A Coarse-to-Fine Framework for Dual-Arm Manipulation of Deformable Linear Objects with Whole-Body Obstacle Avoidance

Mingrui Yu, Kangchen Lv, Changhao Wang, Masayoshi Tomizuka, and Xiang Li

Abstract— **Manipulating deformable linear objects (DLOs) to achieve desired shapes in constrained environments with obstacles is a meaningful but challenging task. Global planning is necessary; however, accurate models of DLOs required by planners are difficult to obtain, and the inevitable modeling errors probably result in task failure if the robot simply open-loop executes the planned path. In this abstract, we propose a coarse-to-fine framework to combine global planning and local control for dual-arm manipulation of DLOs, capable of precisely achieving desired configurations and avoiding potential collisions between the DLO, robot, and obstacles. Both simulations and real-world experiments demonstrate that our framework can robustly achieve desired DLO configurations in constrained environments with imprecise DLO models, which may not be reliably achieved by only planning or control.**

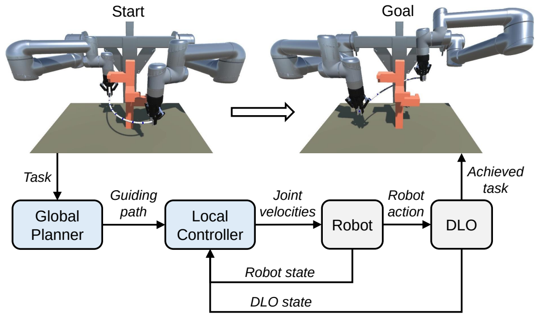


Fig. 1. Overview of the proposed framework for dual-arm manipulation of DLOs in constrained environments with obstacles, in which the global planning and local control complement each other to robustly achieve complex tasks even using imprecise DLO models. The collision avoidance of both the DLO and robot body is considered in the planning and execution.

I. Introduction

Deformable linear objects (DLOs) are common in human life [1]; however, there are new challenges coming from the deformable properties of DLOs when applying classical manipulation methods [2], [3]. We focus on a general problem of DLO manipulation: dual-arm manipulating a DLO from a start configuration to a desired (goal) configuration. In our previous work [4], [5], we have proposed an adaptive controller which can achieve DLO shape control in unobstructed environments. In this work, we further deal with a more complex and practical scenario: manipulating DLOs in constrained environments with obstacles (see Fig. 1).

A series of works have studied the DLO shaping from the perspective of control [6]–[10]. However, these control-only methods usually consider a simplified scenario: no obstacles exist, and the robot end-effectors can move freely without considering the arm bodies; moreover, local controllers cannot prevent DLOs from falling into local optimal shapes.

Global path planning is necessary for manipulating DLOs in constrained environments. Existing studies have proposed different DLO models and incorporated them into classical high-dimensional path planning methods [11]–[13]. Paths of DLO configurations from the start to the goal are planned, and corresponding end-effector trajectories are extracted for open-loop executions [14]–[16]. However, the planning-only methods are more affected by the inevitable DLO modeling errors than control, since no real-time feedback compensates for the errors. The planned end-effector trajectory may not move the DLO exactly along the expected configuration path, which may cause failures of the actual execution. As a result, most of these approaches are restricted to simulations with sufficiently accurate models.

To overcome the shortcomings of the existing methods, we propose a new framework that combines both global planning and local control, with the structure shown in Fig. 1. The planner efficiently computes a coarse and global collision-free path to the desired configuration, based on a simple yet effective DLO model; Then, the controller tracks the planned path as guidance while using closed-loop feedback to compensate for the modeling errors in the planning phase and locally avoid obstacles. Specifically, for the planner, we use an RRT planning framework with a DLO energy model for projecting a random DLO configuration to a stable one; for the controller, we formulate it as a model predictive control (MPC) problem with artificial potentials. The full paper and video are available on the project website[1].

McConachie et al. [17] proposed another framework for interleaving planning and control for manipulating deformable objects. Their controller first attempts to perform tasks directly; if their deadlock predictor predicts the controller will get stuck, their planner will be invoked to move the object to a new region. Note that their aim and combination approach is different from ours: first, their planner uses a more simplified model called “virtual elastic band" focusing on the overstretch of deformable objects caused by grippers or obstacles; second, their planned path of grippers is open-loop executed without local adjustment using real-time feedback; as a result, the DLO may be over-compressed or hooked by obstacles. In contrast, our method considers the full DLO configurations both in planning and control, and employs feedback control during the whole manipulation process.

Notations in this abstract: The DLO configuration is

M. Yu, K. Lv, and X. Li are with the Department of Automation, Tsinghua University, Beijing, China. C. Wang and M. Tomizuka are with the Department of Mechanical Engineering, University of California, Berkeley, CA, USA. Corresponding author: Xiang Li (xiangli@tsinghua.edu.cn)

[1] Project website: https://mingrui-yu.github.io/DLO_planning

represented by multiple *feature points* along the DLO. The position of the k^{th} feature is represented as $\boldsymbol{x}_k \in \Re^3$. The configuration of the DLO is represented as $\boldsymbol{x} := [\boldsymbol{x}_1; \cdots; \boldsymbol{x}_m] \in \Re^{3m}$, where m is the number of features. The joint position of the dual-arm robot is represented as $\boldsymbol{q} \in \Re^n$, where n is the degrees of freedom (DoFs) of the robot. The configuration vector of the robot end-effectors is represented as $\boldsymbol{r} \in \Re^{12}$. The control input $\boldsymbol{u}$ to the system is the joint velocity of the dual-arm robot, as $\dot{\boldsymbol{q}} = \boldsymbol{u}$.

II. Global Planning

The planner is first invoked to efficiently find a coarse global collision-free path in constrained environments using a simple yet effective DLO energy model.

A. Projection onto Stable DLO Configuration Manifold

The stable equilibrium configurations of a DLO are on a lower-dimensional manifold of the raw configuration space $\Re^{3m}$. As direct sampling from the manifold is unlikely, we use projection methods to move a random sample in the raw space onto its neighbor manifold [18]. Denote the potential energy of an elastic DLO as E, which is assumed to be fully determined by $\boldsymbol{x}$. A stable equilibrium of the DLO with two end poses fixed is where the DLO's internal configuration locally minimizes the potential energy E [11], [12]. Thus, a random sample $\boldsymbol{x}^0$ can be projected onto the manifold by formulating it as a local minimization problem of the energy with $\boldsymbol{x}^0$ as the initial value:

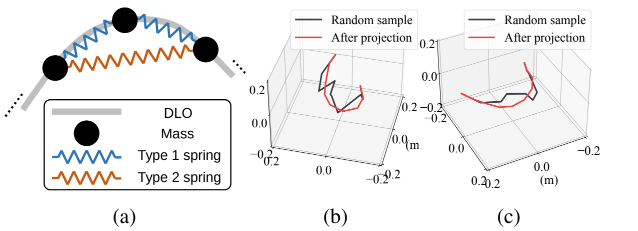
$$\begin{aligned} \boldsymbol{x}^{\text{stable}} = \arg\min_{\boldsymbol{x}} \quad & E(\boldsymbol{x}) \\ \text{s.t.} \quad & \boldsymbol{x}_k = \boldsymbol{x}_k^0, \quad \forall k = 1, 2, m-1, m \end{aligned} \tag{1}$$

where the first two and last two feature points are fixed to represent the fixed end poses in discrete DLO models [19]. We denote such a process as $\boldsymbol{x}^{\text{stable}} = \text{ProjectStableConfig}(\boldsymbol{x}^0)$.

In this work, we use a simple mass-spring model [2] as the energy model (Fig. 2(a)). In our experiments, we use the same stiffness values in all the planning. We visualize two examples of the ProjectStableConfig in Fig. 2(b) and 2(c).

B. Planning Algorithm

Our planning algorithm uses the same framework as the Constrained Bi-directional Rapidly-Exploring Random Tree (CBiRRT) method in [18]. Each node $\mathcal{N}$ in the trees contains both the DLO configuration $\boldsymbol{x}$ and the robot configuration $\boldsymbol{q}$. The key modification is in the function $\mathcal{N}_{\text{reached}} = \text{ConstrainedExtend}(\mathcal{N}_{\text{from}}, \mathcal{N}_{\text{to}})$ in the CBiRRT. In each step in the ConstrainedExtend, a new DLO configuration is calculated by interpolating from the last reached configuration $\boldsymbol{x}_{\text{last}}$ to $\boldsymbol{x}_{\text{to}}$ with a small step size, where we use linear interpolation for the centroid positions and spherical interpolation for the relative deformations irrelevant to translations to keep the overall DLO shapes and avoid overcompression [12]. The ProjectStableConfig is then applied to project the interpolated configuration to a stable $\boldsymbol{x}_{\text{new}}$. The corresponding robot configuration $\boldsymbol{q}_{\text{new}}$ is obtained as the inverse kinematics (IK) solution for the end poses of $\boldsymbol{x}_{\text{new}}$. Finally, the successfully planned path is denoted as $\mathcal{P} = \{\mathcal{N}_{start}, \mathcal{N}_1, \cdots, \mathcal{N}_{goal}\}$ and sent to the local controller as a guiding global path.



(a) (b) (c)

Fig. 2. (a) Illustration of the mass-spring model which models a DLO as a series of masses connected by springs. The i^{th} mass is connected to $(i\pm1)^{\text{th}}$ masses by Type 1 springs and $(i\pm2)^{\text{th}}$ masses by Type 2 springs. (b)(c) Illustrations of projecting a randomly sampled raw DLO configuration to a stable configuration by locally minimizing the potential energy of the mass-spring model as (1).

III. Manipulation with Local Control

We apply an MPC to track the planned path as guidance while using real-time feedback to adjust the robot motion.

A. Manipulation Process

Each node of the planned guiding path is regarded as an intermediate desired configuration. If the distance between the current and desired configuration is less than a threshold, the control objective will be switched to the next planned configuration. Besides, if the controller gets stuck, the global planner will be invoked again to re-plan a path from the current configuration to the final desired configuration.

B. Controlling to an Intermediate Configuration

The control input is specified as the locally optimal solution of the following optimization problem:

$$\begin{aligned} \min_{\boldsymbol{u}} \quad & \mathcal{J} = \lambda_a^{\text{dlo}} U_a^{\text{dlo}}(\boldsymbol{x}(t+\delta t)) + \lambda_r^{\text{dlo}} U_r^{\text{dlo}}(\boldsymbol{x}(t+\delta t)) \\ & \quad + \lambda_a^{\text{arm}} U_a^{\text{arm}}(\boldsymbol{q}(t+\delta t)) + \lambda_r^{\text{arm}} U_r^{\text{arm}}(\boldsymbol{q}(t+\delta t)) \\ & \quad + \frac{1}{2}\boldsymbol{u}^{\mathsf{T}}\boldsymbol{K}_u\boldsymbol{u} \\ \text{s.t.} \quad & \boldsymbol{C}_{\text{dof}}\boldsymbol{J}^{\text{arm}}\boldsymbol{u} = \boldsymbol{0} \\ & \boldsymbol{u}^{\mathsf{T}}\boldsymbol{K}_u\boldsymbol{u} \leq u_{\max}^2 \end{aligned} \tag{2}$$

where $\boldsymbol{u}$ is the control input, U_a^{dlo} and U_a^{arm} are the attractive potentials for driving the DLO to $\mathcal{N}_i.\boldsymbol{x}$ and the arms to $\mathcal{N}_i.\boldsymbol{q}$, and U_r^{dlo} and U_r^{arm} are the designed repulsive potentials to achieve local obstacle avoidance. The $\lambda_a^{\text{dlo}}, \lambda_r^{\text{dlo}}, \lambda_a^{\text{arm}}, \lambda_r^{\text{arm}}$ are weighting coefficients, and δt is the step interval. The $\boldsymbol{K}_u$ is a weighting matrix for the input joint velocities. The first equality constraint is to constrain the allowed DoFs of the robot end-effectors for specific tasks, where $\boldsymbol{J}^{\text{arm}}$ is the robot Jacobian matrix. We employ the method in [4] to predict the DLO configuration at $t + \delta t$. When controlling to the final desired configuration, the λ_a^{arm} is set to zero.

IV. Results

A. Simulations

1) Overall Performance: We first test the overall performance of our method in three different tasks with different

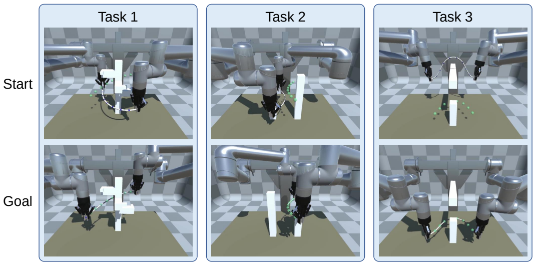


Fig. 3. Three simulated tasks for testing the overall performance of the proposed method. The top row is the start configurations and the bottom row is the goal configurations. The pictures are taken from actual manipulations.

TABLE I
PERFORMANCE OF OUR METHOD IN THE THREE SIMULATED TASKS.

Task	Planning success rate	Planning time (s)[a]	Manipulation success rate	Final task error (cm)[b]
1	8/10	3.29 ± 1.45	8/8	0.55 ± 0.22
2	8/10	4.14 ± 4.31	8/8	0.55 ± 0.11
3	10/10	2.66 ± 1.25	10/10	0.60 ± 0.18

[a,b] The mean value ± standard deviation over the successful cases.

scenarios, DLO properties, and start/goal configuration, as shown in Fig 3. The lengths of the DLOs are 0.5/0.4/0.6m for Task 1/2/3, respectively. For each task, the planner tries 10 queries and all found paths are executed by the controller. The performance is summarized in Table I.

2) Closed-Loop v.s. Open-Loop: We verify the significance of the control module by comparing the manipulation results of executing the planned path in an open-loop manner and in a closed-loop manner using the proposed controller.

The quantitative tests use the same two scenarios as the two cases in Fig. 5, and the DLO lengths are 0.4/0.6m. Five different desired shapes are tested for each scenario, and five paths are planned for each desired shape. Fig. 4 shows the results of the collision time during manipulations and the final task errors. We also visualize the manipulation processes of two specific cases in Fig. 5. As for the open-loop executions, in the case shown in Fig. 5(a), the DLO gradually deviates from the planned DLO path and finally terminates at another stable but undesired shape; in the case shown in Fig 5(b), unexpected collisions between the DLO and obstacles happen and the DLO is hooked. The closed-loop executions for both two cases are successful.

B. Real-World Experiments

Three representative tasks are achieved using the proposed method, as shown in Fig. 6. In Task 1, the DLO needs to be rotated around 180 degrees; in Task 2, the planner in [17] may fail, since the "virtual elastic band" connecting the two end-effectors will not collide with the obstacle along a straight-line robot path, but the actual DLO body will; and in Task 3, the DLO needs to be manipulated from an upper-semicircle shape to a lower-semicircle shape. A 0.3m-long nylon rope, a 0.6m-long HDMI cable, and a 0.45m-long electric wire are used in Task 1, 2, and 3, respectively. For Task 3, we also show the comparison between the proposed closed-loop manner and the open-loop manner using the same planned path.

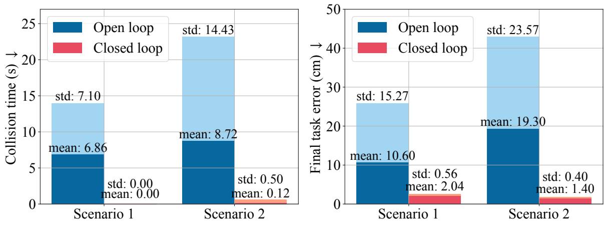
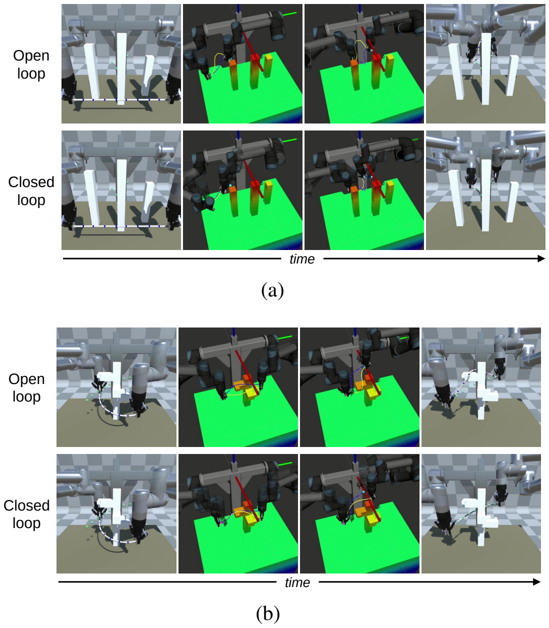


Fig. 4. Quantitative results of the comparison between the proposed closed-loop framework and the open-loop manner. The bars with darker colors refer to the mean values over all 25 manipulations, and those with lighter colors refer to the standard deviations (std).



(a)

(b)

Fig. 5. Two cases for illustrating the comparison between the open-loop manner and proposed closed-loop manner using the same planned paths in Scenario 1 and 2. The first column is the start configurations; the second and third columns are the manipulation processes, where the blue lines and translucent robots refer to the planned intermediate configurations and the yellow lines and non-translucent robots refer to the real-time configurations; and the last column is the final reached configurations, where the translucent green points refer to the final desired DLO configurations.

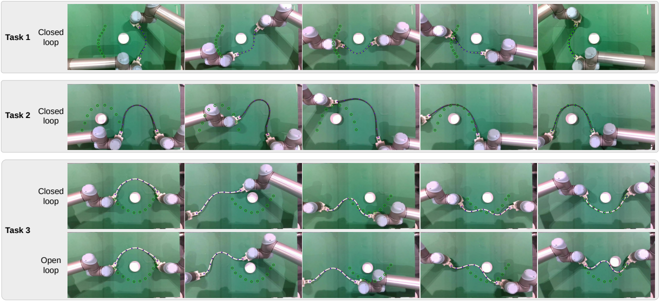


Fig. 6. Manipulation processes of the three real-world tasks. The green+black circles refer to the final desired DLO configurations.

V. CONCLUSION

This abstract proposes a coarse-to-fine framework for closed-loop manipulating DLOs with whole-body planning and control, which considers both the feasibility and accuracy of long-horizon DLO manipulation tasks in constrained environments. Please refer to the full paper [20] for details.

References

[1] J. Sanchez, J.-A. Corrales, B.-C. Bouzgarrou, and Y. Mezouar, "Robotic manipulation and sensing of deformable objects in domestic and industrial applications: a survey," *The International Journal of Robotics Research*, vol. 37, no. 7, pp. 688–716, 2018.

[2] H. Yin, A. Varava, and D. Kragic, "Modeling, learning, perception, and control methods for deformable object manipulation," *Science Robotics*, vol. 6, no. 54, 2021.

[3] J. Zhu, A. Cherubini, C. Dune, D. Navarro-Alarcon, F. Alambeigi, D. Berenson, F. Ficuciello, K. Harada, J. Kober, X. Li *et al.*, "Challenges and outlook in robotic manipulation of deformable objects," *IEEE Robotics and Automation Magazine*, 2021.

[4] 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*, vol. 39, no. 1, pp. 417–436, 2023.

[5] M. Yu, H. Zhong, and X. Li, "Shape control of deformable linear objects with offline and online learning of local linear deformation models," in *2022 IEEE International Conference on Robotics and Automation (ICRA)*, 2022, pp. 1337–1343.

[6] D. Navarro-Alarcon, H. M. Yip, Z. Wang, Y. Liu, F. Zhong, T. Zhang, and P. Li, "Automatic 3-d manipulation of soft objects by robotic arms with an adaptive deformation model," *IEEE Transactions on Robotics*, vol. 32, no. 2, pp. 429–441, 2016.

[7] C. Wang, Y. Zhang, X. Zhang, Z. Wu, X. Zhu, S. Jin, T. Tang, and M. Tomizuka, "Offline-online learning of deformation model for cable manipulation with graph neural networks," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5544–5551, 2022.

[8] J. Zhu, D. Navarro-Alarcon, R. Passama, and A. Cherubini, "Vision-based manipulation of deformable and rigid objects using subspace projections of 2d contours," *Robotics and Autonomous Systems*, vol. 142, p. 103798, 2021.

[9] R. Lagneau, A. Krupa, and M. Marchal, "Automatic shape control of deformable wires based on model-free visual servoing," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5252–5259, 2020.

[10] S. Jin, C. Wang, and M. Tomizuka, "Robust deformation model approximation for robotic cable manipulation," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 6586–6593.

[11] H. Wakamatsu and S. Hirai, "Static modeling of linear object deformation based on differential geometry," *The International Journal of Robotics Research*, vol. 23, no. 3, pp. 293–311, 2004.

[12] M. Moll and L. Kavraki, "Path planning for deformable linear objects," *IEEE Transactions on Robotics*, vol. 22, no. 4, pp. 625–636, 2006.

[13] 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*, vol. 33, no. 1, pp. 48–68, 2014.

[14] O. Roussel, P. Fernbach, and M. Taïx, "Motion planning for an elastic rod using contacts," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 2, pp. 670–683, 2020.

[15] A. Sintov, S. Macenski, A. Borum, and T. Bretl, "Motion planning for dual-arm manipulation of elastic rods," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6065–6072, 2020.

[16] P. Mitrano, D. McConachie, and D. Berenson, "Learning where to trust unreliable models in an unstructured world for deformable object manipulation," *Science Robotics*, vol. 6, no. 54, p. eabd8170, 2021.

[17] D. McConachie, A. Dobson, M. Ruan, and D. Berenson, "Manipulating deformable objects by interleaving prediction, planning, and control," *The International Journal of Robotics Research*, vol. 39, no. 8, pp. 957–982, 2020.

[18] D. Berenson, S. S. Srinivasa, D. Ferguson, and J. J. Kuffner, "Manipulation planning on constraint manifolds," in *2009 IEEE International Conference on Robotics and Automation (ICRA)*, 2009, pp. 625–632.

[19] M. Bergou, M. Wardetzky, S. Robinson, B. Audoly, and E. Grinspun, "Discrete elastic rods," in *ACM SIGGRAPH 2008 Papers*, New York, NY, USA, 2008.

[20] M. Yu, K. Lv, C. Wang, M. Tomizuka, and X. Li, "A coarse-to-fine framework for dual-arm manipulation of deformable linear objects with whole-body obstacle avoidance," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023.