Dual-arm Mobile Manipulation Planning of a Long Deformable Object in Industrial Installation

Yili Qin[1,2], Adrien Escande[3], Fumio Kanehiro[1,2], Eiichi Yoshida[4]

***Abstract*— In this paper, we present a planning method considering regrasp and mobility for the manipulation problem of a long deformable object in an industrial scenario. The object is carried through designated rollers by a dual-arm robot in the presence of environmental constraints. The problem is formulated by modeling the deformable object as a set of line segments and categorizing obstacles into different types. A hierarchical 2.5D manipulation planner is proposed, which first plans the ideal path of a deformable object in a 2D plane and obtains its topological representation. Next, dual-arm regrasps are determined by utilizing the deformation models, and collision-free paths are planned to meet topological constraints for the grippers. The feasibility of the proposed method is experimentally validated using a humanoid robot, by integrating the planner with a loco-manipulation planning framework and QP-based whole-body motion controller.**

I. Introduction

Manipulation of deformable objects with large length, such as cables (1D linear) or belts (2D plane), is frequently seen in daily life and industrial production sites. These objects have a high number of degrees of freedom, and their shapes can be changed by external forces and contacts with the environment. Modeling complexities and difficulties in obtaining their physical parameters make it hard to estimate their shape, and even more difficult to manipulate them.

The task of deformable object insertion is very common in robotic surgery, e.g. inserting a string to a small opening [34], steering a needle in soft tissues [2]. These methods attempt to overcome the challenges from a control viewpoint. However, some problems like cable routing or assembly requires taking a broader planning viewpoint.

In this paper, inspired by an industrial belt installation task in a large-scale assembly (Fig. 1), a solution is proposed for the manipulation problem of a long deformable object with a dual-arm mobile robot. Figure 1 (a) shows the scenario of the belt assembly task. There is a bobbin inside an installation station. The belt is drawn from the bobbin, and passed around the rollers, which constrain the final configuration of the belt. Based on observation for a human's demonstration (Fig. 1(c)), we identified the three features: (1) two hands mainly handle the tail of the belt from the side. Further, to avoid collisions with the surrounding equipment, two hands alternately regrasp the belt to change the grasp point. (2) The tail of the belt should move along a specific path to meet the requirements of the installation task. (3) After the assembly is complete, there is no collision between the tensioned belt and the surrounding equipment.

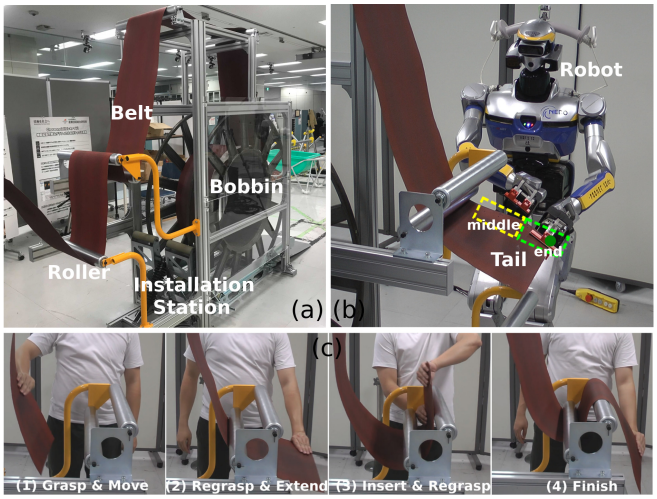


Fig. 1. (a) Scenario of the belt assembly task. (b) On one edge of the belt, we define tail end-point (green point), tail end-part (green box) and tail middle-part (yellow box) for robot manipulation. We set the end-point as the origin position for the grasp and regrasp on the belt. (c) Illustration of the belt assembly task on two rollers by a human.

A. Related Works

1) Topological Representation for a Deformable Object: Numerous studies on manipulating deformable objects focus on specific shapes [37] [7] [26]. Model-based methods like mass-spring systems, finite element methods (FEMs) require accurate physical parameters of the deformable objects, which are often difficult to obtain [35]. Data-driven based methods are therefore investigated in learning a deformation model [36], and further used in learning strategies in manipulation [1] [15].

On the other hand, deformation also brings benefits for manipulation. Collisions with surrounding environment are often not considered as an issue, therefore the exact shape becomes less important. More attention is paid to the relationships between different parts of the deformable object itself [20], or between the object and the environment [38] [18]. An intuitive spatial representation method considering environments was proposed for handling the state transitions for a rope [8]. In a more systematic way, based on the invariants of homology and homotopy in topology, Bhattacharya et al. formulated the problem of separating multiple objects with a flexible cable [3]. Their topological representation method is adopted in our study (Fig. 2).

[1] CNRS-AIST JRL (Joint Robotics Laboratory), IRL3218, and National Institute of Advanced Industrial Science and Technology (AIST), Japan
[2] Intelligence Information Technologies, University of Tsukuba, Japan
[3] INRIA Grenoble - Rhône-Alpes, France
[4] Department of Applied Electronics, Tokyo University of Science, Japan

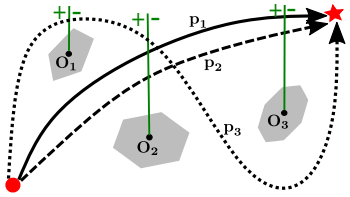


Fig. 2. Homotopy and homology equivalences, and the topological representation [3]: the o_1, o_2 and o_3 indicate the obstacles, while p_1, p_2 and p_3 indicate the paths from the start position (red point) to the goal position (red star). Path p_1 and path p_2 are both homotopic and homologous, while path p_3 is not homotopic and not homologous with them. Following [3], we associate vertical rays with each obstacle, defining a "+" and a "-" side. We note "+" when a path crosses a ray from "+" to "-", and "-" when crossing from "-" to "+". The topological representation for path p_3 is "$o_1^+ o_2^+$", while p_1 and p_2 have the same topological representation "$o_2^+ o_3^+$".

2) Motion Planning for a Deformable Object: Motion planning problem for rigid bodies have been well studied [14]. The preliminary work on motion planning for a deformable elastic plate was developed by Kavraki et al. [10]. Also, based on the equilibrium configurations of an elastic rod [4], a variant of Rapidly-exploring Random Tree (RRT) algorithm for sampling in the admissible space was presented in [19]. Saha et al. proposed a topologically-biased Probabilistic Roadmap (PRM) algorithm to solve the knot problem [20]. Another variant of PRM was used in cable routing near contact space [9].

Research on the navigation and planning of tethered robots can be found in [11] [24]. While they addressed a navigation under cable constraints, manipulation is an intrinsically different problem. The focus is on how the robot interacts with the deformable object and bring it to a goal state. [21] investigated a method to handle a tethered tool while avoiding cable entanglements, and a planner for dual-arm robot to generate motion sequences was proposed. [22] further studied collaborative manipulation for two robots.

Cable routing problem requires robots to place cables to some specific fixtures and has been widely investigated [8] [33] [38]. However, there is still a lot of work to be done to apply these methods to practical applications, including work on more complex scenarios, with possible regrasp needs. To solve this problem, the National Institute of Standards and Technology (NIST) launched several task boards for cable assembly [12], and some were used in competitions in recent years. Most of the solutions pre-set fixed routing positions, and used fixed motions like pulling and wiggling to fit the cable into the open groove. There is still much room for improvement in automation [28].

For our case, motion planning for a long deformable belt can be simplified to plan the motion of its tail. When a gripper holds the tail and moves, by taking advantage of the nature of deformation, we can consider the tail as a tethered point rather than a tethered rigid rod or plate, which will simplify the planning problem and improve the efficiency [11] [24]. When regrasp or handover the tail to the other gripper, treating the tail as a rigid body with dimensions will facilitate its manipulation [27]. One classic methods of moving a bar through obstacles with regrasp motion by a manipulator was presented in Simeon's research [25], some other work on dual-arm regrasp can be found in [23] [31]. However, simply modeling the deformable tail as a rigid body in the planner will cause the following issues: (1) the planner may fail in some small spaces due to the choice of the size for the rigid body; (2) regrasp position on the tail middle-part is not determined by this rigid model, but by the contact state with the rollers [21]. Our methods take into account the difference in modeling the deformable tail for different manipulations. In the high-level deformable object path planner, the tail is modeled as a point. In low-level gripper path planner, deformation properties are introduced. The proposed hierarchical planner is suitable for our deformable object assembly task and has a good performance in planning time.

3) Robot Loco-manipulation Planning: Mobility of the robot provides the capability of manipulating a body over a wide range. Studies in handling a deformable object with movable robots can be found in [29] [18]. By utilizing the reachability map for a robot [31] [30], Murooka et al. proposed a planning framework that quickly finds the state transitions between locomotion and object manipulation of a humanoid robot [17]. In their research, the objects are restricted to rigid objects and the regrasp point on an object is fixed. On the other hand, regrasp position on the deformable object need to be changed in our case. We also extend the framework so that the planner can handle multiple successive paths, rather than just one in their method.

B. Our Contributions

We highlight the specificity of the task we tackle: the robot needs to move to bring a large scale deformable object with a non-negligible weight to pass through obstacles, that induce discrepancies between the deformable object trajectory and the grippers trajectory, and require the robot to change grasps on the object. There are several important aspects to address this challenging problem: topological representation, manipulation planning including regrasp and robot mobility. Our contributions are summarized as follows:

- We present a hierarchical 2.5D (combines 2D object and 3D gripper motion) manipulation planner for 2D shape with height that incorporates the Deformable Linear Object (DLO) and dual-arm manipulation.
- The proposed planner is experimentally validated in an industrial scenario by integrating with a loco-manipulation planner [17] and a whole-body motion controller [32]. The installation station we use and its specifications comes directly from the industry. The proposed approach covers the whole process from the models of the robot and environment to the execution on a real robot.

The paper is organized as follows: after formulating the problem in Sec. II, the details of the 2.5D manipulation planner are given in Sec. III. Sec. IV, introduces the loco-manipulation planner and QP-based whole-body motion controller. Experiments and results are presented in Sec. V.

II. Problem Setting

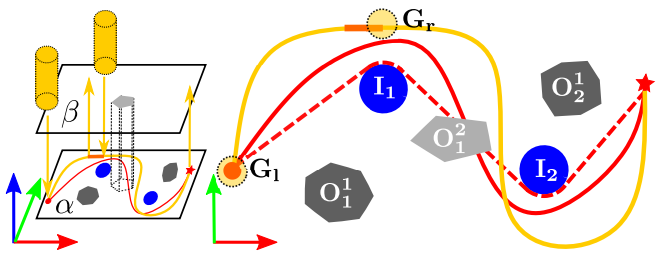


Fig. 3. Illustration of the 2.5D manipulation problem for a DLO. The left figure shows different planes for the motion, the right figure is the top view of the left figure. The blue circles, dark grey polygons and light grey polygon indicate the rollers, type1-obstacles and type2-obstacle respectively. The dashed red line indicates the ideal configuration of the DLO specified by the task. The red line indicates one possible final configuration of the DLO from the start position (red point) to the goal position (red star). Motion of the DLO is in 2D plane α, where the type2-obstacle is in different plane β. The yellow lines indicate the paths for the grippers (yellow cylinders and circles). The grippers are able to traverse between two planes α and β through vertical paths. Since the height of the grippers spans two planes, type2-obstacle should also be considered by grippers' motion. The orange line indicates a regrasp motion, which provides a connection between two paths by the DLO, thus the paths in $\tau^{\mathcal{G}}$ can be regarded as a continuous.

A. Some Assumptions

To model the task of robot manipulation (Fig. 1 (b)), we made the following assumptions: (1) from the side view, the belt can be considered as a cable in 2D plane without considering the twisting. This allows us to treat a deformable object with 2D plane structure in 3D space as a 1D deformable linear object in 2D plane equally. The grippers are simplified to cylinders, and modeled as circles in a 2D plane. (2) Two grippers alternately regrasp the tail, but do not cross their grasp positions on the belt. This is to avoid difficult configurations for dual-arm robots and simplify the manipulation. (3) During the assembly, the belt gradually expands its length and remains sufficiently stretched without slackening. This allows regarding the belt as line segments connected by rotatable joints.

B. Problem Formulation

Based on the assumptions, we formulate the Deformable Linear Object (DLO) assembly problem (Fig. 3). The elements included in are a DLO $\mathcal{L}$ in 2D plane, two grippers $\mathcal{G} = \{G_l, G_r\}$ of a robot, U rollers $\mathcal{I} = \{I_u\}$, where $u \in [0, U]$, and multiple obstacles $\mathcal{O}$. We categorize the obstacles in two different types: (1) type1-obstacles for the deformable object and also for the grippers, we denote it as $\mathcal{O}^1 = \{O_m^1\}$, $m \in [0, M]$; (2) additional type2-obstacles just considered by the grippers, which are denoted as $\mathcal{O}^2 = \{O_n^2\}$, $n \in [0, N]$. The configuration of the DLO is denoted by $q \in \mathcal{C}_{\mathcal{L}}$, where $\mathcal{C}_{\mathcal{L}}$ is the configuration space. We denote its ideal configuration as q_d, in which the DLO has the shortest length between the start and goal positions after assembly. We denote the position of the tail of the DLO in the 2D plane as $X \in R^2$, and a path for a gripper as $\tau(t) \in SE(2)$. Topology of a path or configuration is denoted as $\mathcal{T}$.

The DLO assembly problem looks for a path $\tau^{\mathcal{G}}$ that brings the tail from the start position X_s to the goal position X_e. The path $\tau^{\mathcal{G}}$ consists of an unspecified number $K \geq 1$ subpaths in sequence for the two grippers $\tau^{\mathcal{G}} = (\tau_1^{g_1}, \tau_2^{g_2}, ..., \tau_K^{g_K})$, where the g_i alternate between left and right gripper. The path $\tau^{\mathcal{G}}$, the DLO's final configuration q_e and ideal configuration q_d should satisfy $\mathcal{T}(\tau^{\mathcal{G}}) = \mathcal{T}(q_e) = \mathcal{T}(q_d)$, and the subset $\mathcal{C}(\mathcal{T}(q_e)) \subset \mathcal{C}_{\mathcal{L}}$. Here, since there is a regrasp motion for the tail between two subpaths, we regard the path $\tau^{\mathcal{G}}$ as a continuous path.

From the task description and problem definition above, we are solving a 2.5D object manipulation problem: the motion of the object is in a 2D plane, but the grippers can traverse different 2D planes, and the motion is in 3D space.

III. Path Planning for a DLO

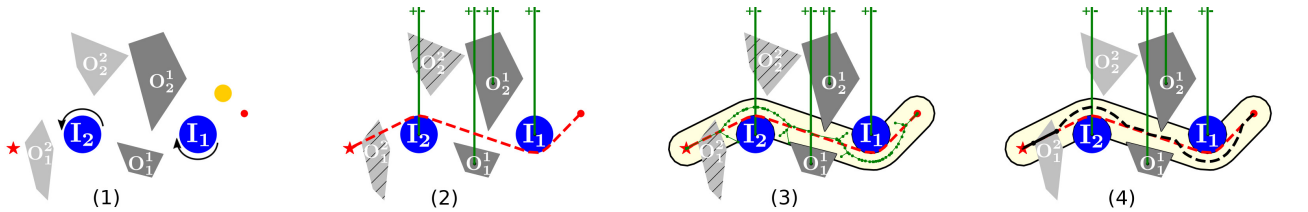
We simplify the full path planning problem to finding paths for the grippers grasping the tail of the DLO. In this section, we introduce a hierarchical planner considering object deformation and robot regrasp. In the DLO path planner part, the focus is placed on obtaining the information to guide the DLO to the goal. We find a topological representation for the ideal path, and use a topology-equal test path to find potential regrasp positions. Then, this information is sent to the gripper path planner to find feasible regrasps and paths of the grippers.

The following are the input to the planner (Fig. 4 (a-1)): (1) the start and goal positions; (2) the order, position, size, and wrapping direction for each roller; (3) the polygons of the type1-obstacles and type2-obstacles; (4) the size of the gripper. The output of the planner is paths for the two grippers in the 2D plane, including the vertical paths leading to the regrasp positions.

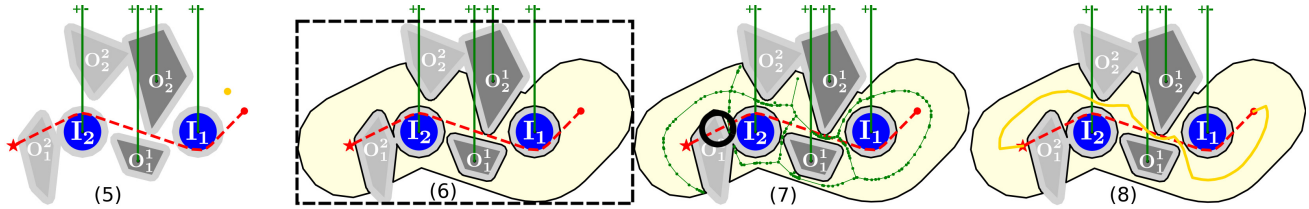
A. DLO Path Planner

1) Ideal Path and Its Topological Representation: When a gripper grasps the DLO, the grasp point on the object adds a position and orientation constraint to its configuration. Regardless of the gripper, we consider the grasp point as a free-flying point and create the configuration space (C-space). The C-space of a point is the same with its workspace. The ideal path, or ideal configuration of a DLO is the shortest path between the start and goal positions. This optimal solution is composed of straight line segments and circular arcs on the rollers. Using geometric methods, an ideal path from the start position to goal position can be quickly generated, following [38]. According to the problem setting, no type1-obstacles should block the ideal path, where type2-obstacles can be anywhere and may block it. The relationships between the ideal path, the rollers, and the type1-obstacles provide the information to guide the grippers to the goal. To let the planner utilize this information, we create the topological representation for the ideal path (Fig. 4 (a-2)).

2) Tunnel Constraint, Voronoi Diagram and Test Path: In the C-space of the grasp point, a tunnel $\mathcal{S}^g$ with width w_s^g along the ideal path is created. The purpose of the tunnel is to find a test path, which will be used to find potential regrasp



(a) DLO path planner.



(b) Gripper path planner.

Fig. 4. Process of the 2.5D planner (Scenario-1). (1) As the input, we have the start (red point) and goal (red star) positions, two rollers with specified wrapping directions in blue, two type1-obstacles in dark grey and two type2-obstaces in light grey. The yellow circle indicates the gripper. (2) The topology of the environment is created from the rays in green. The type2-obstacles are excluded from the representation. The dashed red line indicates the ideal configuration, and its topological representation is "$(O_1^1)^-(I_2)^-$". (3) The light yellow area indicates the tunnel along the ideal path. The Voronoi diagram is in green in the tunnel. (4) The dashed black line indicates the test path. It is topologically equivalent to the ideal path. The black solid line on obstacle O_1^2 indicates a potential regrasp position. (5) In the C-space of the gripper, the gripper can be considered as a point. The rollers and obstacles are extended according to the size of the gripper. Some areas between them are blocked. (6) A new tunnel in yellow is created along the ideal path. To include larger area in searching, the tunnel width here is much bigger than the one in step 3. The dashed box indicates the scenario area. (7) The black circle indicates that use the QRTR (Fig. 6) on the potential regrasp position from step 4. The Voronoi diagram is also created in the tunnel. (8) A path in yellow is found by using the DFS on the graph. In this example, there are no regrasp motions in the solution, this path can be assigned to a gripper.

positions. The test path should have the same topology with the ideal path. Unlike the ideal path that clings to the rollers, the test path has a certain distance between the rollers and the type1-obstacles, making it easier to use. We should guarantee $w_s^g > 0$ to make the test path deviate from the ideal path.

The Voronoi diagram method generates roadmaps in the free space. The roadmaps have the property of maintaining an equal distance from surrounding obstacles. In the free space of the tunnel, roadmaps are generated, and the diagram is then created (Fig. 4 (a-3)). We cast the Voronoi diagram as a directed graph where each edge of the diagram corresponds to two directed edges in the graph, one in each direction. We apply Depth-First Search (DFS) to the graph to enumerate all paths that do not use the same edge twice, and we keep only those with the same topology as the ideal path. The shortest feasible path is selected as a test path.

3) Potential Regrasp Position: A potential regrasp position is a place blocking a gripper's movement along the test path. The grippers need a regrasp motion in 3D space to change the grasp point on the tail, then continue to bring the DLO to move. The following criteria are used to detect a potential regrasp position along the test path (Fig. 4 (a-4)): (1) the test path passes through a type2-obstacle; (2) the test path passes the space between two rollers, two obstacles, or a roller and an obstacle, where the distance between the two objects is smaller than the gripper size.

B. Gripper Path Planner

We denote the size of the gripper as w_g and create the C-space of the gripper (Fig. 4 (b-5)). The closer the paths of the grippers to the ideal path, the better to reduce the extension of the deformable object. A tunnel $\mathcal{S}^l$ with width

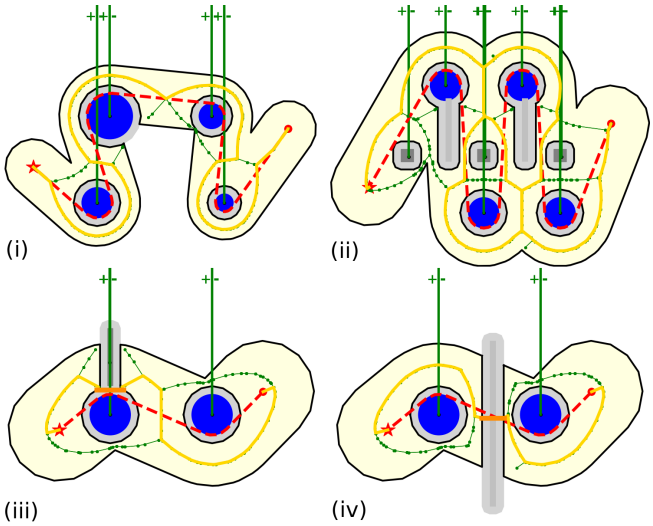


Fig. 5. Some test cases. The paths in yellow indicate the solution for each case. (i) Four rollers with different widths. (ii) Scenario-2: four rollers, three type1-obstacles and two type2-obstacles. (iii) A type1-obstacle is close to the roller, and the space is not big enough for the gripper to pass through. There is a regrasp motion (orange line) in the solution. The regrasp is generated by checking the QRTR and separates the path to two subpaths, which can be assigned to two grippers respectively. (iv) A type2-obstacle blocks the tunnel.

w_s^l is therefore introduced to limit the grippers' movement (Fig. 4 (b-6)). To find a feasible path to the goal, the tunnel width should guarantee $w_s^l > w_g$ and $w_s^l \geq w_s^g$ to cover the tunnel $\mathcal{S}^g$. We often let $w_s^l \geq max(w_s^g, 2w_a + w_g)$ to cover all rollers, where w_a is the maximum size of the rollers. The wider tunnel means the larger area will be included in

searching, thus the higher possibility to find a solution. On the other hand, bigger width also brings larger offset of the planned paths to the ideal path. To facilitate the comparison of tunnel widths in different scenarios (Table I in Section V), we define the tunnel coverage for a scenario, which is the ratio of the tunnel area to the scenario area. The scenario area is the area of smallest rectangular that includes start point, goal point, rollers, obstacles and tunnel.

Similar to the DLO path planner, for the gripper path planner, we create the Voronoi diagram in the free space of the tunnel. Because of the extension of the size of the rollers and obstacles in the gripper's C-space, some previously connected areas may be blocked. Due to loss of connectivity of some regions, searching in the graph may not get the solution. A regrasp motion (Fig. 5 (iii, iv)) to the tail end-part of the DLO therefore provides a connection between two blocked regions. To repeat the regrasp motion to the end-part, as a preparatory action, the regrasp to the tail middle-part should be completed first. By performing the regrasps on the end-part and middle-part alternately, the tail can be passed through the small space to the goal. Next, we will discuss how to plan these regrasps on a DLO.

1) Regrasp the Tail End-part: When the tail middle-part is held by one gripper, the other gripper can regrasp the tail end-part. To regrasp the end-part, knowing its configuration is the first step. Degrees of deformation of the end-part depends on the material's properties, the length, and the direction relative to the gravity. These make it difficult to determine the position to regrasp. Since a DLO usually becomes stiff when short enough, its tail can be regarded as a straight line, within a tolerable error, for a certain length.

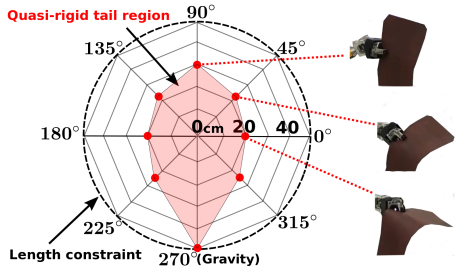


Fig. 6. Illustration of the Quasi-Rigid Tail Region (QRTR).

Within this length, when the gripper holds the tail middle-part and rotates with different angles, the tail will sweep a region. As an ideal situation, this length is a fixed value for a DLO, and this region will become a circle. Due to the gravity, the length will change on different directions, we therefore define the Quasi-Rigid Tail Region (QRTR) for the DLO and use it in planning. In Fig. 6, the black dashed circle indicates the length limitation specified in the task. The red points in different directions indicate the maximum tolerable length in that direction. The QRTR can be created by connecting adjacent red points. The QRTR is obtained empirically and its precision depends on the number of points measured.

2) Regrasp the Tail Middle-part: Based on the assumption 3 in Section II-A, the shape of the DLO between two rollers, between the grasp point and the adjacent roller can be considered as connected straight line segments. The configuration of the DLO can be therefore inferred from the position of the grasp point in the workspace and the DLO's contact state with the rollers. This Contact State with Environment (CSE) should be continuously updated during the assembly process.

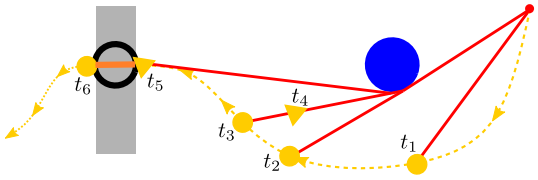


Fig. 7. Illustration of the use of QRTR and CSE in planning. The yellow dashed line indicates a path from the planner. On the path, a regrasp motion in orange has been found on a type-2 obstacle (grey rectangle) by checking the QRTR (black circle). From time t_1 to t_3, the yellow circle gripper holds the tail end-part and moves along the path. The DLO is extended and creates a contact with the roller. Since the planner knows there will be a regrasp motion on the obstacle, a regrasp motion on the tail middle-part by the triangle gripper is thus generated by the planner. According to the CSE at time t_4, the DLO's shape can be known. The regrasp position on the middle-part can be determined by the information of the orange regrasp motion. Then, the circle gripper releases the DLO, and the triangle gripper continuous to move with the DLO. At time t_5 and t_6, two grippers pass the tail through the obstacle. The tail end-part is back in the circle gripper again.

The QRTR is first used to the potential regrasp positions to check if there are feasible regrasps on different directions. Feasible regrasps will be added to the roadmaps of the Voronoi diagram. Then the DFS is used on the graph to find topology-equal paths to the ideal path. Some paths may contain regrasps, some may not. By computing the cost for each path considering the path length, the time required for a regrasp motion and so on, we can choose an optimal path (Fig. 4 (b-7, b-8)). If the optimal path includes multiple subpaths that separated by regrasps, the CSE is then used on each subpath to insert a regrasp motion. The optimal path is divided into more subpaths, which will be assigned to the grippers alternately. In this way, we get the paths for the grippers. Figure 7 shows how to utilize the QRTR and CSE in planning regrasp motion.

In the above discussion, we actually separated the planning of position and the planning of orientation of a gripper. During the assembly, the gripper needs to control the tail end-point to track the path from the planner. In this way, the rotation of a gripper can be planned. When planning for the regrasp, the planner pays attention for the regrasp at time t_3 (Fig. 7) for enough space at time t_4 by extending the DLO (refer to the second regrasp in Fig. 9 (i) Scenario-3). The process of the DLO path planner and the gripper path planner can refer to Fig. 4 and their pseudocode (see Alg. 1, 2). More test cases can be found in Fig. 5.

IV. Loco-manipulation Planning and Whole-body Motion Control

To let a humanoid robot move in front of the installation station, regrasp the deformable object and move it along a

Algorithm 1 DLO Path Planner

compute ideal path q_d with input: (X_s, X_e, $\mathcal{I}$, $\mathcal{O}$)
get topology $\mathcal{T}(q_d)$ with input: (q_d, $\mathcal{I}$, $\mathcal{O}^1$)
create tunnel $\mathcal{S}^g$ with input: (q_d, w_s^g)
free space of the tunnel $\mathcal{S}_f^g$ with input: ($\mathcal{S}^g$, $\mathcal{I}$, $\mathcal{O}$)
create Voronoi graph $\mathcal{V}^g$ with input: $\mathcal{S}_f^g$
compute a feasible path as the test path p_t with input: (X_s, X_e, $\mathcal{I}$, $\mathcal{O}$,$\mathcal{V}^g$, $\mathcal{T}(q_d)$) (refer to Sec. III-A.2, par. 2)
if (p_t passes through $\mathcal{O}^2$) **or** (p_t passes between two objects in $\{\mathcal{I},\mathcal{O}\}$ **and** their *Distance* $< w_g$) **then**
    add potential regrasp to $\mathcal{R}_p$
end if
return q_d, $\mathcal{R}_p$

Algorithm 2 Gripper Path Planner

create tunnel $\mathcal{S}^l$ with input: (q_d, w_s^l)
free space of the tunnel $\mathcal{S}_f^l$ with input: ($\mathcal{S}^l$, $\mathcal{I}$, $\mathcal{O}$, w_g)
create Voronoi graph $\mathcal{V}^l$ with input: $\mathcal{S}_f^l$
for each potential regrasp r in $\mathcal{R}_p$ **do**
    if r is feasible by evaluating with QRTR **then**
        put regrasp r into $\mathcal{R}_{qrtr}$
    end if
end for
compute feasible paths $\mathcal{P}_f$ with input: (X_s, X_e, $\mathcal{I}$, $\mathcal{O}$, $\mathcal{R}_{qrtr}$, $\mathcal{V}^l$, $\mathcal{T}(q_d)$)
select an optimal path p_o from $\mathcal{P}_f$
if *sizeof*($\mathcal{R}_{qrtr} \cap p_o$) $= 0$ **then**
    set path p_o as $\mathcal{P}_{qrtr}$
else
    compute paths $\mathcal{P}_{qrtr}$ with input: (p_o, $\mathcal{R}_{qrtr}$)
end if
use CSE to create regrasps $\mathcal{R}_{cse}$ with input: (p_o, $\mathcal{P}_{qrtr}$)
compute paths $\tau^{\mathcal{G}}$ for the grippers with input: (p_o, $\mathcal{R}_{qrtr}$, $\mathcal{R}_{cse}$, $\mathcal{G}$)
return $\tau^{\mathcal{G}}$

path, we need to generate its gait and whole-body motion. The constraints on the feet and grippers will change during this process, and thus lead to a multi-modal problem. Reachability map [30] is a method for quickly checking the feasibility of a posture by pre-computing the inverse kinematics (Fig. 8). Adapting the approach of [17], we can quickly compute the movement of the gripper, as well as the movement of the object: to find the state transitions between the robot locomotion and object manipulation, we formulate it as a graph search problem. The state of the robot is described as $\{x_{st-foot}, l_{st-foot}, x_{sw-foot}, l_{sw-foot}, x_{obj}, l_{grip}\}$, where x indicates the positions and l indicates the label, for the stance foot, swing foot, object, and the grippers. The successors of the state can be generated from current state by sampling in the range of the movement of the feet and grippers. An Anytime Dynamic A^* is then used to search a solution from the start to the goal state (Fig. 8).

Whole-body motions of the robot are generated by the multi-objective QP-based controller [32]. A stabilizer task

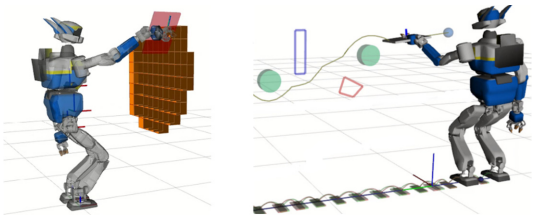


Fig. 8. Left: the reachability map for the right gripper of the humanoid robot HRP-2Kai, which is encoded as a set of cells on a 3D grid. In this paper, each cell represents a reachable couple (position in a given vertical plane, rotation around the normal to that plane) for the gripper (cells for the same position superimpose in the figure). For a given robot and distance to the plane, the map can be computed once and for all. Right: the generated gripper movements and corresponding footsteps obtained by using the reachability map. The plate in the gripper can be regarded as the tail of the DLO.

for the balance [5], two impedance tasks for the grippers, and two surface-transform tasks for the soles of the left and right foot are implemented in the controller [16]. The controller receives the footsteps and grippers' motions from the loco-manipulation planner, generates corresponding tasks and formulates them as a QP problem. The solver then generates trajectories for each joint, the robot can make complex whole-body motions in this way.

V. Experiments and Results

A. Belt Path Planning on the Mockup

The CAD model of the mockup of installation station is expressed as circles and polygons, which include 5 rollers, 0 type1-obstacles and 16 type2-obstacles, and used as input to the 2.5D manipulation planner. The maximum size of the rollers is $w_a = 6$ cm, and the width of the gripper is $w_g = 12$ cm. We set the tunnel widths as $w_s^g = w_s^l = 50$ cm. To guarantee the path can be implemented by our robot, we introduced some constraints to the planner, e.g. rotation limitation of the grippers, the range of movement of the grippers and so on. The start and goal positions are then provided as an inquiry to the planner (Fig. 9). The QRTR of the belt comes from the experiments that obtain the lengths in different sampled directions (Fig. 6). The maximum length is limited to 50 cm. In the direction of gravity, the length of the belt can reach 50 cm. The lengths at $0°$ and $180°$ are 20 cm, and the length at $90°$ is 30 cm.

The planning time of the 2.5D manipulation planner in different scenarios is shown in Table I. It is mainly related to the number of rollers (U), type1-obstacles (M) and type2-obstacles (N), and the coverage of the tunnel $\mathcal{S}^l$. The planner spends most of the time in the search of topology-equal paths using the DFS algorithm. The higher coverage of the tunnel means more likely to find loops in the graph. The planner is incomplete because there is no guarantee that a feasible regrasp will be found on a potential regrasp position. But it is good enough to use for our case. A complete planner will be an interesting problem in the next research. To get 0.75 m lateral footsteps to the Scenario-3, it takes about 5.5 seconds to inquiry four paths of the grippers in the loco-manipulation planner. The robot being in close proximity to

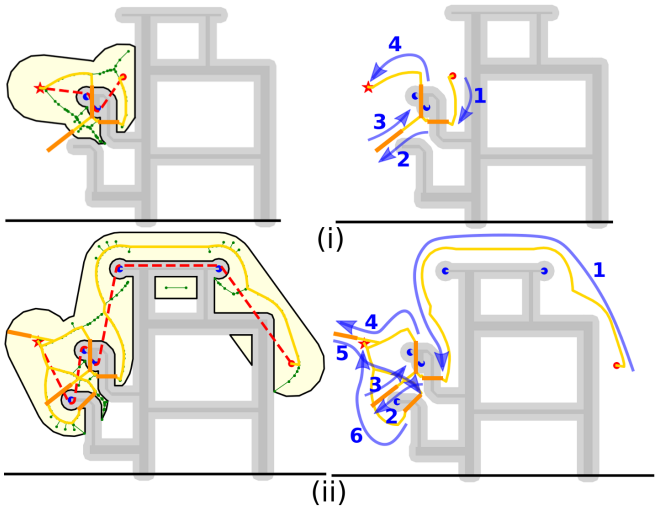


Fig. 9. Belt path planning on the mockup. (i) The planning result of Scenario-3: four paths of the grippers in yellow, and three regrasp motions in orange. The blue lines indicate the directions for the paths in sequence. At the beginning, the right gripper grasps the tail middle-part of the belt at 17.6 cm ($<$ 20 cm at 180°), pulls the belt along the first path. Then the left gripper regrasps the tail end-part at 0 cm (tail end-point), and pulls the belt along the second path. Then the right gripper regrasps the middle-part at 26.7 cm ($<$ 30 cm at 90°) and moves along the third path, inserts the end-part of the belt to the two rollers. At last, the left gripper regrasps the end-part at 0 cm and pulls it along the fourth path to the goal. (ii) The planning result of Scenario-4 includes six paths for the grippers and five regrasp motions for the belt.

the installation station, we maintain a slow execution speed for safety reasons.

TABLE I

PLANNING TIME OF DIFFERENT SCENARIOS

Scenario	U	M	N	Tunnel Coverage ($\mathcal{S}^I$)	Time (ms)
Scenario-1	2	2	2	45%	48
Scenario-2	4	3	2	74%	1.8×10^3
Scenario-3	2	0	16	13%	65
Scenario-4	5	0	16	42%	197

B. Belt Assembly by the Humanoid Robot HRP-2Kai

The map of the environment is created with the Azure Kinect DK depth camera and the RTAB-Map library before the experiment starts [13] [6]. The CAD models of the mockup and the ground are registered to their point clouds by using the ICP method implemented with CUDA, thus we can get the mockup's position in the map. The grippers' paths and the footsteps are positioned according to the mockup's position. At the beginning of the experiment, the robot is placed close to the initial position of the footsteps. During the locomotion, visual SLAM is used to fix the errors of the robot's position and the grippers' motion. The screenshots of the experiment of Scenario-3 are shown in Fig. 10. It takes about 124 seconds to assemble the belt to the middle two rollers of the mockup. These two rollers are close to each other, and there are many obstacles around them, they are the most difficult part of the installation task. The accompanying video shows the experiment with the humanoid robot.

VI. Conclusion and Future Work

In this paper, we presented our work on the DLO manipulation planning problem for 1D long deformable objects, and some 2D long deformable objects that do not need to consider twisting. To solve the problems (1) what paths the deformable object and the grippers should go, (2) how to decide the regrasps on the deformable object, we proposed a hierarchical 2.5D manipulation planner. The QRTR and the CSE are used to determine the regrasps on the deformable object. To let the robot complete the belt assembly task automatically, we incorporate the 2.5D planner with the loco-manipulation planning framework and QP-based whole-body controller. Feasibility of our methods is verified by the experiments on humanoid robot HRP-2Kai. Although we mainly dealt with a humanoid robot throughout the discussion, the method is not limited to this particular type. The loco-manipulation framework based on reachability map can be readily extended to other dual-arm mobile robots.

Currently, the tunnel widths are fixed by hand. In future work, to remove them by optimal planning methods would be a good research direction. In addition, the execution time could also be improved. Methods that combine visual detection, the QRTR, and the CSE to improve robustness will also be an interesting research topic.

VII. Acknowledgement

We would like to thank Dr. Murooka Masaki for sharing the source code of the loco-manipulation planner, and giving advice on the framework extension.

References

[1] Learning where to trust unreliable models in an unstructured world for deformable object manipulation. *Science Robotics*, 6(54):eabd8170, 2021.

[2] R. Alterovitz, K. Goldberg, and A. Okamura. Planning for steerable bevel-tip needle insertion through 2d soft tissue with obstacles. In *Proceedings of the 2005 IEEE international conference on robotics and automation*, pages 1640–1645. IEEE, 2005.

[3] S. Bhattacharya, S. Kim, H. Heidarsson, G. S. Sukhatme, and V. Kumar. A topological approach to using cables to separate and manipulate sets of objects. *The International Journal of Robotics Research*, 34(6):799–815, 2015.

[4] T. Bretl and Z. McCarthy. Quasi-static manipulation of a kirchhoff elastic rod based on a geometric analysis of equilibrium configurations. *The International Journal of Robotics Research*, 33(1):48–68, 2014.

[5] S. Caron, A. Kheddar, and O. Tempier. Stair climbing stabilization of the hrp-4 humanoid robot using whole-body admittance control. In *International Conference on Robotics and Automation*, pages 277–283. IEEE, 2019.

[6] K. Chappellet, G. Caron, F. Kanehiro, K. Sakurada, and A. Kheddar. Benchmarking cameras for open vslam indoors. In *International Conference on Pattern Recognition*, pages 4857–4864. IEEE, 2021.

[7] S. Duenser, R. Poranne, B. Thomaszewski, and S. Coros. Robocut: hot-wire cutting with robot-controlled flexible rods. *ACM Transactions on Graphics (TOG)*, 39(4):98–1, 2020.

[8] S. Jin, W. Lian, C. Wang, M. Tomizuka, and S. Schaal. Robotic cable routing with spatial representation. *IEEE Robotics and Automation Letters*, 7(2):5687–5694, 2022.

[9] I. Kabul, R. Gayle, and M. C. Lin. Cable route planning in complex environments using constrained sampling. In *ACM symposium on Solid and physical modeling*, pages 395–402, 2007.

[10] L. E. Kavraki, F. Lamiraux, and C. Holleman. Towards planning for elastic objects. *Robotics: The Algorithmic Perspective*, pages 313–325, 1998.

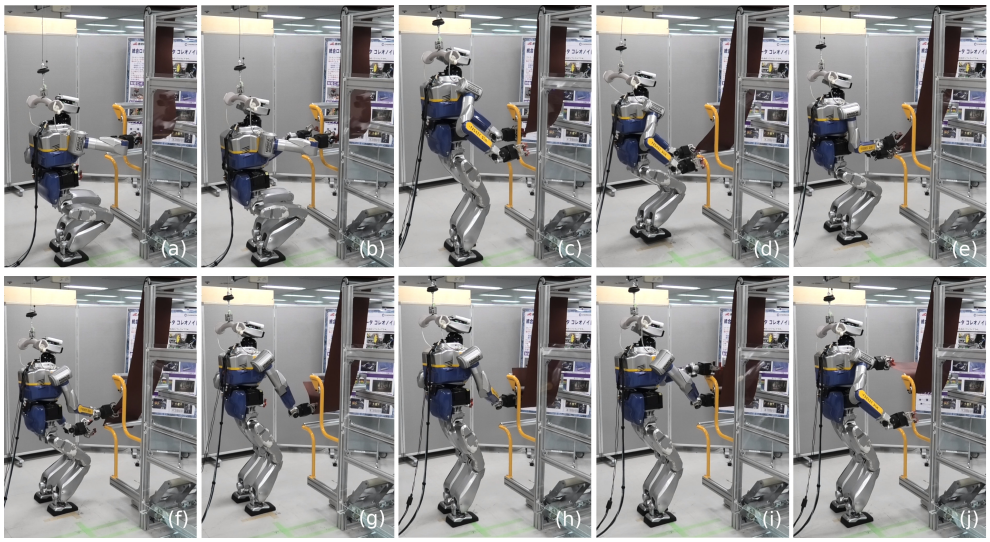


Fig. 10. Snapshots of assembling a belt on a mockup by humanoid robot HRP-2KAI. (a) According to the planning result in Fig. 9 (i), at the initial posture, we put the tail middle-part (at 17.6 cm) into the right gripper. (b) The left gripper regrasps the tail end-point of the belt. (c-d) The robot extends the belt and moves to the left. (e-f) The right gripper regrasps the middle-part of the belt. (g-h) The robot rotates the right gripper and inserts the belt to the small space between two rollers. (i-j) The left gripper regrasps the end-point and pulls the belt.

[11] S. Kim, S. Bhattacharya, and V. Kumar. Path planning for a tethered mobile robot. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1132–1139. IEEE, 2014.

[12] K. Kimble, J. Albrecht, M. Zimmerman, and J. Falco. Performance measures to benchmark the grasping, manipulation, and assembly of deformable objects typical to manufacturing applications. *Frontiers in Robotics and AI*, 9, 2022.

[13] M. Labbé and F. Michaud. Rtab-map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation. *Journal of Field Robotics*, 36(2):416–446, 2019.

[14] S. M. LaValle. *Planning algorithms*. Cambridge university press, 2006.

[15] P. Mitrano and D. Berenson. Data augmentation for manipulation. *arXiv preprint arXiv:2205.02886*, 2022.

[16] M. Murooka, K. Chappellet, A. Tanguy, M. Benallegue, I. Kumagai, M. Morisawa, F. Kanehiro, and A. Kheddar. Humanoid loco-manipulations pattern generation and stabilization control. *IEEE Robotics and Automation Letters*, 6(3):5597–5604, 2021.

[17] M. Murooka, I. Kumagai, M. Morisawa, F. Kanehiro, and A. Kheddar. Humanoid loco-manipulation planning based on graph search and reachability maps. *IEEE Robotics and Automation Letters*, 6(2):1840–1847, 2021.

[18] Y. Qin, A. Escande, and E. Yoshida. Cable installation by a humanoid integrating dual-arm manipulation and walking. In *International Symposium on System Integration*, pages 98–103. IEEE, 2019.

[19] O. Roussel, P. Fernbach, and M. Taïx. Motion planning for an elastic rod using contacts. *IEEE Transactions on Automation Science and Engineering*, 17(2):670–683, 2019.

[20] M. Saha and P. Isto. Manipulation planning for deformable linear objects. *IEEE Transactions on Robotics*, 23(6):1141–1150, 2007.

[21] D. Sánchez, W. Wan, and K. Harada. Tethered tool manipulation planning with cable maneuvering. *IEEE Robotics and Automation Letters*, 5(2):2777–2784, 2020.

[22] D. Sanchez, W. Wan, and K. Harada. Four-arm collaboration: Two dual-arm robots work together to manipulate tethered tools. *IEEE/ASME Transactions on Mechatronics*, 27(5):3286–3296, 2021.

[23] J.-P. Saut, M. Gharbi, J. Cortés, D. Sidobre, and T. Siméon. Planning pick-and-place tasks with two-hand regrasping. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4528–4533. IEEE, 2010.

[24] I. Shnaps and E. Rimon. Online coverage by a tethered autonomous mobile robot in planar unknown environments. *IEEE Transactions on Robotics*, 30(4):966–974, 2014.

[25] T. Siméon, J.-P. Laumond, J. Cortés, and A. Sahbani. Manipulation planning with probabilistic roadmaps. *The International Journal of Robotics Research*, 23(7-8):729–746, 2004.

[26] A. Sintov, S. Macenski, A. Borum, and T. Bretl. Motion planning for dual-arm manipulation of elastic rods. *IEEE Robotics and Automation Letters*, 5(4):6065–6072, 2020.

[27] C. Smith, Y. Karayiannidis, L. Nalpantidis, X. Gratal, P. Qi, D. V. Dimarogonas, and D. Kragic. Dual arm manipulation: A survey. *Robotics and Autonomous systems*, 60(10):1340–1353, 2012.

[28] Y. Sun, J. Falco, M. A. Roa, and B. Calli. Research challenges and progress in robotic grasping and manipulation competitions. *IEEE robotics and automation letters*, 7(2):874–881, 2021.

[29] H. G. Tanner, S. G. Loizou, and K. J. Kyriakopoulos. Nonholonomic navigation and control of cooperating mobile manipulators. *IEEE Transactions on robotics and automation*, 19(1):53–64, 2003.

[30] N. Vahrenkamp, T. Asfour, and R. Dillmann. Robot placement based on reachability inversion. In *2013 IEEE International Conference on Robotics and Automation*, pages 1970–1975. IEEE, 2013.

[31] N. Vahrenkamp, D. Berenson, T. Asfour, J. Kuffner, and R. Dillmann. Humanoid motion planning for dual-arm manipulation and re-grasping tasks. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2464–2470. IEEE, 2009.

[32] J. Vaillant, A. Kheddar, H. Audren, F. Keith, S. Brossette, A. Escande, K. Bouyarmane, K. Kaneko, M. Morisawa, P. Gergondet, et al. Multi-contact vertical ladder climbing with an hrp-2 humanoid. *Autonomous Robots*, 40(3):561–580, 2016.

[33] G. A. Waltersson, R. Laezza, and Y. Karayiannidis. Planning and control for cable-routing with dual-arm robot. In *International Conference on Robotics and Automation*, pages 1046–1052, 2022.

[34] W. Wang, D. Berenson, and D. Balkcom. An online method for tight-tolerance insertion tasks for string and rope. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2488–2495. IEEE, 2015.

[35] H. Yin, A. Varava, and D. Kragic. Modeling, learning, perception, and control methods for deformable object manipulation. *Science Robotics*, 6(54):eabd8803, 2021.

[36] M. Yu, K. Lv, H. Zhong, S. Song, and X. Li. Global model learning for large deformation control of elastic deformable linear objects: An efficient and adaptive approach. *IEEE Transactions on Robotics*, 2022.

[37] J. Zhu, B. Navarro, P. Fraisse, A. Crosnier, and A. Cherubini. Dual-arm robotic manipulation of flexible cables. In *International Conference on Intelligent Robots and Systems*, pages 479–484, 2018.

[38] J. Zhu, B. Navarro, R. Passama, P. Fraisse, A. Crosnier, and A. Cherubini. Robotic manipulation planning for shaping deformable linear objects with environmental contacts. *IEEE Robotics and Automation Letters*, 5(1):16–23, 2019.