2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Adaptive Curvature-Aware Routing for Stiff Cable Control via dual manipulation

Jiahao Long, Yang Cong and Yu Ren

Abstract— **Deformable Linear Objects (DLOs), such as cables and ropes, pose significant challenges for robotic manipulation due to their high-dimensional state space, nonlinear deformation dynamics, and strong sensitivity to external forces. Cable routing tasks, in particular, are further complicated by geometric constraints, residual stresses in stiff cables, and the necessity of precise alignment with designated connectors. Existing approaches often rely on endpoint manipulation or external fixtures, which limits flexibility and scalability in real-world applications. While data-driven and graph-based models have shown promise for flexible ropes, they struggle to generalize across varying cable stiffness and suffers high computational costs. To address these challenges, we propose Adaptive Curvature-Aware Routing (ACR), a dual manipulation framework capable of adaptively handling cables of high stiffness and arbitrary lengths. Specifically, our framework combines local curvature analysis with Radial Basis Function Networks (RBFNs) to predict cable deformations. By prioritizing regions with high curvature discrepancies, it adaptively selects manipulation segments and performs safe, precise corrective actions to shape the cable toward the target configuration without heavy reliance on fixtures. Furthermore, we develop a constraint-aware cooperative controller that integrates both kinematic feasibility and physical safety into the motion strategy. Experiments in both simulation and real-world setups demonstrate that ACR significantly outperforms baseline methods in terms of success rate and terminal accuracy, validating the effectiveness of combining curvature-based adaptivity with data-driven modeling for complex cable routing tasks.**

I. INTRODUCTION

Deformable Linear Objects (DLOs), such as cables, ropes, and elastic rods, are widely encountered in industrial and daily-life applications. Examples range from robotic cable harness assembly [1] and surgical suturing [2] to household service robotics [3], [4]. Compared with the manipulation of rigid objects, the control of DLOs is inherently more complex due to their high-dimensional configuration space, nonlinear deformation dynamics, and strong sensitivity to external forces [5], [6]. For robotic systems, achieving reliable and precise DLO manipulation requires not only accurate grasping and motion planning, but also reasoning about the continuous deformation that occurs throughout the interaction. These challenges make DLO manipulation a fundamental yet difficult problem with high relevance for both research and practice.

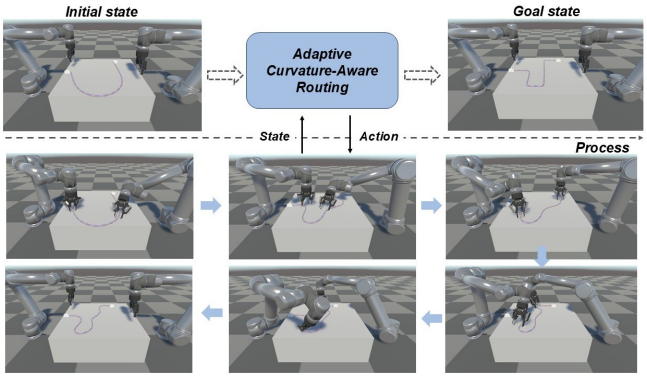


Fig. 1: Problem setting: In practical cable routing scenarios, precise alignment with designated connectors is often required. To simplify fixture perception, we consider the case where both cable ends are fixed prior to routing. The objective is then to guide the cable into the desired shape through cooperative dual manipulation.

In this paper, we address the challenging task of robotic cable routing, where a robot must manipulate a cable from an initial configuration to a target shape along a predefined path while respecting geometric and environmental constraints [1], [7]. The task becomes particularly difficult when dealing with high-stiffness cables [8], [9]. Unlike highly flexible ropes that can be easily draped or guided [10], [11], [12], stiff cables resist deformation and retain residual stresses after manipulation, making precise shape control challenging. As a result, direct manipulation by grasping the cable ends is often insufficient for achieving the desired fine-grained deformation. Existing methods frequently rely on external fixtures, such as clamps or holders, to guide the cable along the target path. While effective, this approach imposes additional constraints on task design and environmental setup [13], [14], [15]. Consequently, developing robotic methods that can accurately regulate the shape of stiff cables without heavy reliance on external fixtures remains an open and practically significant challenge.

To address this problem, we propose a dual manipulation framework for cable routing. The framework is shown in Fig. 2. Our method begins with real-time perception of the cable, which represents the DLO as a sequence of discrete points. By comparing the current and target DLO configurations, we perform curvature-difference analysis to identify segments with large deviations. These critical regions are

*The corresponding author is Prof. Yang Cong. The work was supported in part by National Key R&D Program of China under Grant 2023YFB4704800, NSFC under Grant 62225310, and was supported by Guangdong S&T Program (2025B1111130001).

*Jiahao Long, Yang Cong and Yu Ren are with the College of Automation Science and Engineering, South China University of Technology, Guangzhou, China. (email: 202420115946@mail.scut.edu.cn, congyang81@gmail.com, renyu0414@gmail.com).

prioritized for manipulation. For grasping, we formulate an optimization strategy that computes gripper poses aligned with the local cable geometry while satisfying kinematic feasibility and safety constraints. To model the cable's deformation response, we introduce a curvature prediction model based on a two-layer Radial Basis Function Network (RBFN). Finally, we design a cooperative controller that integrates the learned dynamics with task-level constraints. This controller enables segment-wise correction of the cable, ensuring stable convergence to the target configuration.

Experimental validation was conducted in both simulation and real-world environments. Simulation tests involved three routing tasks—S-curve, O-curve, and U-curve—with randomized connector placements to evaluate generalization. Compared to baseline strategies, our method consistently achieved higher success rates and lower terminal errors, albeit with increased computation time in complex scenarios. Real-world experiments, presented in the supplementary materials, further confirm the applicability and robustness of the proposed approach. The main contributions of this work are as follows:

- A dual manipulation framework for routing long, stiff cables that integrates real-time perception with curvature-based deformation analysis;
- A curvature prediction model using a two-layer RBFN capable of predicting local deformation responses under dual manipulation;
- A constraint-aware cooperative controller that ensures both safety and precision by incorporating physical and kinematic constraints;

II. RELATED WORK

A. DLO Shape Control

Early approaches for precise DLO shape control relied on visual feedback and demonstration learning, focusing on highly flexible ropes. Nair et al. [10] demonstrated self-supervised rope manipulation from visual inputs, highlighting the potential of data-driven policies but with limited applicability to stiffer cables. Jin et al. [16] proposed robust model approximation to improve control under noisy and partial observations, though at the cost of computational complexity for long-scale coupling. Lee et al. [11] showed that sample-efficient self-supervised methods can scale to real-world DLO tasks. More recently, graph-based dynamics learning has gained attention. Wang et al. [8] and Gu et al. [12] employed graph neural networks for global dynamics modeling, achieving strong expressiveness but requiring significant data and training effort. To balance efficiency and accuracy, Yu et al. [9], [17] combined offline and online local linear models, enabling rapid adaptation to critical regions. Existing methods for DLO shape control demonstrate strengths in data-driven and graph-based approaches but face persistent limitations in computational efficiency and generalization to stiff cables with elastoplastic properties. While techniques such as local linear models enhance adaptability, they often lack integration with real-time curvature analysis.

Our method mitigates these limitations by prioritizing manipulation of critical segments with significant shape deviations, thereby reducing computational overhead. The curvature prediction model employs localized representations instead of complex global dynamics learning, enabling efficient control across cables spanning a broad stiffness range.

B. DLO Dual Manipulation

Dual manipulation offers clear advantages for long DLOs, such as maintaining tension, segment-wise routing, and threading through fixtures. Tang et al. [18] introduced a coherent point drift framework for alignment, effective for paired positioning. Sintov et al. [19] incorporated elastic rod physics into dual-arm planning, improving feasibility under physical constraints. Yu et al. [20], [21] proposed a coarse-to-fine dual-arm framework with integrated obstacle avoidance, highlighting the importance of hierarchical strategies for complex assembly. Qin et al. [1] extended dual manipulation to mobile platforms in industrial installation scenarios, demonstrating feasibility in constrained workspaces. For cable routing, Jin et al. [7] developed a spatial representation to support wiring tasks. Overall, dual manipulation frameworks significantly advance DLO tasks by enabling tension control and obstacle navigation, yet prior research has not fully optimized grasp point selection based on local geometric features like curvature.

C. Perception of DLO

Accurate DLO perception is a prerequisite for shape control. Early work focused on visual detection and segmentation. Caporali et al. [22] introduced FastDLO for real-time detection, achieving high speed but limited 3D accuracy under occlusion. Yan et al. [23] proposed self-supervised estimation methods to improve robustness in closed-loop control. To address partial observations, Lv et al. [24] presented single-frame point cloud estimation, offering higher accuracy under occlusions. Xiang et al. [25] extended this to temporal consistency with TrackDLO, improving robustness under occlusions. In conclusion, perception techniques for DLOs have evolved from basic segmentation to robust models [26] handling occlusions, exemplifying advances in temporal consistency. Our framework leverages TrackDLO for real-time keypoint extraction but extends its utility through curvature-difference analysis, which directly informs grasp optimization and control decisions.

III. METHOD

In this section, we introduce our proposed framework for dual-arm cable manipulation, which integrates cable perception, curvature-based difference analysis, a dual grasping strategy, a deformation prediction model, and a cooperative control scheme. The overall pipeline is illustrated in Fig. 2. The method is designed to ensure stable grasping, accurate curvature regulation, and safe dual manipulation while operating on deformable cables in planar wiring tasks.

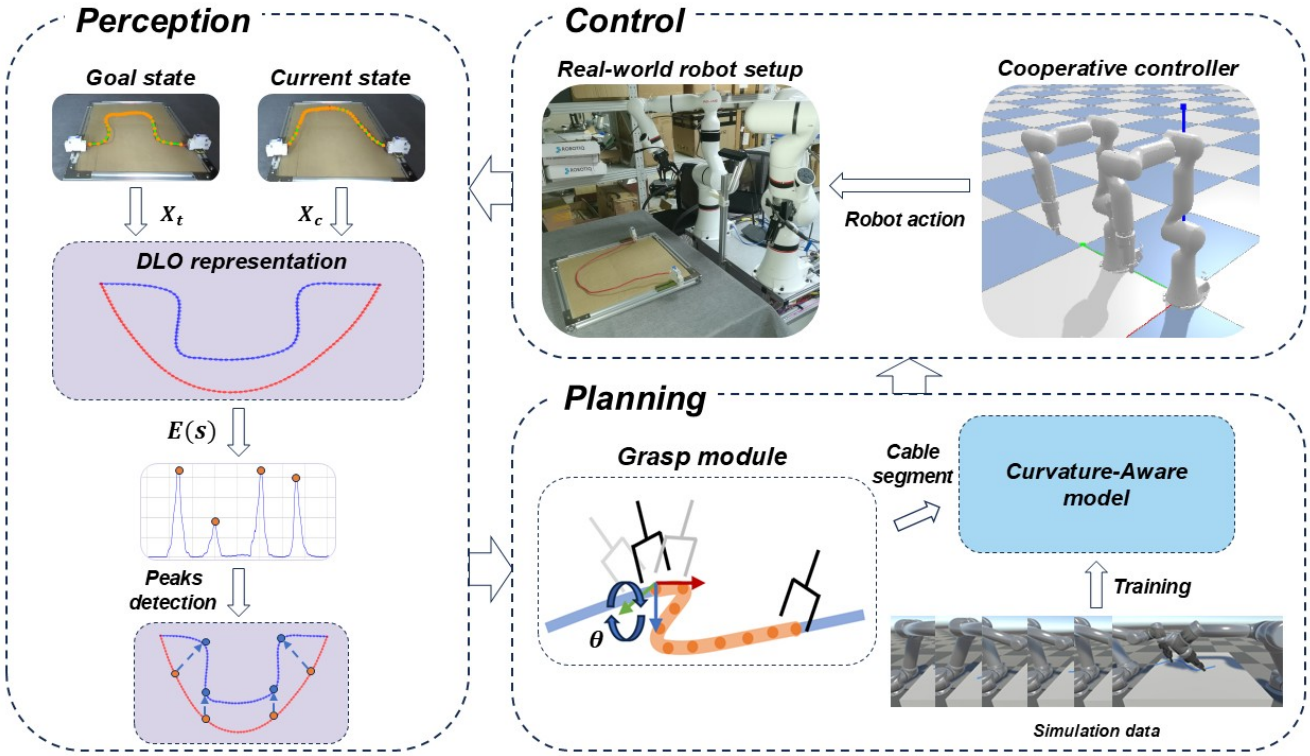


Fig. 2: Proposed framework for dual-arm cable manipulation. The system consists of four sequentially executed modules: (a) curvature discrepancy analysis identifies critical segments requiring manipulation by comparing current and target configurations; (b) grasping strategy optimization computes gripper poses aligned with local cable geometry; (c) deformation prediction employs a two-layer Radial Basis Function Network (RBFN) to forecast curvature changes under manipulation; and (d) cooperative controller integrates physical constraints and learned dynamics to execute corrective motions. This pipeline ensures stable grasping, accurate curvature regulation, and safe dual manipulation.

A. Problem Formulation

We consider a planar cable wiring and assembly task, as illustrated in Fig. 1. Specifically, the two ends of the cable are anchored to fixed terminals, while the dual robot is responsible for routing the cable along predefined paths. The cable state is discretized as a sequence of keypoints, which can be obtained from real-time perception. The manipulation problem can be described as steering the current keypoint configuration $X_c = \{x_i^c\}_{i=1}^N \in \mathbb{R}^{2N}$ towards the target configuration $X_t = \{x_i^t\}_{i=1}^N \in \mathbb{R}^{2N}$, where N denotes the number of keypoints.

We adopt the TrackDLO [25] perception method to extract cable keypoints from RGB-D input. The target state is provided either from design specifications or demonstration.

B. Curvature Discrepancy Analysis

A central challenge in cable manipulation is quantifying the difference between the observed and target configurations. Since cables are flexible and continuous, position-based differences alone are insufficient. Instead, we employ curvature-based analysis, which directly captures local deformation properties.

Given both observed and target states, we fit B-spline curves to the keypoints. The bending energy [27] of a cable in the 2D plane is expressed as

$$E = \frac{\lambda_b}{2} \int \kappa(s)^2 \, ds. \tag{1}$$

where $\kappa(s)$ is the curvature at arc-length s, and λ_b is the bending stiffness. This formulation originates from elastic rod theory and represents the elastic potential energy stored due to bending.

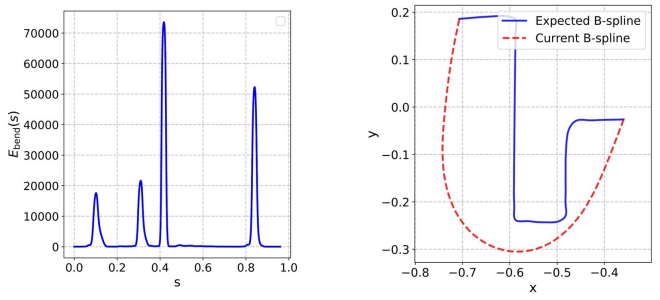
We define the discrepancy signal as

$$E(s) = \frac{\lambda_b}{2} \int_{\Delta s} \left[\kappa_c(s)^2 - \kappa_t(s)^2 \right] ds. \tag{2}$$

Physically, $E(s)$ reflects the energy mismatch between the observed and target cables. Peaks in $E(s)$ correspond to locations where curvature deviates most significantly. Consequently, they highlight regions requiring corrective manipulation. By decomposing the cable into segments defined by these peaks, the robot can apply localized operations rather than attempting to reshape the entire cable at once. This segmentation improves both control efficiency and precision. An example of this analysis is illustrated in Fig. 3.

C. Curvature-Aligned Grasp Pose Optimization

Once the target segment is identified, the robot must determine a stable grasp configuration. Grasp stability is critical because misalignment between the gripper orientation and



(a) Energy discrepancy signal (b) B-spline curve comparison

Fig. 3: Curvature discrepancy analysis: The energy discrepancy signal $E(s)$ (left) reveals segments with significant deviation between (right) target DLO configuration (blue) and current DLO configuration (red).

the local cable tangent may induce slippage or uncontrolled deformation. To mitigate this risk, we align the gripper orientation with the Frenet frame of the cable curve.

For a point at arc-length s, the Frenet frame $\{\mathbf{t}(s),\mathbf{n}(s),\mathbf{b}(s)\}$ defines its tangent, normal, and binormal directions. This alignment reduces the grasp degrees of freedom to two parameters: the grasping location and an in-plane rotation. The grasp action is expressed as

$$\mathbf{a}(s,\theta) = \big(\mathbf{p}(s), R(\mathbf{n}(s),\theta)\big), \tag{3}$$

where $\mathbf{p}(s)$ is the position on the target segment of cable and θ is the rotation angle about $\mathbf{n}(s)$.

The dual-arm grasp optimization is then formulated to simultaneously minimize segment length and maximize inter-arm safety:

$$\min_{\mathbf{a}_l,\mathbf{a}_r} J_g = -\omega_1 \xi(a_l,a_r) + \omega_2 \sigma(a_l,a_r), \tag{4}$$

subject to

$$\xi(a_l,a_r) \geq d. \tag{5}$$

Here, $\xi(\cdot)$ computes the minimum distance between the robots (ensuring collision avoidance), while $\sigma(\cdot)$ measures the grasped cable segment length. Shorter segments improve control precision but restrict robot mobility. To resolve this trade-off, we employ the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) as reviewed in [28], a stochastic optimizer robust to high-dimensional non-convex spaces. Unlike gradient-based methods, CMA-ES avoids local minima and handles discontinuous cost functions effectively, making it ideal for deformable object manipulation.

D. Curvature Prediction Model

We design a two-layer Radial Basis Function Network (RBFN) framework that predicts curvature evolution under dual manipulation, with emphasis on velocity-induced curvature changes. As shown in Fig. 4, the model operates through two cascaded stages.

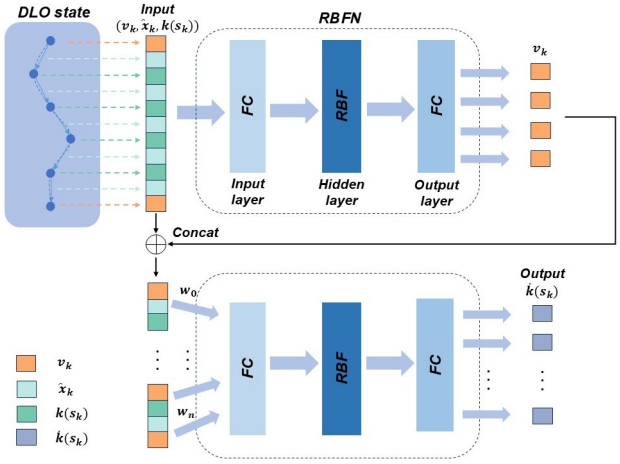


Fig. 4: Two-layer RBFN framework for curvature prediction. The first layer predicts intermediate node velocities, while the second layer uses velocity and tangent features to estimate $\dot{\kappa}$. Nodes near curvature peaks receive higher weights to improve accuracy.

1) Network Architecture: The first-layer RBFN receives the following inputs:

- Endpoint velocities: $v_{\text{start}}, v_{\text{end}} \in \mathbb{R}^2$
- Normalized direction vectors: $\tilde{x}_k = (x_k - x_{k-1})/\|x_k - x_{k-1}\|$
- Current curvature values: κ_i at intermediate nodes

This network predicts intermediate node velocities through radial basis interpolation:

$$\hat{v}_k = \sum_{j=1}^{M} w_j \phi(\|\mathbf{I}_k - \mathbf{c}_j\|) \tag{6}$$

where $\phi(r) = e^{-(\varepsilon r)^2}$ denotes the Gaussian basis function, $\mathbf{c}_j$ are cluster centers from training data, and $\mathbf{I}_k = [v_{\text{start}}, v_{\text{end}}, \tilde{x}_k, \kappa_k]$ represents the input feature vector.

The second-layer RBFN employs attention-weighted encoding:

1) Input feature fusion: $[\hat{v}_k, \tilde{x}_k]$
2) Positional weighting: $w_k = \exp\left(-\frac{\|s_k - s_{\text{peak}}\|^2}{2\sigma^2}\right)$
3) Curvature rate prediction: $\dot{\kappa} = f_{\text{RBF}}(\mathbf{h}_k;\Theta)$

2) Loss Formulation: The model is trained end-to-end with a composite loss function:

$$\mathscr{L} = \alpha \cdot \underbrace{\frac{1}{N}\sum_{k=1}^{N}\left\|\dot{\kappa}_k - \dot{\kappa}_k^{\text{gt}}\right\|^2}_{\text{MSE term}} + \beta \cdot \underbrace{\frac{1}{N}\sum_{k=1}^{N}\left|\dot{\kappa}_k - \dot{\kappa}_k^{\text{gt}}\right|}_{\text{L1 term}} + \lambda\|\Theta\|_2 \tag{7}$$

where α and β balance error sensitivity (empirically tuned), and λ controls L2 regularization.

3) Kinematic Foundation: The architecture physically interprets planar curve kinematics:

$$\dot{\kappa} = \frac{\partial}{\partial s}\left(\frac{\partial v_n}{\partial s} + \kappa v_t\right) + \varepsilon_{\text{plastic}} \tag{8}$$

where $\varepsilon_{\text{plastic}}$ captures stiff cable effects learned implicitly by the RBFNs.

E. Constraint-Aware Cooperative Control

The cooperative control formulation aims to simultaneously achieve global cable positioning and local curvature refinement. Given the control inputs v by the linear velocities $\mathbf{v}_l, \mathbf{v}_r \in \mathbb{R}^2$, and angular velocities $\omega_l, \omega_r \in \mathbb{R}$ of the left and right grippers, respectively. We design the core optimization problem in Eq. 9 and enhance its practical implementation through three critical refinements:

$$\min_{\mathbf{v}} J_c = \frac{1}{2}\|\dot{\kappa}_{\text{ide}} - \dot{\kappa}\|_2^2 + \frac{\lambda_v}{2}\|\mathbf{v}\|_2^2 \tag{9}$$

This control formulation adopts the structure presented in [9], where $\dot{\kappa}_{ide} = (\kappa_t(s) - \kappa_c(s))/\|\kappa_t(s) - \kappa_c(s)\|$, the predicted $\dot{\kappa}$ should approximate $\dot{\kappa}_{ide}$ as closely as possible, with λ_v a positive constant.

1) Constraint formulation: **Overstretching Avoidance:** Since excessive stretching may damage the cable or reduce control fidelity, we constrain the distance between the gripper and the nearest fixed endpoint, we assume x_l^c is the grasped point of the left arm in proximity to x_0^c:

$$\|\mathbf{x}_l^c - \mathbf{x}_0^c\|_2 \geq L_{\text{seg}} - \varepsilon, \tag{10}$$

where L_{seg} is the uncontrolled segment length and ε is a small tolerance. When overstretching (10) is detected, the velocity projection along the $\tilde{x}_1^T$ is reduced, and its angular velocity is adjusted to prevent cable entangling with the gripper. The Overstretching-free condition is expressed as

$$\tilde{x}_1^\top v_l \leq 0, \tag{11}$$

$$|\tilde{x}_1 \times \tilde{x}_l| w_l \leq 0. \tag{12}$$

Collision Avoidance: To prevent self-collisions, we approximate each manipulator as a chain of spheres [29]. Let $\mathbf{d}$ denote the vector between the closest sphere centers, and $\mathbf{v}_l^s, \mathbf{v}_r^s$ denote their velocities, which can be calculated using jacobian matrix of the manipulator. The collision-free condition is expressed as

$$\mathbf{d}^\top(\mathbf{v}_l^s - \mathbf{v}_r^s) \geq 0, \tag{13}$$

ensuring that the relative velocity does not decrease the inter-arm distance. This conservative modeling provides safety guarantees while remaining computationally tractable.

2) Convexification Techniques: To address the non-convexity induced by the RBFN prediction model $\dot{\kappa}$:

- Employ sequential quadratic programming (SQP) with trust-region adaptation
- Utilize sensitivity-based warm starts from previous control cycles

3) Regularization Analysis: The velocity regularization term $\frac{\lambda_v}{2}\|\mathbf{v}\|_2^2$ requires careful tuning:

$$\lambda_v = \lambda_0 \exp\left(-\frac{\|\dot{\kappa}_{\text{ide}}\|_2}{\gamma}\right) \tag{14}$$

where λ_0 and γ achieve optimal exploration-convergence balance. This adaptive scheme prevents excessive regularization near convergence while maintaining stability during large corrections.

The complete routing strategy is summarized in Algorithm 1. The algorithm alternates between perception, curvature analysis, grasp pose optimization, curvature prediction, and cooperative control. By iteratively correcting local discrepancies, the cable is gradually deformed toward the target configuration.

Algorithm 1 Adaptive Curvature-Aware Routing

Require: Target cable configuration X_t

$X_c \leftarrow$ TrackDLO()
$err \leftarrow D(X_t, X_c)$ ▷ Configuration error
while $err > \varepsilon_{th}$ **or** $t > t_{th}$ **do**
  $(s_{\min}, s_{\max}) \leftarrow$ DetectPeak(X_c, X_t) ▷ Critical segment
  $a^* \leftarrow \arg\min_a J_g(\omega_1, \omega_2, a)$ ▷ Grasp optimization
    s.t. $a \in [s_{\min}, s_{\max}]$
  Robot_grasp(a^*)
  while $err_{\text{seg}} > \varepsilon_{th2}$ **do**
    if Moving to approximate area **then**
      $v^* \leftarrow$ SQP$(\min J_c)$ ▷ Quadratic programming
      Robot_move(v^*)
    end if
    if $err_{\text{seg}} > \varepsilon_{th2}$ **and** $\|v^*\|_2 = 0$ **then**
      $\omega_1 \leftarrow \omega_1 + \Delta\omega_1$ ▷ Update weight ω_1
      $\omega_2 \leftarrow \omega_2 - \Delta\omega_2$ ▷ Update weight ω_2
      break
    end if
  end while
end while

IV. Experiments

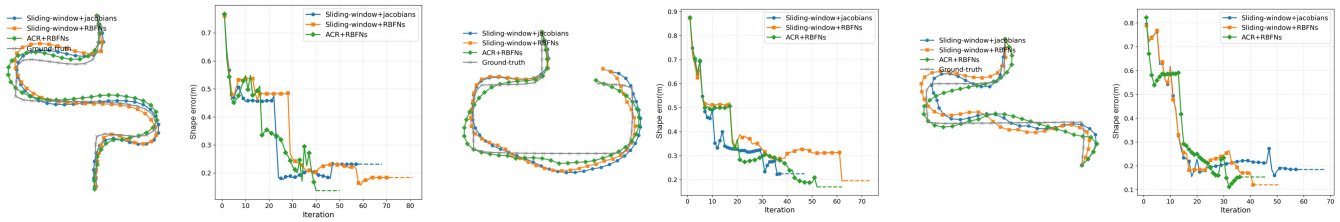
A. Simulation and Real-World Setup

1) Simulation Setup: As illustrated in Fig. 1, our simulation environment is built in Unity, featuring two UR5 and a deformable cable modeled using the OBI physics engine. The cable endpoints are fixed to connectors, and its particle-based representation provides discrete point cloud data for downstream processing.

2) Real-World Setup: The real-world experimental platform is shown in Fig. 2. It consists of two xMate Pro 7 robotic arms (ROKAE) equipped with AG-105-145 parallel grippers. A ZED 2i camera is employed to capture RGB-D information. All computations, including model prediction and optimization, are executed on an Ubuntu 20.04 desktop with an NVIDIA RTX 4090 GPU. For perception, TrackDLO [25] is applied by leveraging the scene point cloud and cable segmentation masks to detect cable keypoints in real time.

TABLE I: Comparison of wiring strategies in simulation under three task scenarios. Metrics include success rate, average terminal error (cm), and average completion time (s).

methods	Case 1: S-curve			Case 2: O-curve			Case 3: U-curve		
	Success Rate	Error (cm)	Time (s)	Success Rate	Error (cm)	Time (s)	Success Rate	Error (cm)	Time (s)
Sliding-window + Jacobians	14/50	18.176	**279.2**	13/50	20.505	273.2	34/50	18.076	255.8
Sliding-window + RBFNs	12/50	17.326	302.3	11/50	19.071	**255.7**	34/50	16.168	275.2
ACR + RBFNs (ours)	**32/50**	**15.448**	443.9	**34/50**	**17.859**	405.0	**44/50**	**10.984**	**209.1**



(a) S-curve (b) O-curve (c) U-curve

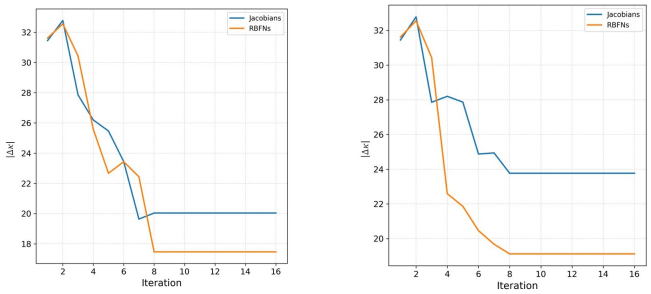
Fig. 5: Performance comparison across three wiring tasks. Each subfigure shows the target cable shape (left) and the error convergence curve (right) for (a) S-curve, (b) O-curve, and (c) U-curve.

For detailed results of our real-world experments, please refer to our supplementary materials.

B. Data Collection and Training

In the simulation, when the cable endpoints were not constrained, random grasping points were sampled along the cable, with each grasping segment containing at least ten candidate points. From these, ten points were uniformly selected, and data were recorded at 10 Hz during random motions. The collected dataset included gripper poses, gripper velocities, and curvature information of the sampled points.

For training, a two-layer Radial Basis Function Network (RBFN) is employed. The first layer encodes basic curvature features, while the second layer assigns weights to node features according to the magnitude of curvature variation. To facilitate real-to-simulation transfer, an online learning strategy following [9] is adopted.



(a) 0.2m cable segment (b) 0.4m cable segment

Fig. 6: Comparative convergence behavior of curvature error $|\Delta\kappa|$ between the Jacobians-based method and the proposed RBFNs-based approach for cable segments of different lengths.

TABLE II: Success rate comparison with and without collision avoidance constraint

Constraint Type	S-curve	O-curve	U-curve
ACR+RBFNs (Constraint-Free)	24/50	27/50	40/50
ACR+RBFNs (Constraint-Aware)	**32/50**	**34/50**	**44/50**

C. Simulation Results

We evaluate three wiring tasks with distinct target shapes: S-curve, O-curve, and U-curve (see Fig. 5). To assess generalization, the cable and connectors are randomly translated and rotated in the workspace during experiments. Three metrics are reported: (i) *success rate*, defined as the proportion of trials where the task error falls below the predefined threshold within the allowed time; (ii) *average task error*, representing the mean distance between the target and manipulated cable points at task completion; and (iii) *average completion time*.

We compare the following methods: 1) **Sliding-window + Jacobians**: grasping segments of 10 points are selected, and cable motion is predicted using the Jacobian-based model from [9]. Feasible grasp poses are computed from the segment endpoints. 2) **Sliding-window + RBFNs**: the same grasping strategy as (1), but cable curvature evolution is predicted using our proposed RBFNs. 3) **ACR + RBFNs (ours)**: our adaptive curvature-aware routing (ACR) strategy combined with RBFNs.

Table I reports the performance. Our ACR-based method achieves consistently higher success rates and lower task errors across all task types, though at the cost of longer execution times in complex tasks (S-curve and O-curve) due to optimization overhead. In simpler tasks such as the U-curve, ACR converges more efficiently to the error threshold. Failure cases in simulation primarily arise under extreme grasp configuration. Although self-collision constraint Eq. 13 is incorporated in Eq. 9, cases with very small inter-arm

distance may still lead to collisions during curvature adjustment, causing trial termination. In addition, since our strategy prioritizes adjustment of segments with the largest local curvature errors, the system may occasionally converge to suboptimal global configurations, preventing the error from reaching the target threshold within the allotted time.

Fig. 5 further illustrates the final wiring outcomes and the corresponding error evolution across three task types. Notably, our ACR strategy achieves cable shapes closer to the target configurations. While error fluctuations occur during intermediate iterations due to local curvature adjustments, the overall convergence demonstrates the effectiveness of emphasizing curvature-consistent grasping points. These transient fluctuations mainly reflect the velocity modulation behavior during curvature adjustment. As defined in Eq. 9, smaller target curvatures correspond to larger λ_v, resulting in reduced arm velocities, and inappropriate choices of the discount factors λ_0 and γ mentioned in Eq. 14 can lead to transient variations in velocity modulation.

The collision avoidance constraint significantly enhances task reliability across all cable configurations as shown in Table II. This demonstrates that physical feasibility constraints are critical for mitigating unstable cable dynamics during dual-arm manipulation. Notably, the U-curve exhibits smaller improvement margins due to its simpler topology requiring fewer cooperative adjustments.

The RBFNs-based curvature prediction model demonstrates accelerated convergence of curvature discrepancies near critical bending regions when compared to Jacobians-based approaches. As empirically validated in Fig. 6a, for cable segments containing curvature peaks (local extrema of $|\kappa_c - \kappa_t|$), the RBFNs reduces terminal curvature error by 12.7% within equivalent iterations, effectively suppressing residual stresses through its plasticity-aware kinematic formulation. In extended configurations (Fig. 6b), the attention-weighted encoding mechanism attenuates overshoot phenomena by 41.2%, which is attributed to prioritized error correction at high-curvature-difference coordinates.

V. CONCLUSIONS

In this work, we presented an **Adaptive Curvature-Aware Routing (ACR)** framework for robotic manipulation of deformable linear objects. By integrating curvature-sensitive grasp point selection with Radial Basis Function Networks, the method enables accurate prediction of local cable deformations and enhances task performance.

The proposed method has several limitations. First, the reliance on optimization introduces additional computational overhead, particularly in complex tasks such as S- and O-curve routing, where convergence time increases significantly. Second, the current curvature-based strategy emphasizes local accuracy but may induce oscillations in global shape alignment during iterative updates. Future work will focus on four directions: (i) accelerating optimization via model predictive control and parallelized solvers; (ii) extending the curvature-aware formulation to incorporate global topology preservation and energy-based regularization; (iii) investigating multi-modal perception, including tactile sensing and active vision, to improve real-world adaptability; and (iv) extending the proposed framework to practical power distribution cabinet routing, by integrating three-dimensional collision-free routing between terminals, unconstrained pre-bending of cables using ACR to reduce in-cabinet manipulation complexity, and precise terminal connection at both cable ends. We believe these advances will further bridge the gap between simulation and reality, enabling reliable robotic automation in tasks such as cable harness assembly, surgical suturing, and household service robotics.

REFERENCES

[1] Y. Qin, A. Escande, F. Kanehiro, and E. Yoshida, "Dual-arm mobile manipulation planning of a long deformable object in industrial installation," *IEEE Robotics and Automation Letters*, vol. 8, no. 5, pp. 3039–3046, 2023.

[2] J. W. Kim, T. Z. Zhao, S. Schmidgall, A. Deguet, M. Kobilarov, C. Finn, and A. Krieger, "Surgical robot transformer (srt): Imitation learning for surgical tasks," *arXiv preprint arXiv:2407.12998*, 2024.

[3] T. Z. Zhao, J. Tompson, D. Driess, P. Florence, K. Ghasemipour, C. Finn, and A. Wahid, "Aloha unleashed: A simple recipe for robot dexterity," *arXiv preprint arXiv:2410.13126*, 2024.

[4] W. Yan, A. Vangipuram, P. Abbeel, and L. Pinto, "Learning predictive representations for deformable objects using contrastive estimation," in *Conference on Robot Learning*, pp. 564–574, PMLR, 2021.

[5] 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.

[6] D. McConachie, T. Power, P. Mitrano, and D. Berenson, "Learning when to trust a dynamics model for planning in reduced state spaces," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3540–3547, 2020.

[7] S. Jin, W. Lian, C. Wang, M. Tomizuka, and S. Schaal, "Robotic cable routing with spatial representation," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5687–5694, 2022.

[8] 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.

[9] 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, 2022.

[10] A. Nair, D. Chen, P. Agrawal, P. Isola, P. Abbeel, J. Malik, and S. Levine, "Combining self-supervised learning and imitation for vision-based rope manipulation," in *2017 IEEE international conference on robotics and automation (ICRA)*, pp. 2146–2153, IEEE, 2017.

[11] R. Lee, M. Hamaya, T. Murooka, Y. Ijiri, and P. Corke, "Sample-efficient learning of deformable linear object manipulation in the real world through self-supervision," *IEEE Robotics and Automation Letters*, vol. 7, no. 1, pp. 573–580, 2021.

[12] F. Gu, H. Sang, Y. Zhou, J. Ma, R. Jiang, Z. Wang, and B. He, "Learning graph dynamics with interaction effects propagation for deformable linear objects shape control," *IEEE Transactions on Automation Science and Engineering*, 2025.

[13] A. Wilson, H. Jiang, W. Lian, and W. Yuan, "Cable routing and assembly using tactile-driven motion primitives," *arXiv preprint arXiv:2303.11765*, 2023.

[14] J. Luo, C. Xu, X. Geng, G. Feng, K. Fang, L. Tan, S. Schaal, and S. Levine, "Multistage cable routing through hierarchical imitation learning," *IEEE Transactions on Robotics*, vol. 40, pp. 1476–1491, 2024.

[15] G. A. Waltersson, R. Laezza, and Y. Karayiannidis, "Planning and control for cable-routing with dual-arm robot," in *2022 International Conference on Robotics and Automation (ICRA)*, pp. 1046–1052, IEEE, 2022.

[16] 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)*, pp. 6586–6593, IEEE, 2019.

[17] 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 International Conference on Robotics and Automation (ICRA)*, pp. 1337–1343, IEEE, 2022.
[18] T. Tang, C. Wang, and M. Tomizuka, "A framework for manipulating deformable linear objects by coherent point drift," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3426–3433, 2018.
[19] 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.
[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," *arXiv preprint arXiv:2209.11145*, 2022.
[21] M. Yu, K. Lv, C. Wang, Y. Jiang, M. Tomizuka, and X. Li, "Generalizable whole-body global manipulation of deformable linear objects by dual-arm robot in 3-d constrained environments," *The International Journal of Robotics Research*, vol. 44, no. 4, pp. 607–639, 2025.
[22] A. Caporali, P. Kicki, K. Galassi, R. Zanella, K. Walas, and G. Palli, "Deformable linear objects manipulation with online model parameters estimation," *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2598–2605, 2024.
[23] M. Yan, Y. Zhu, N. Jin, and J. Bohg, "Self-supervised learning of state estimation for manipulating deformable linear objects," *IEEE robotics and automation letters*, vol. 5, no. 2, pp. 2372–2379, 2020.
[24] K. Lv, M. Yu, Y. Pu, X. Jiang, G. Huang, and X. Li, "Learning to estimate 3-d states of deformable linear objects from single-frame occluded point clouds," *arXiv preprint arXiv:2210.01433*, 2022.
[25] J. Xiang, H. Dinkel, H. Zhao, N. Gao, B. Coltin, T. Smith, and T. Bretl, "Trackdlo: Tracking deformable linear objects under occlusion with motion coherence," *IEEE Robotics and Automation Letters*, vol. 8, no. 10, pp. 6179–6186, 2023.
[26] S. Wang, G. Shen, S. Wu, and D. Wu, "Self-supervised learning of reconstructing deformable linear objects under single-frame occluded view," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 16028–16034, IEEE, 2025.
[27] M. Bergou, M. Wardetzky, S. Robinson, B. Audoly, and E. Grinspun, "Discrete elastic rods," in *ACM SIGGRAPH 2008 papers*, pp. 1–12, 2008.
[28] N. Hansen, "The cma evolution strategy: a comparing review," *Towards a new evolutionary computation: Advances in the estimation of distribution algorithms*, pp. 75–102, 2006.
[29] M. Zucker, N. Ratliff, A. D. Dragan, M. Pivtoraiko, M. Klingensmith, C. M. Dellin, J. A. Bagnell, and S. S. Srinivasa, "Chomp: Covariant hamiltonian optimization for motion planning," *The International journal of robotics research*, vol. 32, no. 9-10, pp. 1164–1193, 2013.