Untangling Dense Non-Planar Knots by Learning Manipulation Features and Recovery Policies

Priya Sundaresan*, Jennifer Grannen*, Brijen Thananjeyan, Ashwin Balakrishna, Jeffrey Ichnowski, Ellen Novoseller, Minho Hwang, Michael Laskey, Joseph E. Gonzalez, Ken Goldberg

Abstract—Robot manipulation for untangling 1D deformable structures such as ropes, cables, and wires is challenging due to their infinite dimensional configuration space, complex dynamics, and tendency to self-occlude. Analytical controllers often fail in the presence of dense configurations, due to the difficulty of grasping between adjacent cable segments. We present two algorithms that enhance robust cable untangling, LOKI and SPiDERMan, which operate alongside HULK, a high-level planner from prior work. LOKI uses a learned model of manipulation features to refine a coarse grasp keypoint prediction to a precise, optimized location and orientation, while SPiDERMan uses a learned model to sense task progress and apply recovery actions. We evaluate these algorithms in physical cable untangling experiments with 336 knots and over 1500 actions on real cables using the da Vinci surgical robot. We find that the combination of HULK, LOKI, and SPiDERMan is able to untangle dense overhand, figure-eight, double-overhand, square, bowline, granny, stevedore, and triple-overhand knots. The composition of these methods successfully untangles a cable from a dense initial configuration in 68.3% of 60 physical experiments and achieves 50% higher success rates than baselines from prior work. Supplementary material, code, and videos can be found at https://tinyurl.com/rssuntangling.

I. INTRODUCTION

Robot manipulation of 1D deformable objects, such as cables, ropes, and wires, can facilitate automation of tasks in industrial, surgical, and household settings [9]. Such manipulation tasks include organization of cables [7, 27], hoses [23], suture thread [31], and wires used in automotive and other electronic assembly [12], as well as reducing clutter and preventing injury in surgical, manufacturing, and home environments [2]. As robots automate more tasks involving 1D deformable objects, which we refer to as “cables,” they will increasingly confront highly complex knots, either because the task itself requires knot untangling or because untangling cables is a prerequisite for a downstream task. Prior work has explored untangling manipulation plans consisting of high-level manipulation primitives (e.g., pick and pull points) [28, 19, 7], but robots often face challenges when attempting to execute these plans in physical manipulation tasks, due to the complexity of dynamics modeling and perception in real-world settings. This motivates the need for a low-level controller which can mitigate the gap between high-level planning and physical execution. This is relevant to any task involving untangling or arrangement of cables where sensing grasping precision or recovering from manipulation failures is critical. We propose a low-level controller consisting of two algorithms that jointly improve the robustness of robot untangling.
Untangling cables is a challenging task due to the infinite-dimensional configuration space, self-occlusions, and visual homogeneity of cables. These properties complicate the problems of both generating untangling manipulation plans and executing these plans effectively, especially in densely knotted configurations. Prior work for cable manipulation has primarily explored manipulation planning methods that learn full state information \cite{25,19} or task-specific features \cite{7}. These methods then execute planned actions through a sequence of analytical, hand-engineered motion primitives, which can be brittle and imprecise under increasing knot complexity and density due to layered self-occlusions, friction at cable intersections, and the need for precision in grasping. Most prior work considers cable configurations that have sufficiently wide space between self-intersections for a gripper jaw to fit or that have only two cable segments at each self-intersection. These manipulation challenges are exacerbated for non-planar configurations, where configuration complexity increases to intersections with three or more segments.

Grannen et al. \cite{7} propose HULK, a high-level vision-based planner for untangling of cables in semi-planar configurations, where self-intersections involve at most two segments of cable. In this work, we extend HULK with two novel algorithms for non-planar knots. The first algorithm, LOKI (Local Oriented Knot Inspection), learns manipulation features to generate high-precision grasping actions. LOKI refines the cable grasp location and orientation from a coarse keypoint by recentering and learning to grasp orthogonal to the cable’s axis, as illustrated in Figure 1. The second algorithm, SPIIDER-Man (Sensing Progress in Dense Entanglements for Recovery Manipulation) detects and recovers from 4 common failure modes: (1) consecutive poor action executions, (2) exceeding a maximum threshold of untangling actions, (3) the cable leaving the reachable workspace, and (4) the robot gripper jaws becoming wedged between two segments of dense cable. Unlike prior work which focuses on high precision for failure avoidance \cite{22,16}, SPIIDER-Man acknowledges the likelihood of failures in real-world settings and plans recovery actions.

We perform physical untangling experiments with cables in dense, non-planar configurations more complex than the semiplanar arrangements considered in prior work and find that the combination of HULK, LOKI, and SPIIDER-Man achieves 50% higher success rates in comparison to analytical controllers.

II. RELATED WORK

Deformable object manipulation has seen a surge of interest recently in the robotics research community \cite{18,34,33,28,6,10,25,26}, but modeling state and action spaces in such tasks remains challenging. Much prior work focuses on the complex task of generating plans for manipulating cables and higher-dimensional deformable objects. Executing the plans, especially in the presence of failure modes, is often left to future work. In this work, we focus on addressing untangling-specific failure modes by learning controllers to robustly execute plans for the task.

A. Deformable Object Manipulation

Several perception-focused techniques have proven effective in multi-step robotic manipulation of cables and other deformable objects such as cloth and bags. One approach performs state estimation and leverages the learned state representation for downstream planning. For instance, Lui and Saxena \cite{19} and Chi and Berenson \cite{3} propose using classical visual feature extraction to estimate the state of deformable rope and cloth, respectively, subject to partial occlusion. Sundaresan et al. \cite{28} investigate object representation learning via dense object descriptors \cite{24,5} for rope knot-tying and arrangement, and Ganapathi et al. \cite{6} extend this methodology to 2D fabric smoothing and folding. Alternative perception-driven approaches explore learning latent state spaces \cite{34} for cloth manipulation, or using semantic keypoint perception \cite{7,33} for rope tracking. However, robust state estimation remains challenging for heavily self-occluded configurations, as are encountered in densely knotted cables. This work builds upon prior work on keypoint prediction for cable untangling, but unlike previous work, we couple perception with robust learned control policies for grasping and failure recovery.

Other methods learn visuomotor policies end-to-end. Imitation learning \cite{25}, video prediction models \cite{10}, model-free reinforcement learning (RL) \cite{17}, and model-based RL \cite{20} have successfully been applied to goal-conditioned fabric manipulation. In other end-to-end approaches, Nair et al. \cite{21} learn a dynamics model for performing template shape matching and knot-tying with rope. Zhang et al. \cite{35} learn to directly regress robotic configurations in joint space for robotic vaulting of cables over obstacles. While these approaches are broadly general and can flexibly apply to many tasks, they can lack precision and do not leverage geometric structure in the problem, making them difficult to apply to dexterous manipulation tasks such as cable untangling, in which geometric reasoning and fine-grained manipulation are critical.

Several recent works have also explored learning robot policies for deformable manipulation entirely in simulation and transferring them to the real world via domain randomization \cite{26,28,34,6,10,26,35}. By contrast, we train perception systems using a combination of data from simulation and the physical world. Crucially, we observe that physical data is critical for tasks that require global reasoning, as it is difficult to simulate the full distribution of configurations encountered during an untangling sequence. However, for tasks that only require local reasoning within specific knots, simulated data can provide sufficient information for robust perception.

B. Cable Untangling

To the best of our knowledge, Lui and Saxena \cite{19} is the first published study of robot cable untying. The authors use RGB-D sensing and classical feature extraction to approximate a tangled cable as a graph consisting of cable crossings and endpoints. We primarily build upon the work of Grannen et al. \cite{7}, which (1) presents a geometric algorithm for cable untangling based on the graphical abstraction from Lui and Saxena \cite{19} and (2) uses deep learning to detect cable crossings with
keypoint inference. These keypoints serve as input to a greedy action planner that iteratively reduces the number of crossings.

Lui and Saxena [19] and Grannen et al. [7] present algorithms for generating untangling manipulation plans for cables with semiplanar knots. However, their executed controllers are based on a set of analytical motion primitives, which make simplified assumptions about cable geometry that do not hold in highly deformed or occluded states.

In this work, we focus on the problem of learning a controller to robustly execute the manipulation plans generated by Grannen et al. [7]. First, in adapting the method to a more challenging class of knots, we employ coarse-to-fine refinement strategies to perform robust cable grasping. Several recent works have also studied coarse-to-fine controllers for manipulation tasks requiring precision, including surgical peg transfer [22] and peg insertion [16, 13]. Secondly, we explore recovery from manipulation-induced errors in contact-based robotics tasks [22] [4, 16]. However, these works do not exploit task recovery from manipulation errors in contact-based robotics of closed-loop untangling. Recent work has also considered recovery from manipulation-induced errors in the context of transfer [22] and peg insertion [16, 13].

We model cable state via an undirected graph $G = (V, E)$ with cable endpoints and crossing locations represented as vertices $v \in V$ and cable segments between vertices represented as edges $e = (u, v) \in E$ for $u, v \in V$, where $u \neq v$. We use the term “node” and “vertex” interchangeably and label the nodes corresponding to the left and right endpoints as $v_l$ and $v_r$, respectively. For a node and adjacent edge $(v, e) \in (V, E)$, we annotate the graph with $X(v, e) \in \{-1, 1, 0\}$ (Equation 1) to designate the cable segment hierarchy at each node:

$$X(v, e) = \begin{cases} +1 & \text{if } v \text{ is an endpoint or if } e \text{ crosses above all other edges at } v \\ -1 & \text{if } e \text{ crosses below only one edge at } v \\ 0 & \text{if } e \text{ crosses below two other edges at } v \end{cases}$$ (Equation 1)

We define an under-crossing as a set of one node $v$ and two incident edges $e_i, e_j \in E$, where $e_i = (v, v')$ and $e_j = (v, v'')$ for some $v', v'' \in V$, $X(e_i, v) = X(e_j, v) \in \{-1, 0, 1\}$, and $e_i, e_j$ are contiguous in the cable. Thus, the cable segment represented by $e_i, e_j$ is occluded by one or two other cable segments. Similarly, an over-crossing is a set of one node $v$ and two incident edges $e_i = (v, v')$ and $e_j = (v, v'')$ for some $v', v'' \in V$, where $X(e_i, v) = X(e_j, v) = +1$ and $e_i, e_j$ are contiguous in the cable. Under the non-planar assumption, each node has a degree of at most 6, corresponding to 3 cable segments at a crossing. Edges with the same annotation $X(v, e)$ at a node $v$ represent the same cable segment on either side of the crossing. A cable with $|V| = 2$ has only endpoint nodes and no crossings, and thus is fully untangled.

Each gripper performs either a holding, pulling, or rotating action at each time $t$. Below, we define all actions with respect to the 2D image frame. The left robot arm performs a pulling action, $a_{l,t}$, by grasping the cable at the pixel $(x_{l,t}, y_{l,t})$ with a (vertical) $z$-axis grasping rotation of $\theta_{t,l}$, lifting by a fixed amount, pulling to $(x_{l,t} + \Delta x_{l,t}, y_{l,t} + \Delta y_{l,t})$, and releasing the cable. Holding actions are indicated via $(\Delta x_{r,t}, \Delta y_{r,t}) = (0, 0)$. We denote moving between points without grasping the cable via $1_{\text{grasp}} = 0$. Lastly, $\Delta \theta_{t,l}$ (in degrees) indicates rotating a grasped cable about the $z$-axis. The right robot arm uses an analogous action formulation, $a_{r,t}$. We show the full action representation for both arms in Equation 2.
\[ a_{t,l} = (x_{t,l}, y_{t,l}, \theta_{t,l}, \Delta x_{t,l}, \Delta y_{t,l}, \Delta \theta_{t,l}, \mathbb{1}_{\text{grasp}}) \]
\[ a_{t,r} = (x_{t,r}, y_{t,r}, \theta_{t,r}, \Delta x_{t,r}, \Delta y_{t,r}, \Delta \theta_{t,r}, \mathbb{1}_{\text{grasp}}). \]

This action space generalizes that of Grannen et al. [7], which only parameterizes grasp points and pull vectors.

IV. PRELIMINARIES

In this section, we review the BRUCE and HULK algorithms introduced in Grannen et al. [7] and discuss relevant algorithmic modifications to accommodate non-planar knots.

A. BRUCE: Basic Reduction of Under-Crossing Entanglements

Grannen et al. [7] study semi-planar knot untangling given observations of the knot’s graphical representation, and propose Basic Reduction of Under-Crossing Entanglements (BRUCE), an algorithm to iteratively untangle knots starting from the rightmost endpoint. BRUCE makes use of two manipulation primitives adapted from Lui and Saxena [19]: (1) Reidemeister moves pull the cable endpoints apart to reduce self-occlusions that are not part of a knot, and (2) Node Deletion moves delete a crossing in the graph by holding an over-crossing edge (+1) and pulling out the under-crossing edge in place with one arm while the other arm pulls out the cable from the corresponding under-crossing.

BRUCE takes a Reidemeister move to disambiguate the cable state, followed by alternating Node Deletion and Reidemeister moves to sequentially delete crossings and attempt to linearize the cable. In semi-planar configurations, Node Deletion moves pull out the under-crossing edge (labeled −1) exiting the first node \( v \) traced from the right endpoint, while holding a corresponding over-crossing edge (+1) at \( v \) in place.

This work extends Node Deletion moves to operate on non-planar crossings, which contain under-crossing edges labeled as both −1 and −2. In this setting, a Node Deletion move holds an over-crossing edge (+1) and pulls out the under-crossing edge (−1 or −2) traced from the right endpoint. This action reduces \(|E|\) and \(|V|\) by at least 2 and 1, respectively, until the algorithm terminates with \(|V| = 2\): a fully linear cable with no crossings and two endpoints.

B. HULK: Hierarchical Untangling from Learned Keypoints

Grannen et al. [7] consider cable untangling from RGB image observations, where the ground truth cable graph representation is not directly observable. To infer untangling actions while bypassing the need for full graph reconstruction, Grannen et al. [7] propose Hierarchical Untangling from Learned Keypoints (HULK), which predicts keypoints from images to execute BRUCE’s manipulation primitives. HULK learns a mapping from an image to four Gaussian heatmaps centered at task-relevant keypoints, \( f : \mathbb{R}^{640 \times 480 \times 3} \rightarrow \mathbb{R}^{640 \times 480 \times 4} \). The means of the four heatmap outputs, denoted by \( \hat{p}_l \), \( \hat{p}_r \), \( \hat{p}_{\text{pull}} \), and \( \hat{p}_{\text{hold}} \), indicate the predicted pixel locations of the left and right cable endpoints (\( \hat{p}_l \) and \( \hat{p}_r \)) and the predicted pull and hold grasp locations for the next planned Node Deletion move (\( \hat{p}_{\text{pull}} \) and \( \hat{p}_{\text{hold}} \)). HULK learns to predict the pull and hold keypoints at the first under/over-crossing pair relative to the rightmost endpoint.

In this work, we adapt the HULK training procedure from Grannen et al. [7], originally trained on semiplanar knots, to operate on dense, non-planar configurations. Since training HULK only requires light supervision in the form of keypoints, we follow the training procedure in [7] and train HULK on 200 pairs of images and hand-annotations for non-planar configurations. As in [7], we use a Gaussian standard deviation of 8 px for the heatmap training data centered about each hand-labelled keypoint, and we apply various data augmentation techniques including image adjustments and affine transformations to generate a training dataset of 3,500 examples. Once trained, HULK uses the predicted keypoints \( \hat{p}_l \), \( \hat{p}_r \), \( \hat{p}_{\text{pull}} \), and \( \hat{p}_{\text{hold}} \) to plan a bilateral move (\( a_{t,l}, a_{t,r} \)) at time \( t \) for the left and right arms, respectively, as follows.

For a Node Deletion move, the right arm holds the configuration at \( \hat{p}_{\text{hold}} \) while the left arm grasps at \( \hat{p}_{\text{pull}} \) and pulls in the direction away from \( \hat{p}_{\text{hold}} \) by some fixed vector \( n = (n_x, n_y) \) to slacken the cable. Both arms use coarse analytical grasp rotations \( \theta_{\text{pull}} = \arctan \left( \frac{\hat{p}_{\text{pull},y} - \hat{p}_{\text{hold},y}}{\hat{p}_{\text{pull},x} - \hat{p}_{\text{hold},x}} \right) \) and \( \theta_{\text{hold}} = \theta_{\text{pull}} + 90^\circ \):

\[ a_{t,l} = (\hat{p}_{\text{pull},x}, \hat{p}_{\text{pull},y}, \hat{p}_{\text{hold},x}, \hat{p}_{\text{hold},y}, 0, 1) \]
\[ a_{t,r} = (\hat{p}_{\text{hold},x}, \hat{p}_{\text{hold},y}, \theta_{\text{hold}}, 0, 0, 0, 1). \]

For a Reidemeister move, the left and right arms simultaneously grasp at \( \hat{p}_l \) and \( \hat{p}_r \) and pull the endpoints to opposing workspace ends \( w_l \) and \( w_r \), with analytical grasp rotations \( \theta_l \) and \( \theta_r \) obtained as the angle between \( (1, 0) \) and the vector orthogonal to the first principal component of a masked crop around \( \hat{p}_l \) and \( \hat{p}_r \):

\[ a_{t,l} = (\hat{p}_{l,x}, \hat{p}_{l,y}, \hat{\theta}_l, w_{l,x} - \hat{p}_{l,x}, w_{l,y} - \hat{p}_{l,y}, 0, 1) \]
\[ a_{t,r} = (\hat{p}_{r,x}, \hat{p}_{r,y}, \hat{\theta}_r, w_{r,x} - \hat{p}_{r,x}, w_{r,y} - \hat{p}_{r,y}, 0, 1). \]

V. METHODS

We present two new algorithms, LOKI (Local Oriented Knot Inspection) and SPiDERMan (Sensing Progress in Dense Entanglements for Recovery Manipulation), which are combined to learn a robust controller to increase robustness of high-level manipulation plans from Grannen et al. [7].
A. LOKI: Local Oriented Knot Inspection

To enable finer-grained grasp planning for the knot untying primitives of HULK, we introduce a low-level controller called LOKI. This controller jointly infers robust antipodal grasps that enclose a cable segment orthogonally and performs coarse-to-fine refinement of keypoints. These adjustments are designed to prevent near-miss grasps, a common failure mode in HULK [7]. While prior work in cable manipulation has employed analytical grasp planning [28, 7], such heuristic methods fail to generalize to dense, non-planar configurations. Thus, we propose LOKI: Local Oriented Knot Inspection. LOKI maps a locally-cropped cable image, centered at one of the four HULK keypoints $\mathbf{p} \in \{\mathbf{p}_1, \mathbf{p}_r, \mathbf{p}_p, \mathbf{p}_h\}$, to (1) $\theta$: an angle about the $z$-axis denoting the top-down grasp orientation and (2) $p_{off} = (p_{off,x}, p_{off,y})$: a local offset in pixel space to recenter the keypoint along the cable width. We implement LOKI as a multi-headed deep neural network trained with a binary cross-entropy loss. To determine when a recovery action will be performed. In this way, SPiDERMan also applies analytical contour-based perception to implement recovery manipulation primitives. Given a

\[
\hat{p}_{off,x}, \hat{p}_{off,y} = \gamma \left( \arg\max_{(u_c, v_c)} H[u, v] - (u_c, v_c) \right),
\]

where $\gamma = 60/200$ is a downsampling factor to account for crop resizing, and $(u_c, v_c) = (100, 100)$ is the upscaled crop center (i.e., coarse keypoint prediction). The refined keypoint $(\mathbf{p}_x + \hat{p}_{off,x}, \mathbf{p}_y + \hat{p}_{off,y})$ and predicted gripper orientation $\theta$ are used for planning grasps across all manipulation primitives.

B. SPiDERMan: Sensing Progress in Dense Entanglements for Recovery Manipulation

To sense and recover from manipulation failures, we propose SPiDERMan. SPiDERMan uses both learned and analytical methods to sense untangling progress and employs two novel manipulation primitives to avoid and recover from failures. We address four manipulation failure modes from Grannen et al. [7]: (1) consecutive poor action executions due to high cable density, (2) task incomplete after exceeding the maximum number of untangling actions, (3) the cable leaving the workspace during manipulation, and (4) the high-density cable becoming wedged in the robot jaws. We first discuss how SPiDERMan detects each failure mode from RGB image inputs and next define two manipulation primitives for failure mode recovery.

1) Detection: SPiDERMan uses two manipulation failures, SPiDERMan compares workspace image observations $I_t \in \mathbb{R}^{240 \times 480 \times 3}$ before and after executing actions, where $t'$ denotes the number of untangling, or Node Deletion, actions executed thus far. SPiDERMan is implemented with a ResNet-18 backbone and learns a classifier $d : \mathbb{R}^{240 \times 480 \times 6} \rightarrow \{0, 1\}$, trained with a binary cross-entropy loss. To determine when a recovery action is needed due to lack of untangling progress at test time, $d(I_1, I_2)$ uses a simple decision threshold of 0.5 over the softmax of the classifier output to determine whether image $I_1$ corresponds to a denser (1) or less dense (0) state than $I_2$. When two successive actions have not effectively loosened the knot, the following condition evaluates to true:

\[
d(I_t', I_t'-1) \text{ and } d(I_t'-1, I_t'-2),
\]

and we conclude that the configuration is pathological and a recovery action will be performed. In this way, SPiDERMan recognizes untangling progress over multiple actions, removing the need for a fixed untangling action limit. Instead, we define a termination condition:

\[
d(I_t', I_t'-5) \text{ or } d(I_{ref}, I_t')
\]

that checks both for untangling success against an untangled reference cable image $I_{ref}$ and for progress every 5 actions to prevent termination as long as there is progress. Condition (4) evaluates to true when no progress has occurred over the last 5 actions, or if the cable reaches a fully-untangled state.

When $I_t'$ is fully untangled, both $I_{ref}$ and $I_t'$ are equally dense. For this reason, we use an image observation of a knot-free cable with a single crossing for $I_{ref}$ (Figure 3) to prevent termination false negatives arising from a bias toward $I_{ref}$.

SPiDERMan also applies analytical contour-based perception to implement recovery manipulation primitives. Given a
workspace image observation $I_t$, we preprocess the image by converting to grayscale and applying binary thresholding. The algorithm detects all contiguous contours in the resulting image using the open-source implementation from Suzuki et al. [29] in OpenCV [1]. We extract an approximate contour of the cable $P = \{p_1, p_2, \ldots , p_n\}$ via contour area filtering. Next, SPiDERMan approximates the cable center as the mean point $\hat{p}_c$ of the contour, projected to the line on the cable: $\hat{p}_c = \arg\min_{p_i \in P} ||p_i - \hat{p}||$. Between actions, the left and right grippers move to known poses with gripper jaw pixel coordinates $g_l$ and $g_r$, respectively, outside of the field of view of the untangling workspace. Given $\hat{p}_c, g_l$, and $g_r$, Condition [5] detects when the cable is wedged in either of the grippers by checking when the distance between the approximate cable center, $\hat{p}_c$, and either gripper tooltip, $g_l \lor g_r$, is below a hand-tuned threshold of 20 px:

$$\min \{||\hat{p}_c - g_l||, ||\hat{p}_c - g_r||\} < 20 \text{ px}.$$ (5)

To detect when the cable is leaving the workspace, we define a second condition over the contour-based perception method above to check when the center of the cable mask $\hat{p}_c$ exceeds a hand-tuned 200 px threshold from the workspace center $w_c$:

$$||\hat{p}_c - w_c|| > 200 \text{ px}.$$ (6)

2) Recovery: We define two novel manipulation primitives to recover once a failure is detected: (1) Re-Posing moves reorient and place a cable at the workspace center, and (2) Wedged Recovery moves detach a gripper jaw wedged between two cable segments.

For a Re-Posing move, we improve the Recentering primitive from Grannen et al. [7] to rotate the cable in addition to centering the configuration in the workspace. The right arm first grasps the cable at the center of the configuration $\hat{p}_c$ with a grasp location $\hat{p}_c + \hat{p}_{c,\text{off}}$ and grasp rotation of $\hat{\theta}_c$ predicted by LOKI. The right arm then lifts by a fixed amount and moves the cable to the predefined workspace center $w_c$. When two successive poor actions are detected via Equation (3), the right gripper then rotates by 180° to re-orient before releasing the grasp. A Re-Posing move (Rotation) includes this rotation, while a Re-Posing move (Translation) does not:

$$\alpha_{t,r} = (\hat{p}_{c,x}, \hat{p}_{c,y}, \hat{\theta}_c, w_{c,x} - \hat{p}_{c,x}, w_{c,y} - \hat{p}_{c,y}, \{0^\circ, 180^\circ\}, 1) .$$

A Wedged Recovery move is comprised of two successive actions. When detecting that the left robot gripper jaw is wedged between cable segments, the left arm first brings the stuck cable to the workspace center $w_c$ and releases its grasp. Given the detected cable contour $P$, the right arm grasps the cable at the far right point on the cable mask $\hat{p}_{\text{hold}} = \arg\min_{p_i \in P} ||p_i - w_c||$ with a grasp location $\hat{p}_{\text{hold}} + \hat{p}_{\text{hold,off}}$ and grasp rotation $\hat{\theta}_{\text{hold}}$ given by LOKI and holds it down on the workspace while the left arm returns to its home position, a fixed height above the predefined point $w_r$:

$$\alpha_{t,l} = (w_{1,x}, w_{1,y}, 0, w_{c,x} - w_{1,x}, w_{c,y} - w_{1,y}, 0, 1)$$

$$\alpha_{t+1,l} = (\hat{p}_{\text{hold,off}}, \hat{p}_{\text{hold}}, \hat{\theta}_{\text{hold}}, 0, 0, 0, 1)$$

$$\alpha_{t+1,l} = (w_{c,x}, w_{c,y}, 0, w_{c,x} - w_{1,x}, w_{c,y} - w_{1,y}, 0, 0).$$

When the cable is wedged in the right robot gripper, we execute the equivalent procedure with the roles of the right and left arms switched. In this case, $\hat{p}_{\text{hold}}$ represents the far left point on the cable mask, and the right arm returns to the home position above the predefined point $w_r$.

Algorithm 1 presents the untangling algorithm combining HULK from Grannen et al. [7] with LOKI and SPiDERMan, and Figure 4 illustrates the system.

VI. EXPERIMENTS

We experimentally evaluate the separate and joint effectiveness of HULK, LOKI, and SPiDERMan in performing physical untangling of dense semi-planar and non-planar knots.

A. Overview of Policies

We compare (1) the proposed method (Alg. [1]) synthesizing HULK, LOKI, and SPiDERMan (H+L+S) against three baselines: (2) HULK alone (H), (3) HULK with only LOKI.
B. Tiers of Difficulty

Contours according to Section V-B1. We also train LOKI from observations augmented 8X and centered around detected cable whether the knot class appeared during training:

- **Tier 1**: A single semiplanar knot (figure-eight or overhand); knots of this class were present in the training data, as in [7].
- **Tier 2**: Two semiplanar knots (figure-eight or overhand); knots of this class were present in the training data, as in [7].
- **Tier 3**: Two semiplanar knots (figure-eight or overhand); knots of this class were present in training data and denser than those of [7].
- **Tier 4**: A single non-planar knot (double overhand or square); knots of this class were present in training data.
- **Tier 5**: A single non-planar knot (stevedore, bowline, ashley stopper, granny, or heaving line); knots of this class were not present in training data.

HULK, LOKI, and SPIDERMan were trained on configurations containing up to two overhand and figure-eight knots, single double overhand knots, and single square knots. While Tiers 1-4 contain knots types that appear in the training dataset, Tier 5 tests the generalization capabilities of HULK, LOKI, and SPIDERMan to knot types absent in the training data.

C. Experimental Setup

We execute all experiments using the bilateral dVRK robot equipped with two 7-DoF arms. The dVRK performs untangling of a cut elastic hairtie on a boxed, foam-padded surface to avoid end-effector damage and to prevent the cable from easily leaving the workspace. Given its small dimensions of 5 mm by 15 cm and its flexible material properties, we find this elastic cable to be conducive to manipulation with the dVRK. The workspace is equipped with an overhead Zivid OnePlus RGBD sensor which captures 1900 × 1200 RGBD image observations, although only the RGB channels are used in HULK, LOKI, and SPIDERMan inference. The dVRK is calibrated with a standard pixel-to-world camera transformation using the fiducial registration procedure described in [11]. The complete experimental setup has workspace bounds \( w_l, w_r \). Each grasp is executed with a 30° approach angle to avoid collisions caused by top down grasping.

D. Results

Given the trained networks, we instantiate the proposed policies and baselines and run 12 trials of dVRK cable untangling for each method and each of the four tiers. At the beginning of each trial, a human supervisor manually ties a dense configuration and places it at \( w_c \). Then, the dVRK executes the untangling procedure in Algorithm 1 without intervention. A successful trial is defined as one that terminates single-crossing stable poses due to its elastic material properties. On an Nvidia GeForce RTX 2080, HULK keypoint inference takes 314 ms, LOKI offset and rotation inference takes 260 ms, and SPIDERMan density comparison takes 18 ms. Node Deletion and Reidemeister moves each take 10 s to execute, while Re-Posing and Wedge Recovery moves take 7 s and 15 s to execute respectively.

We report the untangling success rate and median number of Node Deletion moves, Recovery moves (Re-Posing and Wedged Recovery), and total actions per success in Table I. These results suggest that the combination of HULK, LOKI, and SPIDERMan is effective in performing cable untangling, and exhibits higher empirical success with fewer actions than methods that do not jointly leverage all 3 algorithms. Untangling failures are distributed evenly across initial configuration type for Tiers 1, 3, and 4. For Tier 2, all 3 failures are with the Overhand + Figure 8 initial configuration. For Tier 5, all the...
Lastly, we postulate that 12 trials of untangling experiments planar configurations, presenting a harder perceptual task, while the HULK baseline in this paper is trained on non-planar configurations which can be seen in the cables of Figure 5. HULK as reported in [7] is also trained exclusively on semiplanar configurations. Untangling experiments are given a horizon of 5 node deletion actions to make progress. If the cable is equally or more dense after 5 untangling actions, we conclude that the configuration is pathological and terminate the trial.

We observe 5 failure modes:
(A) gripper collision due to poor keypoint and/or grasp rotation predictions in high-density configurations;
(B) robot gripper jaws wedged between cable segments due to high configuration density and/or failed recovery actions;
(C) premature termination due to density comparison false positives in Equation (3);
(D) termination due to lack of untangling progress;
(E) the cable suddenly springing out of the reachable workspace where Re-Posing moves cannot grasp the cable due to poor keypoint predictions and the cable’s elastic material properties.

Across policies that do not leverage SPiDERMan, the most common failure mode is the tendency of the gripper jaws to become wedged between cable segments (B). This failure mode is exacerbated with increasing density and non-planarity, though somewhat alleviated by LOKI’s grasp refinement. We note a 22% decrease in HULK’s performance on identical initial configurations have one failure each, with the exception of the Ashley-Stopper knot, which had 2 failures in 2 trials.

We experimentally evaluate the separate and collective impact of HULK, LOKI, and SPiDERMan on physical untangling of dense, non-planar knots across five difficulty tiers. We find that when combined, HULK, LOKI and SPiDERMan enable success rates of 72.8% and 50% on untangling knots of seen and unseen classes respectively and notably achieve a success rate of 66.7% on untangling non-planar knots not considered in prior work. In future work, we will consider cables of varying visual and material properties and explore how depth sensing or other sensing modalities can improve performance. We will also consider other grasp parametrizations and explore how ideas from LOKI and SPiDERMan can be applied more broadly to other deformable manipulation tasks.

<table>
<thead>
<tr>
<th>Tier</th>
<th>Policy</th>
<th>Success Rate</th>
<th>Node Deletion Actions</th>
<th>Recovery Actions</th>
<th>Total Actions</th>
<th>Failure Modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>H</td>
<td>6/12</td>
<td>6.5</td>
<td>–</td>
<td>6.5</td>
<td>A  3 0 2 0</td>
</tr>
<tr>
<td>1</td>
<td>H+L</td>
<td>6/12</td>
<td>5</td>
<td>–</td>
<td>5</td>
<td>B  2 0 2 0</td>
</tr>
<tr>
<td>1</td>
<td>H+S</td>
<td>6/12</td>
<td>5.5</td>
<td>–</td>
<td>8.5</td>
<td>C  0 0 0 0</td>
</tr>
<tr>
<td>1</td>
<td>H+L+S</td>
<td>6/12</td>
<td>5</td>
<td>0.5</td>
<td>7.5</td>
<td>D  2 0 1 1</td>
</tr>
<tr>
<td>2</td>
<td>H</td>
<td>6/12</td>
<td>4</td>
<td>–</td>
<td>5</td>
<td>E  1 0 1 0</td>
</tr>
<tr>
<td>2</td>
<td>H+L</td>
<td>6/12</td>
<td>5</td>
<td>–</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>H+S</td>
<td>6/12</td>
<td>4.5</td>
<td>1</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>H+L+S</td>
<td>6/12</td>
<td>4</td>
<td>2</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>H</td>
<td>6/12</td>
<td>4</td>
<td>–</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>H+L</td>
<td>6/12</td>
<td>4</td>
<td>–</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>H+S</td>
<td>6/12</td>
<td>4</td>
<td>–</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>H+L+S</td>
<td>6/12</td>
<td>4</td>
<td>2</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>H</td>
<td>6/12</td>
<td>4</td>
<td>–</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>H+L</td>
<td>6/12</td>
<td>4</td>
<td>–</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>H+S</td>
<td>6/12</td>
<td>4</td>
<td>–</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>H+L+S</td>
<td>6/12</td>
<td>4</td>
<td>2</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>H</td>
<td>6/12</td>
<td>4</td>
<td>–</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>H+L</td>
<td>6/12</td>
<td>4</td>
<td>–</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>H+S</td>
<td>6/12</td>
<td>4</td>
<td>–</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>H+L+S</td>
<td>6/12</td>
<td>4</td>
<td>2</td>
<td>6</td>
<td></td>
</tr>
</tbody>
</table>

**TABLE I**

**Physical Results:** Success rate and efficiency (median number of actions per success) for untangling physical cables containing dense non-planar knots on the dVRK. We categorize initial cable configuration complexity into five tiers: (1) one semi-planar knot seen at train time, (2) two semi-planar knots both seen in training, (3) two semi-planar knots both seen in training and denser than those in (2), (4) one non-planar knot seen in training, and (5) one non-planar knot unseen in training. Untangling experiments are given a horizon of 5 node deletion actions to make progress. If the cable is equally or more dense after 5 untangling actions, we conclude that the configuration is pathological and terminate the trial.

**VII. Conclusion**

We present LOKI and SPiDERMan, two algorithms that increase precision and recovery for robot untangling of dense, non-planar knots. When used in conjunction with HULK from prior work, LOKI and SPiDERMan anticipate, refine, and recover from errors due to HULK’s coarse action planning. We experimentally evaluate the separate and collective impact of HULK, LOKI, and SPiDERMan on physical untangling of dense, non-planar knots across five difficulty tiers. We find that when combined, HULK, LOKI and SPiDERMan enable success rates of 72.8% and 50% on untangling knots of seen and unseen classes respectively and notably achieve a success rate of 66.7% on untangling non-planar knots not considered in prior work. In future work, we will consider cables of varying visual and material properties and explore how depth sensing or other sensing modalities can improve performance. We will also consider other grasp parametrizations and explore how ideas from LOKI and SPiDERMan can be applied more broadly to other deformable manipulation tasks.
ACKNOWLEDGMENTS

This research was performed at the AUTOLAB at UC Berkeley in affiliation with the Berkeley AI Research (BAIR) Lab, the Real-Time Secure Execution (RISE) Lab and the CITRIS "People and Robots" (CPAR) Initiative. The authors extend their appreciation to the opinions and recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the sponsors. The authors were supported in part by donations from Toyota Research Institute, Google, and by equipment grants from Intuitive Surgical. The da Vinci Research Kit is supported by the National Science Foundation, via the National Robotics Initiative (NRI), as part of the collaborative research project “Software Framework for Research in Semi-Autonomous Teleoperation” between The Johns Hopkins University (IS 1637789), Worcester Polytechnic Institute (IS 1637759), and the University of Washington (IS 1637444). Ashwin Balakrishna is supported by an NSF GRFP. We thank our colleagues who provided helpful feedback and suggestions, especially Daniel Seita and Rishi Parikh.

REFERENCES