

Sequential Manipulation of Deformable Linear Object Networks with Endpoint Pose Measurements using Adaptive Model Predictive Control

Tyler Toner^{1,3,*}, Vahidreza Molazadeh³, Miguel Saez³, Dawn M. Tilbury^{1,2}, and Kira Barton^{1,2}

Abstract—Robotic manipulation of deformable linear objects (DLOs) is an active area of research, though emerging applications, like automotive wire harness installation, introduce constraints that have not been considered in prior work. Confined workspaces and limited visibility complicate prior assumptions of multi-robot manipulation and direct measurement of DLO configuration (state). This work focuses on single-arm manipulation of stiff DLOs (StDLOs) connected to form a DLO network (DLON), for which the measurements (output) are the endpoint poses of the DLON, which are subject to unknown dynamics during manipulation. To demonstrate feasibility of output-based control without state estimation, direct input-output dynamics are shown to exist by training neural network models on simulated trajectories. Output dynamics are then approximated with polynomials and found to contain well-known rigid body dynamics terms. A composite model consisting of a rigid body model and an online data-driven residual is developed, which predicts output dynamics more accurately than either model alone, and without prior experience with the system. An adaptive model predictive controller is developed with the composite model for DLON manipulation, which completes DLON installation tasks, both in simulation and with a physical automotive wire harness.

I. INTRODUCTION

Manipulation of deformable objects remains an important challenge in robotics. Many objects of interest fall into the category of deformable *linear* objects (DLOs), including ropes [1], surgical sutures, [2], and cables [3]. Their deformability is challenging to accurately model, complicating standard manipulation techniques developed for rigid objects.

Much work has studied the problem of driving a DLO to a particular shape profile or topology [4], [5], [6], [7], [8], [9], [1], [2], sometimes with an emphasis on physical interaction with the environment, such as for knotting [1] or cable routing [10], [11]. In contrast, few works have focused on precisely driving a particular point on a DLO to a goal pose [12], [13], [14]. Additionally, most DLO manipulation work has involved multiple cooperating robot arms. This assumption appears in early geometric planning approaches [2], [1], later work focused on practical industrial implementation [3], [15], [10], and recent approaches focused on learning and control [5], [4], [14], [12]. Few works have studied single-arm DLO manipulation [8], [9], [13], [4].

Consider the automotive wire harness: a bundle of several heterogeneous, relatively stiff cables, each responsible for delivering power or transmitting data inside a vehicle. Each

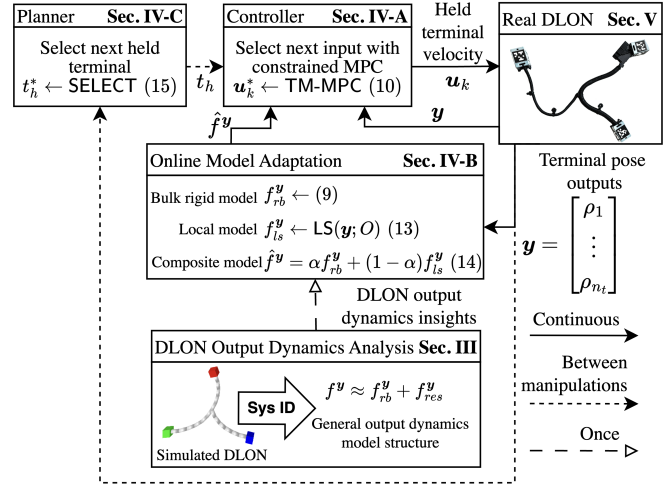


Fig. 1. Framework for robotic installation of deformable linear object networks (DLONs), e.g., automotive wire harnesses, using only terminal pose measurements, or outputs. The controller leverages a composite input-output model containing a DLON-agnostic rigid body model and a DLON-specific local model learned *in situ* without prior learning or online excitation.

cable may be considered an individual DLO with unique mechanical properties, terminating with a specific terminal type. The objective of harness installation is to insert each terminal into its respective receptacle in its environment. Due to its complexity, harness manipulation remains a manual task in industry. This task involves some additional constraints compared to those typically considered in DLO literature.

First, the wire harness system contains a variety of DLOs, but is not itself a DLO. We term this class of objects *DLO networks* (DLONs). Each DLO in a DLON has a connected end and a free end, or *terminal*. Furthermore, each DLO in the DLON is what we denote as a *stiff DLO* (StDLO), meaning that, for fixed terminal poses, the DLOs themselves do not significantly deform, instead roughly returning to an equilibrium configuration, contrasting DLOs such as strings or threads. Additionally, we specifically consider DLOs with predefined rigid grasp points, termed semi-DLOs (SDLOs) in [14], constraining grasps to terminals. Next, the fundamental objective is to manipulate the DLON in such a way as to bring each terminal to a precise pose goal. This contrasts with the majority of prior work concerned with shape or topology goals. Moreover, the shape profile of a DLON may not be measurable. The limited lighting and reflective surfaces encountered during automotive assembly complicate profile detection using contrast-based segmentation [16] or pointcloud processing [5], [17]. Instead, limited observation of only key parts of the DLON, like its terminals, may be

* Tyler Toner is the corresponding author: twtoner@umich.edu

¹Department of Mechanical Engineering, University of Michigan

²Robotics Department, University of Michigan

³General Motors, Research and Development

See videos and more details at sites.google.com/view/robo-harness.

available. For example, marker-based pose tracking may be used [3], [15], [18]. Finally, in many real-world scenarios, coordinated multi-arm manipulation is not practical, being typically realized by carefully constructed programs in static, well-modeled environments [19]. In addition to communication latency and more restrictive self-collision constraints, multi-arm manipulation inside a confined environment like an automotive cabin introduces spatial challenges.

Thus, the problem considered in this work is single-arm manipulation of planar DLONs with terminal pose goals and measurements. A *manipulation* is a robot motion during which the DLON is grasped, and which ends when the grasp is released. A manipulation of the DLON that brings a terminal to its goal is a *terminal manipulation* (TM). Successful installation requires a sequence of TMs that drive the DLON to a state at which all terminals are simultaneously at their goal poses. We do not consider insertion [14], [20], but instead assume a reaction force representing insertion is activated upon a successful TM for simplified analysis.

The work most similar to ours is [13], which studies planar, single-arm DLO manipulation with endpoint measurements and goals. By leveraging physical experiments to generate simulators, open-loop policies are learned to position the free end of a specific DLO by moving the held end. In contrast, we consider sequential manipulation of multi-terminal DLONs with continuous endpoint pose measurements, which we solve with model predictive control (MPC) using an online adaptive model, allowing enforcement of hard constraints while requiring no offline learning.

Our objective is a planning and control framework that can solve feasible DLON installation problems, is robust to initial DLON configurations, and operates on novel DLONs without prior training or system identification. The main contributions of this work are:

- 1) Analysis of planar DLON output dynamics, which admit decomposition into simple rigid body dynamics and residual dynamics,
- 2) Development of a TM planning and control methodology for TM sequence selection and real-time control using an adaptive composite model that enforces feasibility of future TMs, and
- 3) Experimental validation of our proposed approach with both a simulator and a physical wire harness.

Our framework is summarized in Fig. 1. Output dynamics are studied in section III, revealing a decomposition into a rigid body model f_{rb}^y and residual f_{res}^y . An adaptive MPC is designed (section IV-A) for constrained TMs using a composite model (section IV-B) which leverages the decomposition found in section III-C. The TM sequence is planned using a simple, yet effective, heuristic (section IV-C). The planning and control approach is shown to generalize to installation of novel, real DLONs (section V).

II. PROBLEM STATEMENT

A DLON consists of a set of connected stiff DLOs (StDLOs), each terminating with one of n_t terminals. The objective of DLON installation is to bring each terminal to its

respective receptacle. During a terminal manipulation (TM), if a *held* terminal is successfully brought to its receptacle, it is *mated* and is held in place by a reaction force. Other terminals are *free*, and are subject to harness dynamics.

Due to its flexibility, the configuration, or *state*, of a DLON is not finite, but may be approximated by a set of n features, $\mathbf{x} \in \mathbb{R}^n$, for example a set of coordinates along its profile [5], [12], [16]. Furthermore, we only consider planar DLONs, similar to [5], [11], [13], [18]. The pose, $\rho = [x \ y \ \theta]^\top \in SE(2)$ of each terminal with respect to the robot base frame is continuously monitored by a vision system, resulting in a measurement, or *output*, $\mathbf{y} = [\rho_1^\top \ \dots \ \rho_{n_t}^\top]^\top \in SE(2)^{n_t}$.

A manipulator can grasp any terminal within a workspace $W \subset SE(2)$ with a minimum clearance of r_ϵ to the nearest obstacle $o \in O \subset \mathbb{R}^2$. During manipulation, the held terminal is attached to the gripper, which realizes velocity commands $\mathbf{u} = [v_x \ v_y \ \omega_z]^\top \in \mathcal{U}$ with negligible dynamics. The held terminal is directly controlled with $\dot{\rho}_h = \mathbf{u}$, mated terminals are static, $\dot{\rho}_m = 0$, and free terminals are subject to:

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k) \quad (1)$$

$$\mathbf{y}_k = h(\mathbf{x}_k) \quad (2)$$

with discrete time state dynamics f and output map h , both of which are unique to a particular DLON. After each TM, f changes, so neither f nor h are assumed *a priori*.

The problem considered in this work is as follows. **Given** continuous terminal pose output measurements \mathbf{y} of a planar DLON with n_t terminals with goals $\mathbf{g} = [\rho_{1,g}^\top \ \dots \ \rho_{n_t,g}^\top]^\top$ manipulated by a robot within a workspace W with r_ϵ -constrained grasping in the presence of obstacles O , **find** a sequence of terminal manipulations that drive all terminals to their goals, $\mathbf{y} \rightarrow \mathbf{g}$. We make the following assumptions:

- (A1) \exists no explicit constraints on \mathbf{x} , only \mathbf{y} , and
- (A2) over any TM, $\|\dot{\mathbf{x}}\|$ is sufficiently small that the system can be modeled as quasi-static.

III. OUTPUT DYNAMICS

To control a DLON subject to constraints, a dynamic model of its free terminals trajectories in response to held terminal velocity inputs \mathbf{u} is critical. As both f and h are unknown, and the state \mathbf{x} is not available, a prerequisite for output-based control is to determine whether terminal pose output dynamics can be predicted with output \mathbf{y} measurements alone; in other words, direct output dynamics $\mathbf{y}_{k+1} = f^y(\mathbf{y}_k, \mathbf{u}_k)$ must exist. To answer this question, a simulated DLON is used to gather a dataset of \mathbf{x} and \mathbf{y} (section III-A), from which we find an approximate inverse to the output map h (section III-B), indicating the existence of f^y . Output dynamics are approximated by a polynomial model (section III-C), revealing a decomposition later leveraged for an adaptive control-oriented model (section IV-B).

A. DLON simulation and dataset collection

To study DLON behavior, a simulated proxy (Fig. 3) of a real harness (Fig. 4(a)) was developed with the goal of

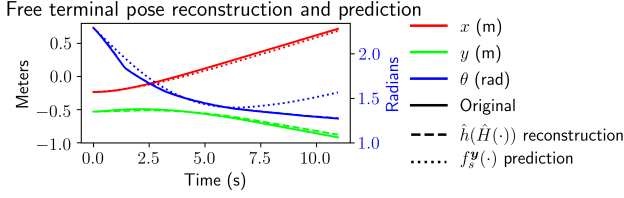


Fig. 2. Representative pose trajectory $\rho_{t,k} = [x_{t,k} \ y_{t,k} \ \theta_{t,k}]^\top$ of one free terminal t , selected from \mathcal{D} (solid lines). The reconstruction of $\rho_{t,k}$ using the output map network and its inverse network, $\hat{h}(\hat{H}(\cdot))$ (dashed lines) is close enough to the original to be mostly obscured. The prediction of $\rho_{t,k}$ from $\rho_{t,0}$ using the polynomial model f_s^y (dotted lines) tracks the original initially, but deviates in θ after several seconds.

emulating its qualitative behavior. A DLON model was generated with three branches, roughly 650 mm in the longest direction, terminating with $n_t = 3$ cuboid terminals with 45 mm side lengths. Flexibility of the intermediate DLOs was achieved by arranging n_L rigid cylindrical links connected by spherical joints (reducing to z axis revolute joints during x - y planar manipulation), each subject to internal damping, constrained angular displacement, and friction.

The DLON was simulated in Pybullet [21], allowing continuous measurement of joint angles θ_j link poses ρ , as well as application of forces and torques at the $F_s = 240$ Hz simulation rate. Controllers were developed to generate forces and torques that track reference position and velocity commands. In each simulation, terminal $t = 0$ was held and commanded at desired velocities \mathbf{u} . To explore the state space, the DLON was excited with constant \mathbf{u} commands, uniformly sampled from the admissible control space \mathcal{U} , for 15 seconds. The DLON was initialized with the same initial conditions, but the beginning of each trajectory was removed in order to ensure the initial state of each trajectory is unique.

Trajectories of θ_j and $\mathbf{y} = [\rho_0^\top \ \rho_1^\top \ \rho_2^\top]^\top$ were recorded and filtered, with velocities $\dot{\mathbf{y}}$ estimated using central differences. Each recording was downsampled to a realistic $F_c = 30$ Hz. With $\mathbf{x} = [\rho_0^\top \ \theta_1 \ \dots \ \theta_{n_L}]^\top$, the uniform-length trajectories were formed into the dataset:

$$\mathcal{D} = \{ \{ \mathbf{x}_k, \mathbf{y}_k, \dot{\mathbf{y}}_k, \mathbf{u}_k \}_{k=0}^{N_s} \}. \quad (3)$$

B. Output invertibility

Although DLONs vary in terms of structure, stiffness, and other physical properties, we qualitatively observe from real and simulated systems that wire harnesses have unique stable states, the defining characteristic of a StDLOs. For a fixed set of terminal poses $\bar{\mathbf{y}}$, the DLON tends to return to a unique state $\bar{\mathbf{x}}$ when external forces along the DLON are removed. Stable, minimum energy configurations of DLOs have been studied in other work and serve as a useful analytical simplification [2]. We hypothesize that there exists

$$H : SE(2)^{n_t} \rightarrow \mathcal{B}(\mathbf{x}, \epsilon_x) \subset \mathbb{R}^n \quad (4)$$

that approximates the inverse of h , such that

$$\|h(H(\mathbf{y})) - \mathbf{y}\| < \epsilon_y \quad (5)$$

TABLE I
MEAN MODEL PREDICTION ERRORS

Model error metric	Number of parameters	Translational Error (m)	Rotational Error (rad)
$\ \hat{h}(\mathbf{x}) - \mathbf{y}\ $	5631	1.50×10^{-3}	1.53×10^{-3}
$\ \hat{H}(\mathbf{y}) - \mathbf{x}\ $	9939	N/A	5.19×10^{-2}
$\ \hat{h}(\hat{H}(\mathbf{y})) - \mathbf{y}\ $	N/A	2.08×10^{-3}	5.28×10^{-3}
$\ f_s^y(\mathbf{y}_k) - \mathbf{y}_{k+1}\ $	110	3.57×10^{-4}	1.09×10^{-3}

where $\mathcal{B}(\mathbf{x}, \epsilon_x)$ is a ball centered at \mathbf{x} with radius ϵ_x , and $\epsilon_x, \epsilon_y > 0$ are thresholds of acceptable precision. To evaluate this hypothesis, we utilize the dataset \mathcal{D} gathered in section III-A to learn neural network approximations \hat{h} and \hat{H} using a standard supervised learning approach.

To aid training, outputs were altered to contain poses of free terminals relative to the held terminal and the rotation re-encoded as $\mathbf{y} = [({}^0\rho_1^{sc})^\top \ ({}^0\rho_2^{sc})^\top]^\top$ with $\rho^{sc} = [x \ y \ \sin(\theta) \ \cos(\theta)]^\top$. A 70%-15%-15% dataset split was used to separate training, validation, and testing data. Multi-layer perceptrons were used for both \hat{h} and \hat{H} . Optuna [22] was used for hyperparameter tuning with minimization of L_2 loss on the validation set as the tuning objective.

Error metrics of the final networks are shown on the test dataset in Table I, which indicate \hat{h} and \hat{H} have approximated both the output map and its inverse successfully. The output reconstruction error $\|\hat{h}(\hat{H}(\mathbf{y})) - \mathbf{y}\|$ is likewise small, indicating that \hat{H} approximates the output of \hat{h} as desired. Moreover, applying the two networks to reconstruct a representative trajectory illustrates the fit (Fig. 2).

Substituting (1) into (2), we have $\mathbf{y}_{k+1} = h(f(\mathbf{x}_k, \mathbf{u}_k))$. Supposing that H exists, it follows that:

$$\mathbf{y}_{k+1} \approx h(f(H(\mathbf{y}_k), \mathbf{u}_k)) = f^y(\mathbf{y}_k, \mathbf{u}_k), \quad (6)$$

indicating the existence of direct input-output dynamics f^y , allowing output trajectories to be uniquely predicted without knowledge of the underlying state.

C. Sparse identification of output dynamics

Next, we seek an interpretable model of f^y capable of providing insight into the structure of DLON output dynamics, and turn to the sparse identification of nonlinear dynamics (SINDy) technique [23], [24]. Consider a library of nonlinear function candidates $\Theta(\mathbf{y}, \mathbf{u})$, in which each column represents a nonlinear function of \mathbf{y} and \mathbf{u} . For a continuous time system of the form $\dot{\mathbf{y}} = f^y(\mathbf{y}, \mathbf{u})$, SINDy seeks an approximation $f^y(\mathbf{y}, \mathbf{u}) \approx f_s^y(\mathbf{y}, \mathbf{u}) = \Theta(\mathbf{y}, \mathbf{u})\Xi$ containing as few terms as possible. To find Ξ^* , solve:

$$\min_{\Xi} \sum_{\mathcal{D}} \|\dot{\mathbf{y}}_k - \Theta(\mathbf{y}_k, \mathbf{u}_k)\Xi\|_2^2 + \lambda \|\Xi\|_1 \text{ s.t. } \min |\Xi| \geq T \quad (7)$$

where $\lambda > 0$ promotes a sparse solution and T is the minimum coefficient threshold to avoid miniscule terms.

The PySINDy package [25] was used to solve (7). A simple polynomial library on \mathbf{y}, \mathbf{u} terms with maximum degree 2 was utilized. Once again, the dataset \mathcal{D} was split and Optuna was utilized to select optimal λ, T while solving

TABLE II
MAXIMUM TRAJECTORY PREDICTION ERROR (MEAN \pm STD)

Model	Translational Error (m)	Rotational Error (rad)
Rigid Body B_{rb} (9)	0.065 ± 0.102	0.278 ± 0.532
Least Squares B_{ls} (13)	0.043 ± 0.037	0.174 ± 0.202
Composite \hat{B} (14)	0.036 ± 0.049	0.156 ± 0.293

(7) over the training set, in order to minimize prediction error on the validation set. The resulting continuous time model $f_s^y(\mathbf{y}, \mathbf{u}) = \Theta(\mathbf{y}, \mathbf{u})\Xi^*$ was discretized by assuming a zero-order hold over the time step Δt : $f_s^y(\mathbf{y}, \mathbf{u}) = \mathbf{y} + \Delta t f_s^y(\mathbf{y}, \mathbf{u})$. An optimal model with only 110 terms was found using $\lambda = 0.957$, $T = 0.108$.

The polynomial model f_s^y was evaluated in terms of one-step prediction accuracy in Table I and trajectory prediction performance over 11 seconds in Fig. 2. f_s^y predicts translational dynamics well, with reduced performance in the rotational dimension. Despite the imperfect fit, the polynomial model is interpretable; upon inspection, we observe that it can be decomposed into two distinct components:

$$f_s^y(\mathbf{y}, \mathbf{u}) = C f_{rb}^y(\mathbf{y}, \mathbf{u}) + f_{res}^y(\mathbf{y}, \mathbf{u}) \quad (8)$$

where $f_{rb}^y(\mathbf{y}, \mathbf{u})$ are rigid body dynamics with diagonal $C \succ 0$ and $f_{res}^y(\mathbf{y}, \mathbf{u})$ are residual terms. Rigid body dynamics between the held terminal h and a terminal t are given by:

$$f_{rb}^{\rho t}(\dot{\rho}_t, \mathbf{u}) = B_{rb}(\rho_t)\mathbf{u} = \begin{bmatrix} 1 & 0 & y_h - y_t \\ 0 & 1 & x_t - x_h \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix} \quad (9)$$

which can be stacked for: $f_{rb}^y = [f_{rb}^{\rho_1} \dots f_{rb}^{\rho_n}]^T$. Note that (9) is a classical equation for rigid body motion in a plane, and is not specific to any particular DLON.

To evaluate how well rigid body dynamics f_{rb}^y predict the true continuous-time dynamics f^y , both were computed over all samples in the dataset \mathcal{D} , with $f^y(\mathbf{y}_k, \mathbf{u}_k) \approx \dot{\mathbf{y}}_k$, and the coefficient of determination, or R^2 , was computed between all values of f_{rb}^y and $\dot{\mathbf{y}}_k$. Averaging over both free terminals, translational and rotational R^2 values were found to be 0.972 and 0.477, respectively, indicating that rigid body dynamics (9) are significantly predictive of translational dynamics and moderately predictive of rotational dynamics.

Although f and h are unknown, output dynamics were found to be well-approximated by a polynomial model containing significant rigid body motion. Bulk output dynamics, available *a priori* for any DLON, may be predicted as rigid, with comparatively minor DLON-specific residuals. These results are based on the a single simulated DLON dataset, but we see the resulting adaptive model derived from these modeling insights (section IV-B) generalizes to a physical DLON (section 4). Limitations are discussed in section VI.

IV. PLANNING AND CONTROL

To realize DLON installation, a methodology for planning TM sequences and robustly executing them is developed. For

TM execution, an adaptive model predictive control (MPC) approach with appropriate constraints is given in section IV-A, with online model adaptation covered in section IV-B, which leverages insights from section III-C. Finally, the TM sequence planning algorithm is given in section IV-C.

A. Model predictive control

DLON installation is sequential: each TM must be executed such that future TMs remain feasible, so free terminals must not be driven to ungraspable regions, either due to obstacle proximity or by exiting the robot workspace. As found in section III, simple models predict output dynamics well over short horizons, so we propose an adaptive MPC:

$$\begin{aligned} \text{TM-MPC}(\mathbf{y}_0; \mathbf{g}, O) : \quad & \min_{\mathbf{u}_0, \dots, \mathbf{u}_{N-1}} \sum_{j=0}^{N-1} l(\mathbf{y}_j, \mathbf{u}_j) + l_t(\mathbf{y}_N) \\ & \text{s.t. } \forall j = 0, \dots, N \\ & \mathbf{y}_{j+1} = \hat{f}^y(\mathbf{y}_j, \mathbf{u}_j) \\ & c(\mathbf{y}_j) \leq 0 \\ & \mathbf{y}_j \in \mathcal{Y}, \mathbf{u}_j \in \mathcal{U} \end{aligned} \quad (10)$$

where (10) is solved over a horizon of N future steps to optimize a stage cost l and terminal cost l_t while satisfying constraints: the system dynamics, inequality constraints, and set memberships. Dynamics \hat{f}^y are continuously updated online with a procedure covered later in section IV-B. At each time step of a TM, output \mathbf{y}_k is taken as \mathbf{y}_0 for TM-MPC (10) and the first optimal input $\mathbf{u}_k = \mathbf{u}_0^*$ is executed. We define an $SE(2)$ metric with rotational weight β as:

$$\mathbf{d}_\beta(\rho_1, \rho_2) = \left\| \begin{bmatrix} x_1 - x_2 \\ y_1 - y_2 \end{bmatrix} \right\|_2^2 + \beta \left\| \begin{bmatrix} \cos(\theta_1) - \cos(\theta_2) \\ \sin(\theta_1) - \sin(\theta_2) \end{bmatrix} \right\|_2^2. \quad (11)$$

Then, we define the costs as

$$\begin{aligned} l(\mathbf{y}_k, \mathbf{u}_k) &= \mathbf{d}_\beta(\rho_{h,k}, \rho_{h,g}) + \mathbf{u}_k^\top Q \mathbf{u}_k + \Delta \mathbf{u}_k^\top Q_\Delta \Delta \mathbf{u}_k \\ l_t(\mathbf{y}_N) &= p \mathbf{d}_\beta(\rho_{h,N}, \rho_{h,g}) \end{aligned} \quad (12)$$

where $Q, Q_\Delta \succ 0$ and $p \geq 0$ define relative weights on control cost, control increment cost, and terminal cost. The goal pose $\rho_{t,g}$ of each terminal is computed from the pose of its respective receptacle, ρ_r and a predefined insertion offset $r \rho_{t,g}$, both of which are static and known.

Ensuring future TM feasibility can be conservatively achieved by enforcing free TM graspability at all times. Containment in the reachable workspace is encoded by defining $\mathcal{Y} = W^{n_t}$. A graspability-preserving clearance of at least r_ϵ is achieved with $c(\mathbf{y})$ by stacking $r_{t,o} - \mathbf{d}_0(\rho_{t,k}, \rho_o)$ terms for all obstacles o and terminals t . The safety radius $r_{t,o} = r_o + r_t + r_\epsilon$ is defined by circumscribing all terminals and obstacles with circles of radii r_t and r_o , respectively.

B. Online model adaptation

To estimate \hat{f}^y , a common approach is to find a local linear model $\dot{\mathbf{y}} = \hat{f}_{ls}^y(\mathbf{u}) = B_{ls}\mathbf{u}$ using least squares:

$$\text{LS}(\dot{\mathbf{Y}}, \mathbf{U}) : \min_{B_{ls}} \sum_{j=1}^{N_e} \|\dot{\mathbf{y}}_{k-j} - B_{ls}\mathbf{u}_{k-j}\|^2 \quad (13)$$

where $\dot{\mathbf{Y}} \in \mathbb{R}^{N_e \times 3n_t}$ and $\mathbf{U} \in \mathbb{R}^{N_e \times 3}$ are data matrices containing the last N_e samples of $\dot{\mathbf{y}}_k$ and \mathbf{u}_k , respectively.

Necessary conditions for a meaningful solution to (13) are given in [16] for DLO shape control, including delayed application of (13) until all directions of \mathbf{u}_k are present over N_e , without which B_{ls} cannot predict dynamics in new directions. As a result, controllers must explore all directions of \mathbf{u} to collect sufficient local data, delaying the primary task.

In section III-C, rigid body dynamics (9), which assume $\dot{\mathbf{y}} = B_{rb}(\mathbf{y})\mathbf{u}$, were found to account for a substantial portion of total dynamics. Such a model can predict bulk dynamics in any direction without delay, but does not model more complex local dynamics. To capture the benefits of both approaches and enable instantaneous control without exploration while still learning local dynamics, we propose a residual model: $\hat{f}^y(\mathbf{y}_k, \mathbf{u}_k) = \mathbf{y}_k + \Delta t \hat{B}(\mathbf{y}_k)\mathbf{u}_k$, where

$$\hat{B}(\mathbf{y}_k) = \alpha_k B_{rb}(\mathbf{y}_k) + (1 - \alpha_k) B_{ls} \quad (14)$$

and $\alpha_k \in [0, 1]$ indicates the degree to which the rigid body model is favored. To enable immediate control without excitation while gradually incorporating learned local dynamics, let $\alpha_{k+1} = \gamma\alpha_k$ with $\gamma \in (0, 1)$ and $\alpha_0 = 1$.

To validate (14), for each trajectory in \mathcal{D} we compare B_{rb} , B_{ls} , and \hat{B} . With $N_e = F_c$ (estimate over last 1 second), we predict the next 5 seconds ($5F_c$ steps); maximum prediction error statistics are shown in Table II, illustrating that the composite model outperforms each constituent model alone. Moreover, the composite model can be deployed without system excitation to immediately apply TM-MPC (10).

C. Sequence planner

While TM-MPC executes a TM, the TM sequence must be planned. We opt for a simple, but effective, heuristic that selects the terminal closest to constraint violation. Among a set of n_f free terminals, the next held terminal is selected with (15). Finally, the entire Install-DLON procedure including planning and control is given in Algorithm 1.

$$\begin{aligned} \text{SELECT}(\mathbf{y}; O) : \min_{t \in [0, n_f]} \min_{o \in O} (r_{t,o} - \mathbf{d}_0(\rho_t, \rho_o)) \\ \text{s.t. } \min_{o \in O} (r_{t,o} - \mathbf{d}_0(\rho_t, \rho_o)) > 0 \quad (15) \\ \rho_t \in W. \end{aligned}$$

V. EXPERIMENTS AND DISCUSSION

A. Simulated experiments

The DLON model used for dataset collection (section III-A) was likewise used for experiments. Recall that our approach does not involve pretraining, so experiments do not benefit from the dataset. Four problems were considered (Fig. 3), in which each terminal of the DLON is to be mated

TABLE III
SIMULATION PROBLEM CONSTRAINT $c(\mathbf{y})$ SATISFACTION

Problem	Model Type	Maximum $c(\mathbf{y})$	Final $c(\mathbf{y})$
easy	Rigid Body	-0.210	-0.302
	Composite	-0.210	-0.305
rotated	Rigid Body	-0.136	-0.264
	Composite	-0.143	-0.290
obstacles	Rigid Body	Did not finish	
	Composite	0.031	-0.015
wall	Rigid Body	0.012	-0.049
	Composite	0.017	-0.115

to the receptacle of the same color. Install-DLON was applied to each problem with both the rigid body model (9) and the composite model (14), with resulting performance metrics in Table III: maximum and final constraint values between TMs, where positive values represent constraint violations.

Problems *easy* and *rotated* involve an obstacle-free workspace. Constraints are respected with both models, though constraints are farther from violation with the composite model. In *obstacles*, obstacles create a narrowing through which the DLON must maneuver. With the rigid model, the DLON becomes stuck and does not finish. With the composite model, a constraint violation is observed, but the problem is solved. In *wall*, the DLON is initialized behind a wall of obstacles; *wall* is solved with both models, though constraint violations occur that are later recovered.

B. Physical experiments

To evaluate generalization of Install-DLON and the composite model (14), studies were conducted on a physical system, shown in Fig. 4(a). A modification of the setup used in our prior work [18], a real automotive wire harness with three terminals, approximately 750 mm in the longest direction, is selected. Blocks of sidelength 45 mm are added at the end of each terminal to ease grasping with a 6-DOF UR5 manipulator with a parallel jaw gripper. AprilTag

Algorithm 1 Install-DLON(terms. $\{t\}$, goals g , obstacles O)

```

1: while True do ▷ Planning loop
2:   Observe output  $\mathbf{y}$ 
3:    $t_h \leftarrow \text{SELECT}(\mathbf{y}; O)$  (15)
4:   if no solution to SELECT (15) then
5:     break ▷ no feasible TMs remaining
6:   Grasp terminal  $t_h$  with manipulator
7:   Initialize  $\alpha_0 = 1$ ,  $\dot{\mathbf{Y}} \leftarrow \emptyset$ ,  $\mathbf{U} \leftarrow \emptyset$ 
8:   for  $k \in [0, \bar{N}]$  do ▷ Control loop w/ max. dur.  $\bar{N}$ 
9:     Observe output  $\mathbf{y}_k$ 
10:    Estimate  $B_{ls} \leftarrow \text{LS}(\dot{\mathbf{Y}}, \mathbf{U})$  (13)
11:    Update model  $\hat{B} \leftarrow \alpha_k B_{rb}(\mathbf{y}_k) + (1 - \alpha_k) B_{ls}$  (14)
12:    Execute  $\mathbf{u}_k \leftarrow \text{TM-MPC}(\mathbf{y}_k; g, O)$  (10)
13:    if  $\mathbf{d}_\beta(\rho_{h,k}, \rho_{k,g}) < \epsilon_g$  then
14:      break ▷ TM complete
15:    Discount  $\alpha_{k+1} \leftarrow \gamma\alpha_k$ 
16:    Estimate  $\dot{\mathbf{y}}_k$  and update LS data matrices  $\dot{\mathbf{Y}}$  and  $\mathbf{U}$ 
17:    Release  $t_h$  and return manipulator to home configuration

```

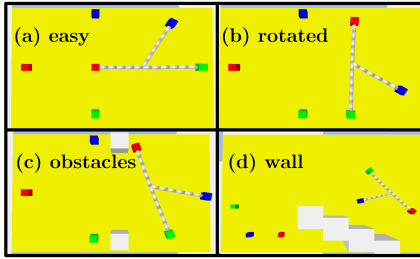


Fig. 3. Simulated DLO experiments. Each terminal must be brought to the static receptacle of the same color without leaving the yellow workspace. In (a)-(b), the only obstacles are the receptacles themselves. In (c), the DLO must fit through a narrowing created by obstacles (white). In (d), the DLO must navigate around a blocking wall of obstacles.

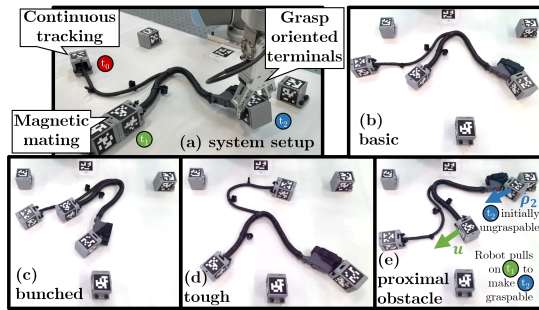


Fig. 4. (a) Physical setup with 6-DOF manipulator and modified 3-terminal automotive wire harness. (b-e) Physical experiments with harness configurations in increasing order of challenge. In (b)-(d), terminals are initialized away from obstacles. In (e), terminal t_2 is initialized near a receptacle, preventing grasping; the robot grasps and uses it to pull the violating terminal away from the obstacle before continuing installation.

markers [26] on terminals and receptacles allow continuous pose tracking with an external Intel Realsense L515 camera.

The harness was placed in four initial configurations, shown in Fig. 4(b-e). In each case, Install-DLO was applied with the composite model, and installation was successful. In the `proximal-obstacle` problem, terminal t_2 is placed very close to a receptacle, preventing grasping. The robot selects t_1 to hold, which it uses to pull t_2 out of constraint violation before moving on to the insertion of t_1 , thus ensuring feasibility of a later t_2 insertion.

C. Discussion of results

The experimental results indicate the suitability of our planning and control approach for solving the DLO installation problem, as defined in section II. MPC naturally allows incorporation of the constraints required by TMs, and the main challenge becomes selection of a sufficiently accurate dynamics model for the MPC. Simulations indicate that the adaptive composite model outperforms the rigid body model in terms of predicting constraint violation, where the latter fails entirely in one problem. The composite model, which was originally developed with insights from simulation data, generalizes to a new, physical DLO, enabling prediction of free terminal dynamics suitable for effective MPC.

Our proposed approach is capable of manipulating a DLO to endpoint goals without monitoring its internal state, and reveals that the complex, data-driven global models

[16] developed for DLO state control may not be necessary for endpoint control. Compared to DLO endpoint control policies learned offline from real data [13], our composite model predicts free terminal trajectories sufficiently accurately for control purposes without offline learning.

VI. CONCLUSIONS AND FUTURE WORK

In this work, planar manipulation of DLO networks (DLONs), collections of connected stiff DLOs (StDLOs), is studied. Focus is directed toward DLONs with measured terminal poses (output) but unmeasured configurations (state). The existence of output dynamics without explicit state dependence is demonstrated in simulation. The output dynamics are approximated, revealing significant rigid body motion. This structural insight leads to an adaptive composite model, consisting of a simple rigid body model and a continuously estimated local linear model, which is used by an MPC for constrained, single-robot DLON terminal manipulation (TM). A simple TM sequence planner based on constraint proximity is proposed. The efficacy of our framework is demonstrated in simulated and physical experiments.

This study demonstrates the feasibility of output-based control on a class of planar DLON problems, but practical problems are more complex. Assumption (A1) prohibits state constraints, preventing the framework from being deployed in environments with obstacles likely to collide with the DLOs during a TM. Moreover, although predefined grasp points may be necessary to prevent DLON damage, restricting grasps to terminals is limiting. Future work towards explicit state estimation would relax (A1) and expand candidate grasp poses, enabling more complex manipulation sequences.

Although our method is not constrained in principle by DLON complexity, it was only validated on 3-terminal DLONs with similar characteristics. Future work is needed to validate our models and understand the limitations of our planning and control approach. In particular, understanding the range of physical properties that qualify as an StDLO, such that the output map is approximately invertible, is critical when only sparse terminal pose measurements are available. Additionally, extending the approach to non-planar DLONs will be necessary for some applications, such as harness installation within an automotive door assembly.

As harness installation is a repetitive task, future work will investigate how data from repeated DLON installations could be leveraged to build more complex models from real data *in situ* without deliberate excitation, as in [5], [13], to better inform both control-oriented models and high-level plans. Such an approach could use our local, DLON-agnostic composite model to enable initial installations without *a priori* data, while allowing performance gains over time from more accurate global models.

ACKNOWLEDGEMENT

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. DGE-1841052. Any opinion, findings, and conclusions or recommendations expressed in this material are those of the authors(s) and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

- [1] M. Saha and P. Isto, "Manipulation Planning for Deformable Linear Objects," *IEEE Transactions on Robotics*, vol. 23, pp. 1141–1150, Dec. 2007.
- [2] M. Moll and L. Kavraki, "Path planning for deformable linear objects," *IEEE Transactions on Robotics*, vol. 22, pp. 625–636, Aug. 2006.
- [3] X. Jiang, K.-m. Koo, K. Kikuchi, A. Konno, and M. Uchiyama, "Robotized Assembly of a Wire Harness in Car Production Line," in *International Conference on Intelligent Robots and Systems*, 2010.
- [4] N. Lv, J. Liu, and Y. Jia, "Dynamic Modeling and Control of Deformable Linear Objects for Single-Arm and Dual-Arm Robot Manipulations," *IEEE Transactions on Robotics*, vol. 38, pp. 2341–2353, Aug. 2022.
- [5] 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, pp. 5544–5551, Apr. 2022.
- [6] Y. Wu, W. Yan, T. Kurutach, L. Pinto, and P. Abbeel, "Learning to manipulate deformable objects without demonstrations," in *16th Robotics: Science and Systems, RSS 2020*, MIT Press Journals, 2020.
- [7] 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.
- [8] H. Han, G. Paul, and T. Matsubara, "Model-based reinforcement learning approach for deformable linear object manipulation," in *2017 13th IEEE Conference on Automation Science and Engineering (CASE)*, (Xi'an), pp. 750–755, IEEE, Aug. 2017.
- [9] N. Alvarez and K. Yamazaki, "An interactive simulator for deformable linear objects manipulation planning," in *2016 IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAN)*, (San Francisco, CA, USA), pp. 259–267, IEEE, Dec. 2016.
- [10] P. M. Fresnillo, S. Vasudevan, and W. M. Mohammed, "An approach for the bimanual manipulation of a deformable linear object using a dual-arm industrial robot: cable routing use case," in *2022 IEEE 5th International Conference on Industrial Cyber-Physical Systems (ICPS)*, (Coventry, United Kingdom), pp. 1–8, IEEE, May 2022.
- [11] J. Zhu, B. Navarro, R. Passama, P. Fraisse, A. Crosnier, and A. Cherubini, "Robotic Manipulation Planning for Shaping Deformable Linear Objects With Environmental Contacts," *IEEE Robotics and Automation Letters*, vol. 5, pp. 16–23, Jan. 2020.
- [12] M. Yu, H. Zhong, F. Zhong, and X. Li, "Adaptive control for robotic manipulation of deformable linear objects with offline and online learning of unknown models," *arXiv e-prints*, pp. arXiv–2107, 2021.
- [13] V. Lim, H. Huang, L. Y. Chen, J. Wang, J. Ichnowski, D. Seita, M. Laskey, and K. Goldberg, "Real2sim2real: Self-supervised learning of physical single-step dynamic actions for planar robot casting," in *2022 International Conference on Robotics and Automation (ICRA)*, pp. 8282–8289, 2022.
- [14] H. Zhou, S. Li, Q. Lu, and J. Qian, "A Practical Solution to Deformable Linear Object Manipulation: A Case Study on Cable Harness Connection," in *2020 5th International Conference on Advanced Robotics and Mechatronics (ICARM)*, (Shenzhen, China), pp. 329–333, IEEE, Dec. 2020.
- [15] K. M. Koo, X. Jiang, K. Kikuchi, A. Konno, and M. Uchiyama, "Development of a robot car wiring system," in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM*, pp. 862–867, 2008.
- [16] J. Zhu, B. Navarro, P. Fraisse, A. Crosnier, and A. Cherubini, "Dual-arm robotic manipulation of flexible cables," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 479–484, IEEE.
- [17] T. P. Nguyen and J. Yoon, "A novel vision-based method for 3D profile extraction of wire harness in robotized assembly process," *Journal of Manufacturing Systems*, vol. 61, pp. 365–374, oct 2021.
- [18] T. Toner, M. Saez, D. M. Tilbury, and K. Barton, "Opportunities and challenges in applying reinforcement learning to robotic manipulation: an industrial case study," *Manufacturing Letters*, 2023.
- [19] M. Saez and P. Spicer, "Robot-to-robot collaboration for fixtureless assembly: Challenges and opportunities in the automotive industry," in *ASME 2020 15th International Manufacturing Science and Engineering Conference, MSEC 2020*, vol. 2, pp. 8–10, 2020.
- [20] T. Z. Zhao, J. Luo, O. Sushkov, R. Pevceciciute, N. Heess, J. Scholz, S. Schaal, and S. Levine, "Offline Meta-Reinforcement Learning for Industrial Insertion," in *International Conference on Robotics and Automation (ICRA)*, 2022.
- [21] E. Coumans and Y. Bai, "Pybullet, a python module for physics simulation for games, robotics and machine learning." <http://pybullet.org>, 2016–2019.
- [22] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A Next-generation Hyperparameter Optimization Framework," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 2623–2631, 2019.
- [23] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," *Proceedings of the National Academy of Sciences*, vol. 113, pp. 3932–3937, Apr. 2016.
- [24] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Sparse Identification of Nonlinear Dynamics with Control (SINDYc)," *IFAC-PapersOnLine*, vol. 49, no. 18, pp. 710–715, 2016.
- [25] B. de Silva, K. Champion, M. Quade, J.-C. Loiseau, J. Kutz, and S. Brunton, "Pysindy: A python package for the sparse identification of nonlinear dynamical systems from data," *The Journal of Open Source Software*, vol. 5, no. 49, p. 2104, 2020.
- [26] J. Wang and E. Olson, "AprilTag 2: Efficient and robust fiducial detection," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4193–4198, IEEE, Oct 2016.