19(g). We

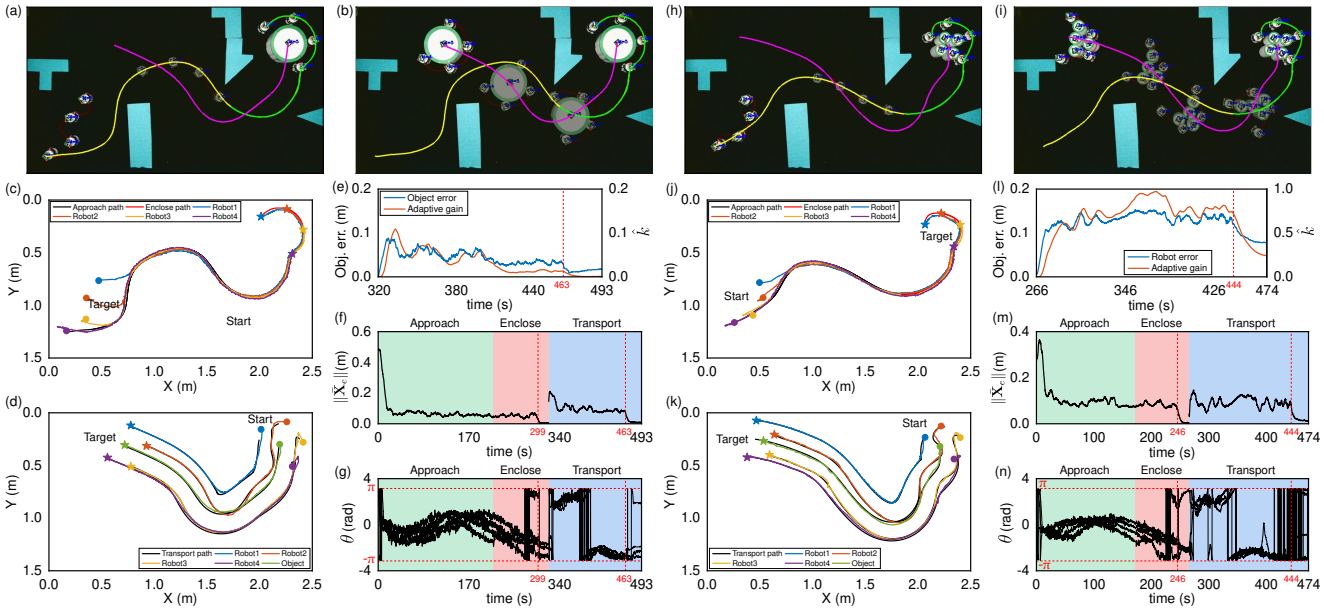


Figure 20. Experiments of distributed control: (a), (b) Snapshots of the experiment with a circular object; (c), (d) Robots' and object's trajectories; (e) Position error of the circular object and the adaptive gain $\hat{k}$; (f), (g) Robots' position errors and orientations; (h), (i) Snapshots of the experiment with multiple objects; (j), (k) Robots' and objects' trajectories; (l) Position error of the multiple objects and the adaptive gain $\hat{k}$; (m), (n) Robots' position errors and orientations.

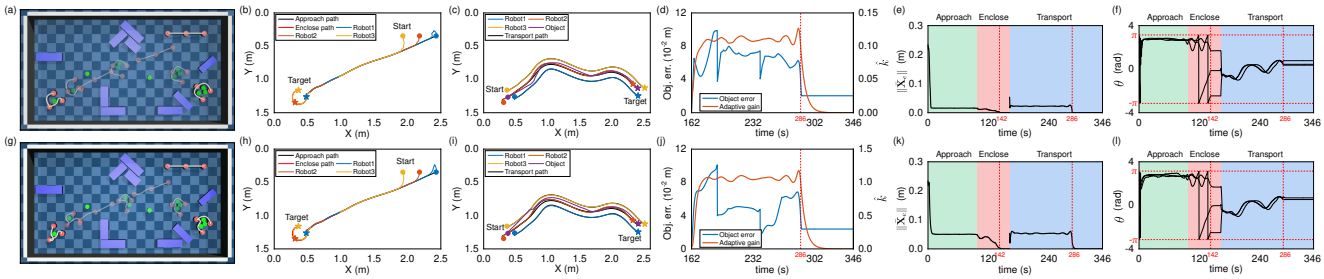


Figure 21. Simulations of collecting sparse objects: (a), (g) Snapshots of simulations with centralized/distributed control; (b), (h) Robots' trajectories during approaching and enclosing steps; (c), (i) Robots' and objects' trajectories during the transporting step; (d), (j) Objects' position errors and the adaptive gain $\hat{k}$; (e), (k) Position errors of the robots; (f), (l) Orientations of the robots.

can see that the position error of the objects is bounded and ends with a value of 7.72cm. These results also follow the expected behaviors in the stability analysis of the proposed centralized controller, which validates the effectiveness of it with multiple objects to be transported.

5.3.4 Distributed Control In this section, we validate the effectiveness of the proposed distributed controller for tube-connected robots in transporting objects. Since we have already validated the effectiveness of the proposed path planning method, we only present the results of the distributed control experiments. As in the centralized control experiments, we consider a single object and multiple objects to be transported in an environment with several convex and non-convex obstacles. The results of the distributed control experiments with a circular object and multiple objects are shown in Fig. 20.

From the snapshots and trajectories in Fig. 20(a)-(d) and Fig. 20(h)-(k), we can see that the tube-connected robots can successfully approach and enclose the object(s) and then transport it/them to the desired position while avoiding the obstacles in the environment. Objects' position errors and robots' position errors and orientations during the experiments all have performance similar to that in the centralized control cases. The single circular object ends with a position error of 1.79cm and the multiple objects end with a position error of 7.83cm. These results follow the expected behaviors in the stability analysis of the proposed distributed controller, which validates the effectiveness of it with both single and multiple objects.

Remark 4. *Physical limits on linear and angular velocities may clip the commands obtained from* $\mathbf{u}_i = \mathbf{R}_i^{-1}\bar{\mathbf{u}}_i$*, with small* $|\ell|$ *increasing the angular-velocity demand. Active saturation alters the nominal shifted dynamics and can slow convergence or increase tracking errors. The stability bounds in Propositions 1–4 apply when the commanded velocities remain within the actuator limits and do not directly extend to active saturation. Reference speeds and feedback gains should therefore be selected jointly with the actuator limits.*

5.4 Transport of Multiple Sparse Objects

In previous sections on multiple objects, we place the objects as a cluster in the environment. However, in some scenarios, the objects may be sparsely distributed in the environment and the tube-connected robots need to collect them and

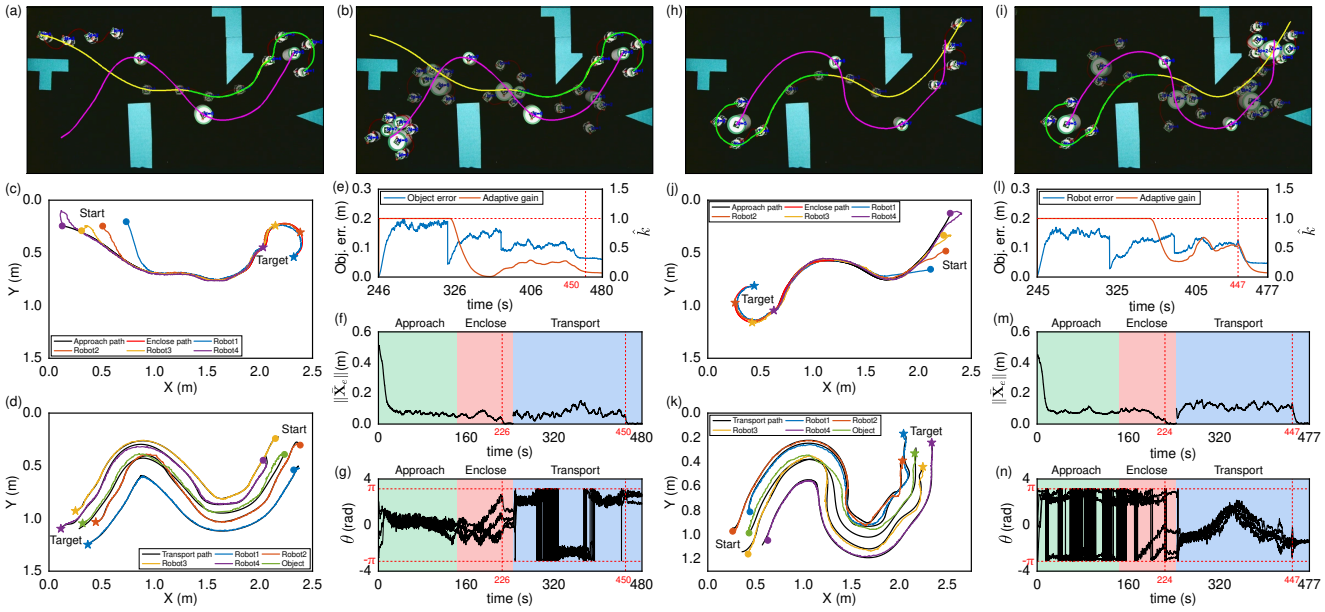


Figure 22. Experiments of collecting sparse objects: (a), (b) Snapshots of centralized control; (c), (d) Trajectories of the robots with centralized control; (e) Objects' position error with centralized control and the adaptive gain $\hat{k}$; (f), (g) Position errors and orientations of the robots with centralized control; (h), (i) Snapshots of distributed control; (j), (k) Trajectories of the robots with distributed control; (l) Objects' position error with distributed control and the adaptive gain k; (m), (n) Position errors and orientations of the robots with distributed control.

transport them to the desired position. Our proposed methods can be conveniently extended to handle this scenario.

In this scenario, we generate the enclosing section as a fixed shape, which is a part of a circle with a proper radius, to enclose the sparse objects. When planning the transporting path, we take the geometric center of each object as a waypoint and generate a path that passes through all waypoints. We conduct simulations and experiments to validate the effectiveness of the proposed methods in collecting sparse objects. The simulation results with three sparse circular objects are shown in Fig. 21. From the snapshots and trajectories in Fig. 21(a)-(c) and Fig. 21(g)-(i), we can see that the tube-connected robots can successfully collect all sparse objects and transport them to the desired position while avoiding the obstacles under both centralized and distributed control. The position errors and orientations of the robots and the position errors of the objects are shown in Fig. 21(e)-(f) and Fig. 21(k)-(l). We can observe performance similar to that in the previous simulations: robots' position errors are bounded and then converge to zero during the holding time; robots' orientations converge to constant values during the holding time. Position errors of the objects are shown in Fig. 21(d) and Fig. 21(j). We measure the position errors of the objects by the geometric center of objects that are already collected. Therefore, we can observe two sudden changes in objects' position errors, which are caused by the change of the geometric center of the collected objects. The ending position errors of the objects are 2.04cm and 2.39cm for the centralized and distributed control, respectively.

The experimental results with three sparse circular objects of different sizes are shown in Fig. 22. From the snapshots in Fig. 22(a)-(b) and Fig. 22(h)-(i), we can see that the planned transporting paths successfully pass through all the waypoints. The trajectories of robots, position errors of objects, position errors and orientations of the robots are shown in Fig. 22(c)-(g) and Fig. 22(j)-(n), respectively. We can observe performance similar to that in the simulations: robots' position errors are bounded and then converge to zero during the hold time; robots' orientations converge to constant values during the hold time. Objects' position errors end with 6.10cm and 4.67cm for the centralized and distributed control, respectively.

All these results follow the theoretical analysis in Section 4.2 and Section 4.3, which validates the effectiveness of the proposed methods in collecting sparse objects.

6 Conclusion

This paper presented a complete hug and haul framework for path planning and tracking control of object transport by deformable tube-connected MRS. The planning method constructs collision-free travelling corridors, generates smooth minimum-jerk Bézier paths, and reparametrizes them to maintain suitable inter-robot spacing. Centralized and distributed controllers were developed for the approaching, enclosing, and transporting stages, with an adaptive compensation term introduced to address unknown object and environmental resistance. The stability analysis showed that the robot position errors remain bounded while the reference curve evolves and converge to zero after the reference curve stops changing. Simulations and experiments with objects of different shapes, quantities, and spatial distributions demonstrated collision-free motion, stable path tracking, and successful object transport.

The proposed method still has several limitations. For example, we don't consider dynamic environments in this work; Fixed-radius inflation in generating the occupancy map and square convex corridors could lead to conservatism in the path planning; Transport of non-star-shaped objects is

still unresolved. Future work will focus on addressing these limitations.

A Proof of Proposition 1

Considering the length parameter condition in (25) and stacking the controller (24) for all robots, we can obtain the following control input for the tube-connected robots:

$$\bar{\mathbf{u}} = -k_1 \bar{\mathbf{L}}_c \mathbf{G}\left(s(t)\right) \boldsymbol{\xi}_e - k_2 \left(\bar{\mathbf{x}} - \mathbf{G}(s(t))\boldsymbol{\xi}\right) \tag{35}$$

where $\bar{\mathbf{L}}_c = \mathbf{L}_c \otimes \mathbf{I}_2$ is the extended Laplacian matrix.

The proof of Proposition 1 and other propositions is based on the following assumption:

Assumption 3. *The number of robots n and the degree of the reference path (the reparametrized Bézier curve) p_r satisfy the condition $n \leq p_r + 1$.*

Since $\mathbf{G} \in \mathbb{R}^{2n \times 2(p_r+1)}$ is a matrix of Bernstein basis functions, with Assumption 3, we can easily obtain that $\mathbf{G}$ is full row rank and its pseudoinverse exists and is given by $\mathbf{G}^+ = \mathbf{G}^\top (\mathbf{G}\mathbf{G}^\top)^{-1}$ (Barata and Hussein, 2012). With the pseudoinverse of $\mathbf{G}$ and from (22) and (23), we can obtain:

$$\mathbf{G}\boldsymbol{\xi}_e = \mathbf{G}\mathbf{G}^+\bar{\mathbf{x}} - \mathbf{G}\boldsymbol{\xi} = \bar{\mathbf{x}} - \mathbf{G}\boldsymbol{\xi} = \bar{\mathbf{x}}_e \tag{36}$$

Proof. Since the reference path is fixed, $\boldsymbol{\xi}$ is constant, while $\mathbf{G}(s(t))$ varies with the length parameters. Differentiating (22) and using (35) and (36) give

$$\begin{aligned}
\dot{\bar{\mathbf{x}}}_e &= \dot{\bar{\mathbf{x}}} - \frac{d}{dt}\left(\mathbf{G}(s(t))\boldsymbol{\xi}\right) \\
&= \bar{\mathbf{u}} - \dot{\mathbf{G}}(s(t))\boldsymbol{\xi} \\
&= -k_1 \bar{\mathbf{L}}_c \mathbf{G}(s(t))\boldsymbol{\xi}_e - k_2\left(\bar{\mathbf{x}} - \mathbf{G}(s(t))\boldsymbol{\xi}\right) - \dot{\mathbf{G}}(s(t))\boldsymbol{\xi} \\
&= -\left(k_1 \bar{\mathbf{L}}_c + k_2 \mathbf{I}_{2n}\right)\bar{\mathbf{x}}_e - \dot{\mathbf{G}}(s(t))\boldsymbol{\xi} \\
&= -\mathbf{F}\bar{\mathbf{x}}_e + \mathbf{d}(t)
\end{aligned} \tag{37}$$

where $\mathbf{F} = k_1 \bar{\mathbf{L}}_c + k_2 \mathbf{I}_{2n}$ and $\mathbf{d}(t) = -\dot{\mathbf{G}}(s(t))\boldsymbol{\xi}$. Because the interaction graph is connected and undirected, $\bar{\mathbf{L}}_c$ is symmetric and positive semidefinite. Hence, $\mathbf{F}$ is positive definite, with $\alpha := \lambda_{\min}(\mathbf{F}) = k_2 > 0$. By the variation-of-constants formula,

$$\begin{aligned}
\left\|\bar{\mathbf{x}}_e(t)\right\| &\leq e^{-\alpha t}\left\|\bar{\mathbf{x}}_e(0)\right\| + \int_0^t e^{-\alpha(t-\tau)}\left\|\mathbf{d}(\tau)\right\| d\tau \\
&\leq e^{-\alpha t}\left\|\bar{\mathbf{x}}_e(0)\right\| + \frac{1-e^{-\alpha t}}{\alpha}\left\|\mathbf{d}\right\|_{\infty,[0,t]}
\end{aligned} \tag{38}$$

where $\left\|\mathbf{d}\right\|_{\infty,[0,t]} = \sup_{\tau\in[0,t]}\left\|\mathbf{d}(\tau)\right\|$. This is an ISS estimate with respect to $\mathbf{d}(t)$. Since the Bernstein basis has bounded derivatives on the compact domain $s_i \in [0,1]$ and $\dot{s}_i = v_s$ while the robots move along the path, $\mathbf{d}(t)$ is bounded. Therefore, $\bar{\mathbf{x}}_e$ remains bounded while the length parameters vary, with

$$\limsup_{t\to\infty}\left\|\bar{\mathbf{x}}_e(t)\right\| \leq \frac{1}{k_2}\left\|\dot{\mathbf{G}}(s(t))\boldsymbol{\xi}\right\|_\infty \tag{39}$$

Suppose the length parameters stop changing at $t = t_s$. Then $\mathbf{d}(t) = \mathbf{0}$ for $t \geq t_s$, and (37) becomes

$$\dot{\bar{\mathbf{x}}}_e = -\mathbf{F}\bar{\mathbf{x}}_e \tag{40}$$

Consequently, $\left\|\bar{\mathbf{x}}_e(t)\right\| \leq e^{-\alpha(t-t_s)}\left\|\bar{\mathbf{x}}_e(t_s)\right\|$, so the position error converges to zero exponentially.

With (22) and (23), we can obtain that $\boldsymbol{\xi}_e = \mathbf{G}^+\bar{\mathbf{x}}_e + (\mathbf{G}^+\mathbf{G} - \mathbf{I}_{2(p_r+1)})\boldsymbol{\xi}$. Since $\bar{\mathbf{x}}_e$ converges to zero exponentially, we can obtain:

$$\lim_{t\to\infty}\boldsymbol{\xi}_e = \left(\mathbf{G}^+\mathbf{G} - \mathbf{I}_{2(p_r+1)}\right)\boldsymbol{\xi} \tag{41}$$

Substituting (41) into (35), we can obtain:

$$\lim_{t\to\infty}\bar{\mathbf{u}} = -k_1\bar{\mathbf{L}}_c \underbrace{\mathbf{G}\left(\mathbf{G}^+\mathbf{G} - \mathbf{I}_{2(p_r+1)}\right)}_{\mathbf{0}}\boldsymbol{\xi} = \mathbf{0} \tag{42}$$

Considering $\mathbf{u} = \mathbf{R}^{-1}\bar{\mathbf{u}}$ and $\mathbf{u}_i = [v_i, \omega_i]^\mathsf{T}$, we can derive the zero dynamics of robots' orientations $\dot{\theta}_i = 0$. This indicates that the orientations of the robots converge to constant values.

B Proof of Proposition 2

Since we add an object compensation term in the transporting controller, the stacked control input for the tube-connected robots during the transporting step can be written as:

$$\bar{\mathbf{u}} = -k_1\bar{\mathbf{L}}_c\mathbf{G}\boldsymbol{\xi}_e - \left(k_2 + \hat{k}\right)\left(\bar{\mathbf{x}} - \mathbf{G}\boldsymbol{\xi}(t)\right) \tag{43}$$

The proof of Proposition 2 depends on an extra assumption as follows:

Assumption 4. *The curve coefficients $\boldsymbol{\xi}(t)$ are bounded and the derivative of $\boldsymbol{\xi}(t)$ is bounded, i.e., $\left\|\boldsymbol{\xi}(t)\right\|_\infty < \infty$ and $\left\|\dot{\boldsymbol{\xi}}(t)\right\|_\infty < \infty$.*

Proof. Since $\mathbf{G}$ is constant and $\boldsymbol{\xi}(t)$ is time-varying, differentiating (22) and using (43) and (36) give

$$\begin{aligned}
\dot{\bar{\mathbf{x}}}_e &= \dot{\bar{\mathbf{x}}} - \mathbf{G}\dot{\boldsymbol{\xi}}(t) \\
&= -k_1\bar{\mathbf{L}}_c\mathbf{G}\boldsymbol{\xi}_e - \left(k_2 + \hat{k}\right)\bar{\mathbf{x}}_e - \mathbf{G}\dot{\boldsymbol{\xi}}(t) \\
&= -\mathbf{F}(t)\bar{\mathbf{x}}_e + \mathbf{d}(t)
\end{aligned} \tag{44}$$

where $\mathbf{F}(t) = k_1\bar{\mathbf{L}}_c + \left(k_2 + \hat{k}(t)\right)\mathbf{I}_{2n}$ and $\mathbf{d}(t) = -\mathbf{G}\dot{\boldsymbol{\xi}}(t)$. Since $\bar{\mathbf{L}}_c$ is positive semidefinite and $0 \leq \hat{k}(t) \leq k_{\max}$, $\lambda_{\min}(\mathbf{F}(t)) \geq k_2 > 0$. Therefore,

$$\frac{d}{dt}\left\|\bar{\mathbf{x}}_e\right\| \leq -k_2\left\|\bar{\mathbf{x}}_e\right\| + \left\|\mathbf{d}(t)\right\| \tag{45}$$

which yields

$$\left\|\bar{\mathbf{x}}_e(t)\right\| \leq e^{-k_2 t}\left\|\bar{\mathbf{x}}_e(0)\right\| + \frac{1-e^{-k_2 t}}{k_2}\left\|\mathbf{d}\right\|_{\infty,[0,t]} \tag{46}$$

Thus, the position error dynamics are ISS with respect to $\mathbf{d}(t)$. Under Assumption 4, $\mathbf{d}(t)$ is bounded, and hence $\bar{\mathbf{x}}_e$ remains bounded, with

$$\limsup_{t\to\infty}\left\|\bar{\mathbf{x}}_e(t)\right\| \leq \frac{1}{k_2}\left\|\mathbf{G}\dot{\boldsymbol{\xi}}(t)\right\|_\infty \tag{47}$$

When the curve coefficients stop changing at $t = t_s$, $\mathbf{d}(t) = \mathbf{0}$ and

$$\dot{\bar{\mathbf{x}}}_e = -\mathbf{F}(t)\bar{\mathbf{x}}_e, \qquad t \geq t_s \tag{48}$$

Therefore, $\bar{\mathbf{x}}_e$ converges to zero exponentially. Following the coefficient-error analysis in the proof of Proposition 1, substituting the limit in (41) into (43) gives $\lim_{t\to\infty}\bar{\mathbf{u}}=\mathbf{0}$. Considering $\mathbf{u}=\mathbf{R}^{-1}\bar{\mathbf{u}}$ and $\mathbf{u}_i=[v_i,\omega_i]^\mathsf{T}$, we obtain the zero dynamics $\dot{\theta}_i=0$. This indicates that the orientations of the robots converge to constant values.

C Proof of Proposition 3

Considering the length parameter condition in (25) and stacking the distributed controller (29) for all robots, we can obtain the following control input for the tube-connected robots:

$$\bar{\mathbf{u}}=-k_1\bar{\mathbf{L}}_c\mathbf{G}\left(s(t)\right)\boldsymbol{\xi}_e-k_2\boldsymbol{\Lambda}\left(\bar{\mathbf{x}}-\mathbf{G}(s(t))\boldsymbol{\xi}\right) \tag{49}$$

where $\boldsymbol{\Lambda}=\mathrm{diag}\left(\lambda_i\right)\otimes\mathbf{I}_2$ with λ_i defined by (30).

Proof. Since the reference path is fixed, $\boldsymbol{\xi}$ is constant. As in the proof of Proposition 1, differentiating (22) and using (49) and (36) give the following position error dynamics:

$$\dot{\bar{\mathbf{x}}}_e=-\mathbf{F}\bar{\mathbf{x}}_e+\mathbf{d}(t) \tag{50}$$

except that here $\mathbf{F}=k_1\bar{\mathbf{L}}_c+k_2\boldsymbol{\Lambda}$. Since the graph is connected and undirected, $\mathbf{F}$ is Hurwitz (Hong et al., 2006), i.e., positive definite. Let $\alpha=\lambda_{\min}(\mathbf{F})>0$. Applying the variation-of-constants formula to (50) yields

$$\left\|\bar{\mathbf{x}}_e(t)\right\|\le e^{-\alpha t}\left\|\bar{\mathbf{x}}_e(0)\right\|+\frac{1}{\alpha}\left(1-e^{-\alpha t}\right)\left\|\mathbf{d}\right\|_{\infty,[0,t]} \tag{51}$$

Therefore, the error dynamics are ISS with respect to $\mathbf{d}(t)$. As established in the proof of Proposition 1, $\mathbf{d}(t)$ is bounded while the length parameters vary, so $\bar{\mathbf{x}}_e$ remains bounded and satisfies

$$\limsup_{t\to\infty}\left\|\bar{\mathbf{x}}_e(t)\right\|\le\frac{1}{\alpha}\left\|\dot{\mathbf{G}}(s(t))\boldsymbol{\xi}\right\|_\infty \tag{52}$$

When the length parameters stop changing at $t=t_s$, $\mathbf{d}(t)=\mathbf{0}$ and

$$\dot{\bar{\mathbf{x}}}_e=-\mathbf{F}\bar{\mathbf{x}}_e,\qquad t\ge t_s \tag{53}$$

Thus, $\bar{\mathbf{x}}_e$ converges to zero exponentially. The coefficient-error relation used in (41) remains valid after $\mathbf{G}$ becomes constant. Substituting its limit into (49) gives $\lim_{t\to\infty}\bar{\mathbf{u}}=\mathbf{0}$. Considering $\mathbf{u}=\mathbf{R}^{-1}\bar{\mathbf{u}}$ and $\mathbf{u}_i=[v_i,\omega_i]^\mathsf{T}$, we obtain the zero dynamics $\dot{\theta}_i=0$. Therefore, the robots' orientations converge to constant values.

D Proof of Proposition 4

Considering the object compensation term (31), the stacked distributed control input for the tube-connected robots during the transporting step can be written as:

$$\bar{\mathbf{u}}=-k_1\bar{\mathbf{L}}_c\mathbf{G}\boldsymbol{\xi}_e-\left(k_2+\hat{k}\right)\boldsymbol{\Lambda}\left(\bar{\mathbf{x}}-\mathbf{G}\boldsymbol{\xi}(t)\right) \tag{54}$$

The proof of Proposition 4 also depends on Assumption 4.

Proof. Since $\mathbf{G}$ is constant, following the derivation in the proof of Proposition 2 gives

$$\dot{\bar{\mathbf{x}}}_e=-\mathbf{F}(t)\bar{\mathbf{x}}_e+\mathbf{d}(t) \tag{55}$$

except that here $\mathbf{F}(t)=k_1\bar{\mathbf{L}}_c+\left(k_2+\hat{k}(t)\right)\boldsymbol{\Lambda}$, while $\mathbf{d}(t)=-\mathbf{G}\dot{\boldsymbol{\xi}}(t)$. Since the graph is connected and undirected and at least one robot is a leader, $\mathbf{F}_0:=k_1\bar{\mathbf{L}}_c+k_2\boldsymbol{\Lambda}$ is positive definite. Moreover, $\mathbf{F}(t)\succeq\mathbf{F}_0$ because $\hat{k}(t)\ge 0$. Let $\alpha=\lambda_{\min}(\mathbf{F}_0)>0$. Following the same steps as in (46), we obtain

$$\left\|\bar{\mathbf{x}}_e(t)\right\|\le e^{-\alpha t}\left\|\bar{\mathbf{x}}_e(0)\right\|+\frac{1-e^{-\alpha t}}{\alpha}\left\|\mathbf{d}\right\|_{\infty,[0,t]} \tag{56}$$

Under Assumption 4, $\mathbf{d}(t)$ is bounded. Therefore, the position error dynamics are ISS and

$$\limsup_{t\to\infty}\left\|\bar{\mathbf{x}}_e(t)\right\|\le\frac{1}{\alpha}\left\|\mathbf{G}\dot{\boldsymbol{\xi}}(t)\right\|_\infty \tag{57}$$

When the curve coefficients stop changing at $t=t_s$, $\mathbf{d}(t)=\mathbf{0}$ and

$$\dot{\bar{\mathbf{x}}}_e=-\mathbf{F}(t)\bar{\mathbf{x}}_e,\qquad t\ge t_s \tag{58}$$

Hence, $\bar{\mathbf{x}}_e$ converges to zero exponentially. Following the coefficient-error and orientation analysis in the proof of Proposition 2, we obtain $\lim_{t\to\infty}\bar{\mathbf{u}}=\mathbf{0}$ and the zero dynamics $\dot{\theta}_i=0$, i.e., robots' orientations reach constant values.

References

Arvin F, Espinosa J, Bird B, West A, Watson S and Lennox B (2019) Mona: An affordable open-source mobile robot for education and research. *J. Intell. Robot. Syst.* 94(3–4): 761–775.

Barata JCA and Hussein MS (2012) The Moore–Penrose pseudoinverse: A tutorial review of the theory. *Braz. J. Phys.* 42(1–2): 146–165.

Bertoncelli F, Selvaggio M, Ruggiero F and Sabattini L (2022) Task-oriented contact optimization for pushing manipulation with mobile robots. In: *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*. pp. 1639–1646.

Chen Y, Wang W, Abdollahi Z, Wang Z, Schulte J, Krovi V and Jia Y (2018) A robotic lift assister: A smart companion for heavy payload transport and manipulation in automotive assembly. *IEEE Robot. Autom. Mag.* 25(2): 107–119.

Chung HY, Hou CC, Chen YS and Chao CL (2013) An intelligent service robot for transporting object. In: *Proc. IEEE Int. Symp. Ind. Electron. (ISIE)*. pp. 1–6.

Du Y, Zhang J, Cheng X and Cui S (2025) Swept volume-based continuous object gathering trajectory generation for tethered robot duo. In: *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*. pp. 7640–7647.

Ebel H, Rosenfelder M and Eberhard P (2024) Cooperative object transportation with differential-drive mobile robots: Control and experimentation. *Robot. Auton. Syst.* 173. Art. no. 104612.

Flores FG and Kecskeméthy A (2013) Time-optimal path planning for the general waiter motion problem. In: Kumar V, Schmiedeler J, Sreenivasan SV and Su HJ (eds.) *Advances in Mechanisms, Robotics and Design Education and Research*, *Mechanisms and Machine Science*, volume 14. Cham, Switzerland: Springer, pp. 189–203.

Hong Y, Hu J and Gao L (2006) Tracking control for multi-agent consensus with an active leader and variable topology. *Automatica* 42(7): 1177–1182.

Hu J, Liu W, Zhang H, Yi J and Xiong Z (2022) Multi-robot object transport motion planning with a deformable sheet. *IEEE Robot. Autom. Lett.* 7(4): 9350–9357.

Huang Y and Zhang S (2024) Cooperative object transport by two robots connected with a ball-string-ball structure. *IEEE Robot. Autom. Lett.* 9(5): 4313–4320.

Jiao Z, Niu Y, Zhang Z, Wu Y, Su Y, Zhu Y, Liu H and Zhu SC (2025) Integration of robot and scene kinematics for sequential mobile manipulation planning. *IEEE Trans. Robot.* 41: 5679–5699.

Karaman S and Frazzoli E (2011) Sampling-based algorithms for optimal motion planning. *Int. J. Robot. Res.* 30(7): 846–894.

Koung D, Kermorgant O, Fantoni I and Belouaer L (2021) Cooperative multi-robot object transportation system based on hierarchical quadratic programming. *IEEE Robot. Autom. Lett.* 6(4): 6466–6472.

Levé V, Moura J, Fujita S, Miyake T, Tonneau S and Vijayakumar S (2025) Scaling whole-body multi-contact manipulation with contact optimization. In: *Proc. 24th IEEE-RAS Int. Conf. Humanoid Robots (Humanoids)*. pp. 920–927.

Lynch KM (1992) The mechanics of fine manipulation by pushing. In: *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*. pp. 2269–2276.

Lynch KM and Mason MT (1996) Stable pushing: Mechanics, controllability, and planning. *Int. J. Robot. Res.* 15(6): 533–556.

Ma W, Hu J, Li J, Yi J and Xiong Z (2026) Stable kinematics for multi-robot collaborative transporting system with a deformable sheet. *IEEE Trans. Robot.* 42: 837–853.

Mason MT (1986) On the scope of quasi-static pushing. In: *Proc. Int. Symp. Robot. Res. (ISRR)*. pp. 229–233.

Mellinger D and Kumar V (2011) Minimum snap trajectory generation and control for quadrotors. In: *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*. pp. 2520–2525.

Mesbahi M and Egerstedt M (2010) *Graph-Theoretic Methods in Multiagent Networks*. Princeton, NJ, USA: Princeton Univ. Press. Ch. 2, pp. 14–41.

Oh KK, Park MC and Ahn HS (2015) A survey of multi-agent formation control. *Automatica* 53: 424–440.

Sewlia M, Verginis CK and Dimarogonas DV (2022) Cooperative object manipulation under signal temporal logic tasks and uncertain dynamics. *IEEE Robot. Autom. Lett.* 7(4): 11561–11568.

Siciliano B, Sciavicco L, Villani L and Oriolo G (2009) *Robotics: Modelling, Planning and Control*. London, U.K.: Springer. Ch. 11, pp. 507–508.

Su Y, Jiang Y, Zhu Y and Liu H (2022) Object gathering with a tethered robot duo. *IEEE Robot. Autom. Lett.* 7(2): 2132–2139.

Talia S, Thareja A, Mavrogiannis C, Schmittle M and Srinivasa SS (2023) PuSHR: A multirobot system for nonprehensile rearrangement. In: *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*. pp. 5380–5387.

Tang Z, Feng Y and Guo M (2024) Collaborative planar pushing of polytopic objects with multiple robots in complex scenes. In: *Proc. Robot.: Sci. Syst. (RSS)*. Delft, The Netherlands.

Tang Z, Zhang Y and Guo M (2026) PushingBots: Collaborative pushing via neural-accelerated combinatorial hybrid optimization. *IEEE Trans. Robot.* 42.

Tuci E, Alkilabi MHM and Akanyeti O (2018) Cooperative object transport in multi-robot systems: A review of the state-of-the-art. *Front. Robot. AI* 5. Art. no. 59.

Verginis CK, Zelazo D and Dimarogonas DV (2023) Cooperative manipulation via internal force regulation: A rigidity theory perspective. *IEEE Trans. Control Netw. Syst.* 10(3): 1222–1233.

Verhagen J and Tumova J (2025) Collaborative object transportation in space via impact interactions. In: *Proc. Robot.: Sci. Syst. (RSS)*. Los Angeles, CA, USA.

Wahrmann D, Hildebrandt AC, Schuetz C, Wittmann R and Rixen D (2019) An autonomous and flexible robotic framework for logistics applications. *J. Intell. Robot. Syst.* 93(3–4): 419–431.

Wan W, Fukui R, Shimosaka M, Sato T and Kuniyoshi Y (2012) Cooperative manipulation with least number of robots via robust caging. In: *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics (AIM)*. pp. 896–903.

Wang Z and Kumar V (2002) Object closure and manipulation by multiple cooperating mobile robots. In: *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, volume 1. pp. 394–399.

Wang Z and Schwager M (2016) Force-amplifying N-robot transport system (Force-ANTS) for cooperative planar manipulation without communication. *Int. J. Robot. Res.* 35(13): 1564–1586.

Xuan OW, Selamat H and Muslim MT (2023) Autonomous mobile robot for transporting goods in warehouse and production. In: *Proc. Int. Conf. Mechatronics Intell. Robot. (ICMIR)*. pp. 555–565.

Zhang B, Shao X, Zhi H, Qiu L, Romero JG and Navarro-Alarcon D (2024) Leader-follower formation control of perturbed nonholonomic agents along parametric curves with directed communication. *IEEE Robot. Autom. Lett.* 9(10): 8603–8610.

Zhang B, Zhi H, Romero JG, Labazanova L, Duan A, Li X and Navarro-Alarcon D (2023) Fourier-based multi-agent formation control to track evolving closed boundaries. *IEEE Trans. Circuits Syst. I, Reg. Papers* 70(11): 4549–4559.

Zhao Z, Li Y, Li W, Qi Z, Ruan L, Zhu Y and Althoefer K (2025) Tac-Man: Tactile-informed prior-free manipulation of articulated objects. *IEEE Trans. Robot.* 41: 538–557.

Zhi H, Zhang B, Qi J, Romero JG, Shao X, Yang C and Navarro-Alarcon D (2024) Non-prehensile object transport by nonholonomic robots connected by linear deformable elements. *IEEE Robot. Autom. Lett.* 9(10): 8651–8658.