

# Robotic Software Architecture for In-Space Outfitting Operations

Amy M. Quartaro\*

*FASER Lab at Virginia Tech, Blacksburg, VA, 24060, United States*

John R. Cooper†

*NASA Langley Research Center, Hampton, VA, 23666, United States*

Erik E. Komendera‡

*FASER Lab at Virginia Tech, Blacksburg, VA, 24060, United States*

Space exploration is expanding into longer missions, larger payloads, and more complex operations. To make these larger scale missions a reality, it is necessary to perform assembly, construction, and maintenance tasks via a robotic workforce in addition to crewed operations. While there has been significant research into in-space assembly and manufacturing, it is primarily focused on rigid structural elements, such as ISRU printing or truss construction. Outfitting tasks, such as cable routing, are a critical step to a fully operational in-space facility. This paper seeks to provide a reduced order state model and an optimized combination of state-of-the-art robotics algorithms applied to a cable routing scenario. Simulation results are expected to advance approaches to online autonomous robotic manipulation of non-rigid elements.

## I. Nomenclature

|                    |   |                                                                                                                               |
|--------------------|---|-------------------------------------------------------------------------------------------------------------------------------|
| $g$                | = | gravity constant $\left[\frac{m}{s^2}\right]$                                                                                 |
| $k_\theta$         | = | rotational (torsional) spring constant $\left[\frac{N}{rad}\right]$                                                           |
| $k_l$              | = | linear spring constant $\left[\frac{N}{m}\right]$                                                                             |
| $l$                | = | total length of cable $[m]$                                                                                                   |
| $m$                | = | total mass of cable $[kg]$                                                                                                    |
| $m_{link}$         | = | mass of single link $[kg]$                                                                                                    |
| $n$                | = | number of link segments per cable                                                                                             |
| $\mathbf{p}_i$     | = | position vector $\in \mathbb{R}^3$                                                                                            |
| $\mathbf{p}_{des}$ | = | desired end point of cable                                                                                                    |
| $q$                | = | optimization variables                                                                                                        |
| $\mathbf{R}_j^i$   | = | $SO(3)$ rotation matrix $\in \mathbb{R}^{3 \times 3}$ , the orientation of frame $j$ relative to the frame $i$                |
| $\mathbf{T}_j^i$   | = | $SE(3)$ transformation matrix $\in \mathbb{R}^{4 \times 4}$ , position and orientation of frame $j$ relative to the frame $i$ |
| $U$                | = | potential energy $[J]$                                                                                                        |
| $\theta$           | = | relative rotation between links $[rad]$                                                                                       |
| $\delta x$         | = | deformation of linear spring $[m]$                                                                                            |

## II. Introduction

THIS paper proposes a modeling and control architecture to perform in-space outfitting tasks, a challenging and complex problem area necessary for sustained presence in space. As the space industry moves towards establishing a long-term presence on planetary surfaces, such as the Moon and Mars, the need for a robust robotic workforce is

---

\*FASER Graduate Research Assistant, Virginia Tech Mechanical Engineering Dept, and AIAA Student Member, Contact: aquartaro@vt.edu

†Aerospace Engineer, Autonomous Integrated Systems Research Branch, Non-Member

‡Assistant Professor and FASER Lab Director, Virginia Tech Mechanical Engineering Dept, and AIAA Member



**Fig. 1 In-space significance of DLOs on ISS (left, credit: NASA) [10] and Europa Clipper spacecraft (right, credit: Johns Hopkins APL/Ed Whitman)[11]**

critical [1]. Robotic operations are crucial to the longevity and efficiency of large, complex space missions such as the NASA Gateway [2], a large-aperture assembled space telescope [3], or a constructed long-term settlement on a planetary surface [4, 5]. There has historically been significant interest in utilizing robotic capabilities for in-space assembly and maintenance of rigid structures [6, 7]. However, robotic manipulation of objects required for outfitting tasks, such as cabling/tubing required for power/life support systems, insulation and radiation shielding, and any other tasks required for a functional facility after the rigid structure is complete, have historically not received the same attention. One class of objects seen often in outfitting tasks includes a wide variety of non-rigid continuous lines, such as cables, wires, harnesses, etc., referred to as deformable linear objects (DLOs). DLO components are critical to a fully functional in-space facility, demonstrated by the applications in Fig. 1, but have not garnered significant research and focus for robotic manipulation until very recently. The architecture proposed here focuses on the manipulation and routing of cable harnesses.

Maneuvering and securing cables for any engineering project requires a proper understanding of the cable and the surrounding environment to ensure alignment, avoid collisions and snags, and have the final assembly efficiently placed relative to the environment. These constraints are a challenge in any application; however, space systems include many additional requirements and complications critical to safe operations [8, 9]. Any robotic operations for cable routing and other outfitting tasks require a system capable of tracking many state objects to safely maneuver the cable, while also rapidly adapting the motion plan if the cable or surrounding environment behaves unpredictably.

The purpose of this paper is to further research into DLO manipulation for in-space outfitting applications, with a focus on cable routing, considering some of the unique situations that arise in an in-space environment. Low gravity, large temperature changes, and communication latencies with operators are all challenges to in-space cable outfitting tasks. There is significant need for a near real-time robot control architecture that addresses challenges specific to in-space applications for robotic construction and maintenance to be a viable component of the space exploration portfolio. A software architecture is proposed, hereafter referred to as DLO Manipulation Architecture for Cluttered Environments (DLO-MACE), and the foundational multibody model used to represent a cable is described in detail.

### **A. Robotic In-Space Assembly and Construction**

Robotic technologies applied to assembly, manufacturing, and construction for space exploration is a well established discipline and recognized as a necessary technology area for a persistent presence in-space and on extraplanetary bodies. In addition to architecture level concepts outlining the need for constructed facilities [3–5], significant effort has gone into establishing viability of robotic operations for in-space assembly and construction. Robotic methods have been demonstrated to be a viable method for in-space assembly, primarily manipulating specialty design structural

components or modules [12]. There are robotic agents specifically developed to aide in assembly and construction of in-space facilities, with a specific focus on autonomous robotics, to expand the capabilities of an in space facility [13–15]. Utilizing robotic systems for In-Situ Resource Utilization (ISRU) construction is an additional area of development [16]. Heterogenous robotic agents are a promising area for robotic assembly and construction, enabling a long reach manipulator to maneuver a dexterous manipulator and construction materials into a specific environment for assembly or maintenance operations [17]. Robotic agents have also been investigated for their ability to inspect and verify assembled structures [18].

While the research into autonomous robotic assembly of structural elements has seen significant effort, the outfitting of such assembled structures is more limited - inhibiting technology for robotic implementation of a fully functional facility. Without a robotic architecture to outfit an assembled structure, there is limited power, communication, and life support that can be provided. There has been some investigation into how to perform outfitting on windows or ports while building space habitats with additive manufacturing [19, 20].

## B. DLO Manipulation

General manipulation and modeling of DLOs has seen significant interest in recent years, spanning a wide variety of disciplines including computer graphics, flexible robotic joints, and continuum robotic systems in addition to DLO manipulation by a traditional rigid robotic agent. Computer graphics and animation routinely model flexible components such as trees, hair, and mechanical cables, but can have limitations in rendering time and accuracy [21, 22]. Flexible robotic joints fall in between a fully defined robot with six or less degrees of freedom (DOF) and a continuum system that approaches infinite DOF. Introducing flexibility into robotic joints, even if the rest of the robot remains rigid, can better represent a physical mechanical joint [23, 24]. Continuum robotics utilizes the DLO itself as the robotic manipulator, and there have been a variety of mechanical and modeling approaches to manipulating a highly flexible robot [25, 26]. Robotic manipulation of DLOs has seen a recent surge in research endeavors, with applications including household/medical settings, towing welding cables and identification via image segmentation being explored [27–30].

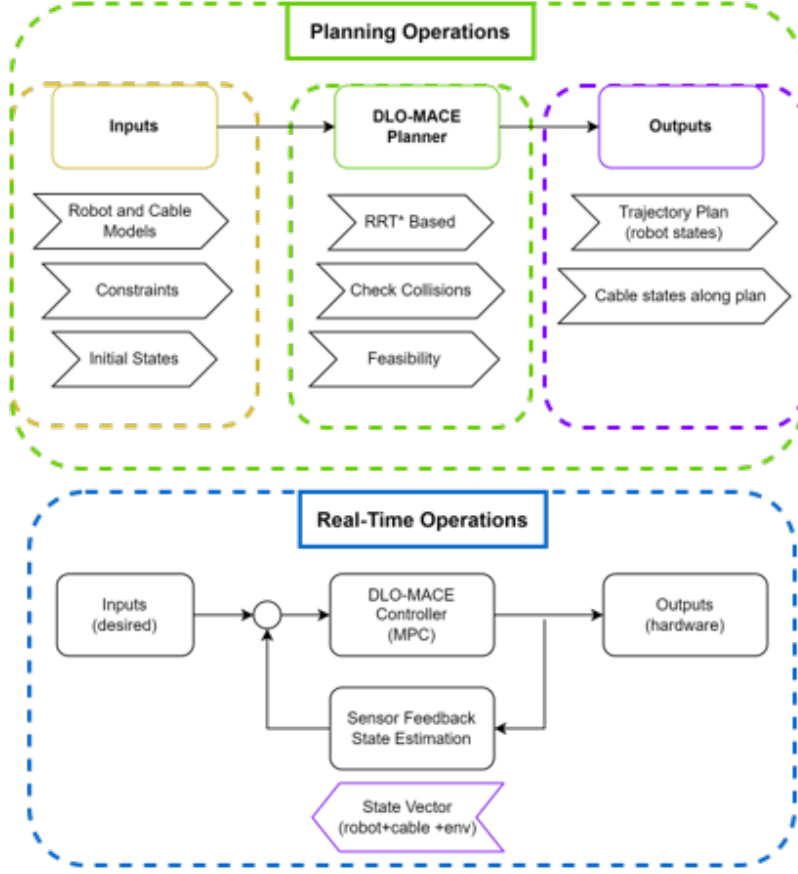
There are a few commonly used model-based methods for DLOs, each with varying accuracy and computational complexity [21]. One approach is to treat the DLO as a serialized multibody model, constructing the DLO similar to a physically realizable spring-damper system in discrete segments. The multibody approach can be limited in accuracy (increasing accuracy for an increasing number of segments), but is relatively simple in computational complexity [28, 31]. Another model for DLOs is the Cosserat Rod model, which incorporates more mechanical properties and evaluates the curve across a centerline (often determined using B-Spline or Bezier curves) [22, 32]. A spline approach has also been utilized in force based applications [33]. A final DLO modeling technique is a data-driven approach, utilizing Neural-Networks to generate an understanding of DLO behavior during manipulation [34].

The proposed method here utilizes a multibody based approach, treating the DLO as a serially connected chain of links that is parametrically built to give the designer control over accuracy and size of the state space (computational complexity). Upcoming work from the authors will explore comparisons between different modeling methods and the trade-off between model accuracy and complexity. The cable model plays a significant role in the DLO-MACE architecture, as the robotic agent must understand how the cable will react to motion in order to plan and execute routing tasks. Describing the cable model is therefore a significant portion of this paper.

## III. Simulation Architecture

Fundamentally, the proposed software architecture consists of predetermined kinematic and dynamic models for all components in the workspace; the cable to be manipulated, the structural environment to be routed through, and the robot agent to be controlled. DLO-MACE utilizes optimized models for the robot and cable (DLO object) to perform planning and real-time operation in an obstacle dense space. Such a framework must track a high number of states in order to perform planning and control operations. Consequently, DLO-MACE requires a minimum-state representation of the environment, which can be done utilizing knowledge from optimization and structural analysis disciplines. Minimized in this sense refers to an optimal state vector length that trades model accuracy with computational complexity. Using a minimized state vector with optimized path planning and control schemes for driving the robot, near real-time cable routing operations will become achievable via autonomous robotic agents.

The remainder of this section outlines elements of the overall DLO-MACE framework, and how the various components can be integrated and increased in complexity as the architecture is implemented. A brief outline of how the various components of DLO-MACE fit together is illustrated in Fig. 2 Details about the internal cable model that drives the various elements of DLO-MACE is described in Section IV. The results described in this paper serve to



**Fig. 2 DLO-MACE Overview**

verify the static cable model, enabling quasi-static motion of the robot, and does not implement the full architecture stack at this time.

### A. Planning Operations

For the initial implementation of DLO-MACE, path planning will be done offline prior to running a real-time simulation using well established methods for near-optimal planners such as the optimal Rapidly-exploring Random Trees (RRT\*) or Probabilistic RoadMap (PRM\*) algorithms and previous work in DLO path planning [32]. Such a sampling based planner can be used to both identify a feasible path and optimize the trajectory in the feasible space based on the cable model, robot model, and state of the environment.

The next step in complexity after evaluating only a cable and robot is to introduce obstacles in the environment, as is likely to be seen in a real world DLO routing application. Another addition to the planning scheme that is likely to appear in a real-world application is additional constraints that require the cable to pass through certain fixture point along a structure - such as to restrain the cable along the routing distance or to place fixture points along the route. By adding obstacles and additional constraints, the difficulty of the robot's path planning increases significantly as there are now more constraints on both the robot motion and the motion/reaction of the cable as it's pulled along, in addition to the increased size of the state vector. Path planning and obstacle avoidance must be understood in order for the DLO-MACE design to trade off accuracy and complexity - a less accurate model means the required collision mesh must be larger and may not be feasible in a cluttered environment. By incorporating dynamic and kinematic models of scenario components with a high-fidelity offline ideal path, robotic manipulation of DLOs in highly unpredictable environments will move closer to being autonomously viable for complex in-space missions.

## B. Real-Time Operations

After a feasible plan has been generated and there is a desired trajectory for the robot to move along and route the cable, it is necessary to incorporate real-time control systems and state estimation techniques in order to safely perform the cable routing procedure. Utilizing a control scheme such as Model Predictive Control (MPC) is a viable method for control as an optimization problem for the cable is already known, the planned model horizon is obtainable [35]. MPC is a demonstrated control scheme for DLO manipulation [34].

Common state estimation methods rely on large sampling sizes for near optimal results (Particle Filter) or are heavily tuned to match a system and deteriorate for increasingly nonlinear systems (Extended Kalman Filter). Algorithms such as FastSLAM or GraphSLAM are to be explored for the cable routing scenario to increase the speed and accuracy of the cable and robot model, as well as low-sample nonlinear filters [36] [37].

Possible sensing paradigms for DLO-MACE include leveraging vision systems to control the robotic manipulator, taking advantage of methods proposed for DLO tracking including fiducial markers at fixed points. Recent work in advancing visual techniques for DLO tracking could further increase confidence in the estimate from visual fiducials [29, 38]. If the design of the cable can be included, the use of fiber optic shape sensing would significantly increase the accuracy of the DLO-MACE sensor input and reduce the computational load of relying on an image stream [39].

## IV. Problem Definition

The specific outfitting task explored in this paper is a cable routing task. While more complex simulation tasks can be achieved in the future, the current model consists of a single cable, manipulated by a single robot, in a defined environment. For the DLO-MACE verification simulation, the robotic agent is treated as a high-fidelity serial manipulator, similar to a UR10e arm such as the one shown in Fig. 3, with encoders on each revolute joint and end effector [40]. Such a serial arm was chosen to align with research test environments and current state-of-the-art for robotic DLO manipulation. The cable modeled in this scenario is considered to be a uniform cylinder with constant diameter. The robot forward and inverse kinematics throughout the scenario will utilize a screw theory formulation to map between end-effector position and joint positions.

The  $SE(3)$  representation of position and orientation is used throughout this model to represent both the robot and cable in the same manner. The  $SE(3)$  transformation matrix  $T_j^i \in \mathbb{R}^{4 \times 4}$  contains the position and orientation of frame  $j$  relative to frame  $i$  and is equivalent to Eq. 1. The superscript frame  $i$  is often the spatial (global) frame, and is implied when no superscript is present throughout this section.

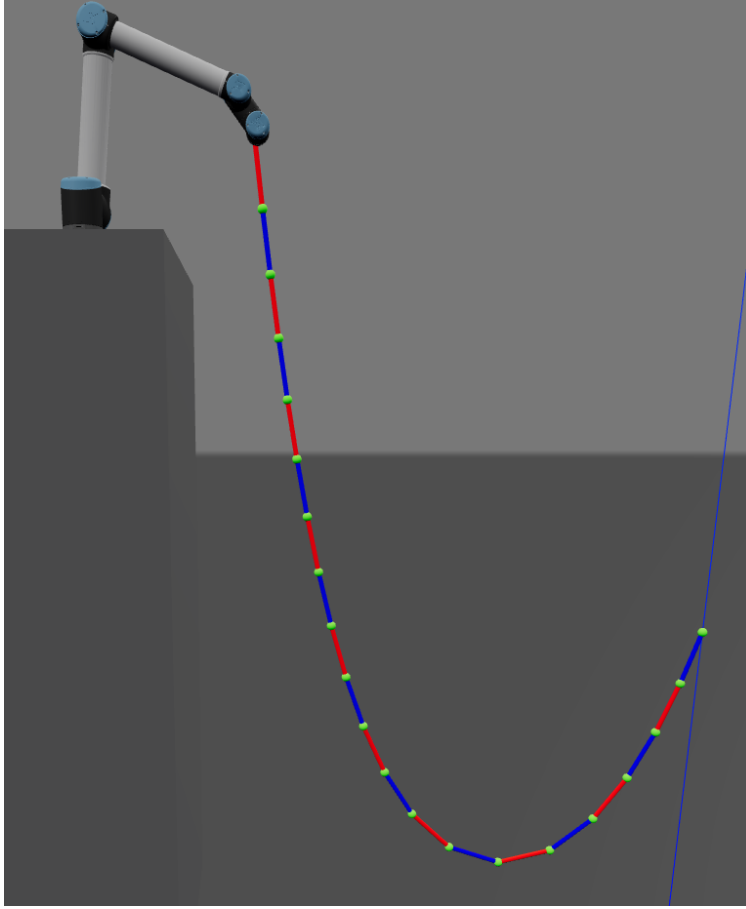
$$\mathbf{T}_j^i = \begin{bmatrix} \mathbf{R}_j^i & \mathbf{p}_{ij} \\ 0 & 1 \end{bmatrix} \quad (1)$$

To model the robot kinematics and dynamics, and to visualize them in a clear way, standard models using available robotics Python libraries and commercially available robot models are used [40, 41]. The orientation and position of the robot end-effector is represented with a transformation matrix, which is then used as the end position constraint of the cable model generation.

### A. Cable Model

The initial construction of the DLO-MACE scenario investigates a cable being manipulated by a serial robot in an open environment (no structural obstacles to route around) as shown in Fig. 3. Such a simple case allows for evaluation of the cable and robot models as a near-ideal manipulation case, a necessary step before incorporating more complex operations. Even the simple case in operation can have complications, such as turning too quickly and putting too much stress on the cable, or an unmodeled mode in the cable's behavior causing diverging vibrations that the robot must attempt to stabilize. A parametric approach to the simulation allows for the size of the state vector to be swiftly modified during model development to compare the number of tracked points on the cable with performance and computational requirements of a DLO-MACE instance.

Figure 3 illustrates the initial problem setup, where the cable is broken up into  $n$  link segments. Each link has uniform mass, determined by the overall cable mass and the number of links ( $m_{link} = \frac{m}{n}$ ) and consists of two nodes, a base and forward node, at either end of a rigid length; nodes are represented as green dots in Fig. 3. The forward node of the  $i^{th}$  link connects to the base node of the  $(i + 1)^{th}$  link. This connection is treated as a multibody joint with the relative position and orientation dictated by rotational and linear springs and dampers, a similar approach as in [28] and [31]. Figure 4 illustrates the defined springs and frame locations. Additionally, there is a  $(n + 1)$  spring between the end



**Fig. 3 Initial Problem Setup**

of the final link segment and the cable end position to provide symmetry in the model. The formulation described in this paper solves the static case, and therefore only the springs are considered. In a dynamic system, there are nonzero velocities and dampers further defining model behavior.

### 1. Minimum Potential Energy

By minimizing the potential energy of a given system, the static configuration of a cable that satisfies given constraints can be determined. This section describes the mathematical formulation and optimization problem required to generate a static cable configuration utilizing linear and rotational springs. The problem outlined here is formulated for planar motions for simplicity in the derivation and explanations, however the rotation and transformation matrices used are expressed in 3D cartesian space to emphasize the anticipated expansion into a 3D potential energy model. Only one rotational spring is evaluated at this time, with ongoing work to incorporate the 2 additional rotation axes. The global coordinate frame utilized is *y-up*, and the direction of gravity is in the  $-y$  direction. Additionally, the rotation matrix notation used throughout this model description is  $\mathbf{R}(\theta_i)$  and indicates a rotation matrix with a rotation  $\theta_i$  about the  $z$  axis. The rotation matrix then simplifies to Eq. 2.

$$\mathbf{R}(\theta_i) = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 \\ \sin \theta_i & \cos \theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

The spring deformation values are the independent variables in this problem, and can be combined into a single optimization variable vector  $\mathbf{q}$ , shown in Eq. 3, which represents the generalized coordinates of the multibody system. The spring deformation variables  $\theta_i$  and  $\delta x_i$  directly represent the relative rotation and translation between the  $(i - 1)$

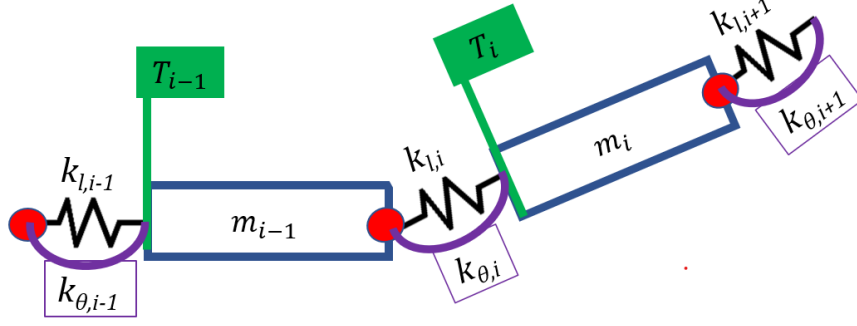


Fig. 4 Two link definition example with torsional and linear springs

and  $i^{th}$  nodes, respectively, as can be seen in Fig. 4. The linear deformation  $\delta x_i$  can be treated as the magnitude of the translation vector that connects the nodes.

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_1 & \dots & \mathbf{q}_n \end{bmatrix} \quad \text{where } \mathbf{q}_i = \begin{bmatrix} \theta_i & \delta x_i \end{bmatrix} \text{ for } i = 1 \dots n+1 \quad (3)$$

The optimization problem is formulated using total cable potential energy as the cost function to be minimized, and includes potential energy terms for torsional springs, linear springs, and gravity at each link's center of mass (COM) and is a scalar value as seen in Eq. 4.

$$U(\mathbf{q}) = \sum_{i=1}^n \frac{1}{2} k_r \theta_i^2 + \frac{1}{2} k_l \delta x_i^2 + m_{link} g y_i(\mathbf{q}) \quad (4)$$

The height of link  $i$  in the global frame,  $y_i$ , is a function of the preceding and current spring deformations and is found through the relationship in Eq. 5.

$$y_i(\mathbf{q}) = \mathbf{e}_2 \mathbf{T}_i(\mathbf{q}) \mathbf{T}_{COM} \mathbf{e}_4 \quad (5)$$

The basis vector  $\mathbf{e}_i \in \mathbb{R}^4$  indicates all zeros except in the  $i^{th}$  entry ranging from 1 to 4.  $\mathbf{T}_i$  is the transformation of the base node of link  $i$  with respect to the global frame, as defined in Eq. 6. The location of the base node and  $\mathbf{T}_i$  for a given link is shown in Fig. 4.

$$\mathbf{T}_i = \mathbf{T}_{i-1} \mathbf{T}_{link} \mathbf{T}_{spr,i}^{link}(\theta_i, \delta x_i) \text{ for } i = 2 \dots n+1 \quad (6)$$

As the first link is connected to the origin and not a previous link,  $\mathbf{T}_1 = \mathbf{T}_{spr,i}(\theta_i, \delta x_i)$ . The constant transformation matrices are defined in Eq. 7.  $\mathbf{T}_{COM}$  is the transformation matrix representation of the translation across a link from its base node to the link COM given the cable length  $l$  and number of link segments  $n$  in the cable.  $\mathbf{T}_{link}$  is the transformation that connects the two nodes of a link segment, from base node to forward node, creating a rigid body with mass  $m_{link}$  as indicated in Fig. 4. The link segments defined here are considered to be identical, with uniform mass distributions across the length of the cable. Therefore, the constant matrices in Eq. 7 are tied to the definition of the cable and do not change between links, and as such the  $i$  term is left off the notation to emphasize this uniformity. The superscript from the  $\mathbf{T}$  notation is also left off as these transformation matrices to further emphasize these frames as operators and not treated as frames in the model by themselves.

$$\mathbf{T}_{COM} = \begin{bmatrix} \mathbf{I}_3 & \begin{bmatrix} 0.5 \frac{l}{n} \\ \mathbf{0} \end{bmatrix} \\ \mathbf{0} & 1 \end{bmatrix} \quad \mathbf{T}_{link} = \begin{bmatrix} \mathbf{I}_3 & \begin{bmatrix} \frac{l}{n} \\ \mathbf{0} \end{bmatrix} \\ \mathbf{0} & 1 \end{bmatrix} \quad (7)$$

The transformation matrix for a given spring relative to some given frame  $a$  is as defined in Eq. 8.

$$\mathbf{T}_{spr,i}^a = \begin{bmatrix} \mathbf{R}(\theta_i) & \mathbf{R}(\theta_i) \begin{bmatrix} \delta x_i \\ \mathbf{0} \end{bmatrix} \\ \mathbf{0} & 1 \end{bmatrix} \quad (8)$$

The position is rotated internally because  $\mathbf{T}_{spr,i}^a$  is *from* frame  $a$  *to* the spring frame - the linear deformation of the spring would not be applied axially without first applying the rotation of the spring.

The desired end points of the cable are given as in Eq. 9, and are the driving constraints in solving for the cable configuration.

$$\mathbf{p}_0 = \mathbf{0}_3 \quad \mathbf{p}_n = \mathbf{T}_n(\mathbf{q})\mathbf{T}_{link}\mathbf{T}_{spr,n+1}^{link} \quad \mathbf{p}_{des} = \begin{bmatrix} x_{des} & y_{des} & 0 \end{bmatrix} \quad (9)$$

Utilizing the definitions above to describe the static goal configuration and contributions to potential energy consolidated into a single cost function, an optimization problem can be described as in Eq. 10. Sequential Quadratic Programming (SQP) via the Sequential Least Squares Programming ('SLSQP') solver in SciPy is used to solve this optimization problem [42, 43].

$$\begin{aligned} \min_{\mathbf{q}} f(\mathbf{q}) &= \sum_{i=0}^n \frac{1}{2} k_r \theta_i^2 + \frac{1}{2} k_l \delta x_i^2 + m_{link} g y_i(\mathbf{q}) \\ \text{subject to } g(\mathbf{q}) &= |\mathbf{p}_n - \mathbf{p}_{des}| \leq \mathbf{0} \\ h(\mathbf{q}) &= \mathbf{p}_0 = \mathbf{0} \end{aligned} \quad (10)$$

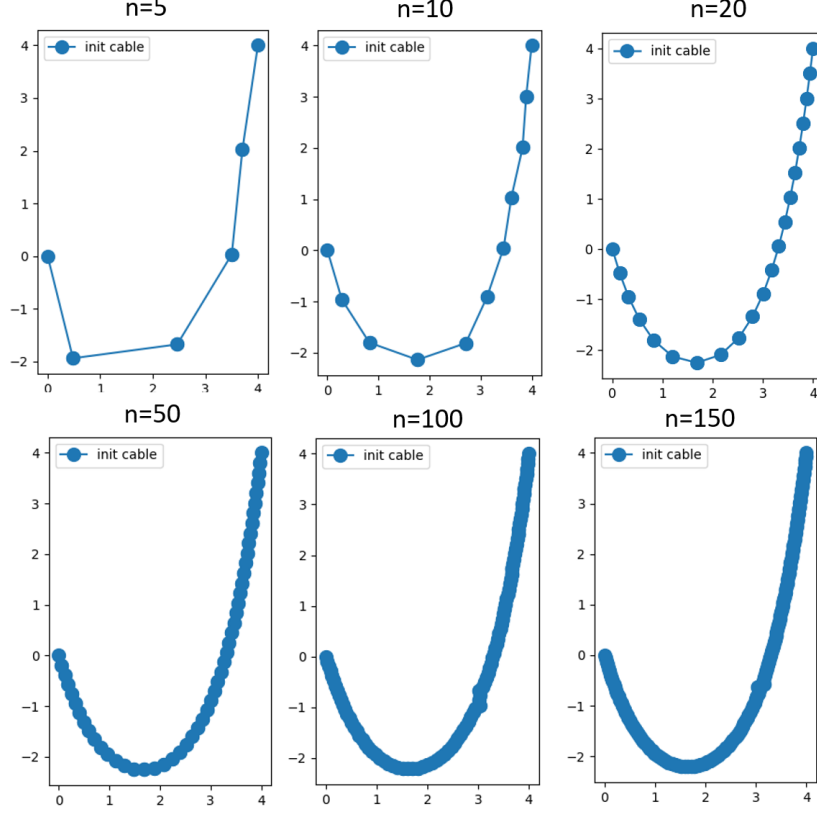
The inequality constraint is the absolute value between the calculated end point and the desired end point, and is formatted in such a way as to assist the chosen solver, which has some hidden functionality due to using an external library. As the complexity of DLO-MACE increases, additional constraints to resolve collision checking and cable waypoints will be added.

Most nonlinear solvers straight out of the package require the cost functions, constraint functions, and optimization variables and utilize numerical methods to approximate the gradient of the cost and constraint functions for converging onto a solution. The gradient is necessary because nonlinear solvers such as SLSQP sequentially try to solve the Karush–Kuhn–Tucker (KKT) conditions and iterate towards  $\nabla_{\mathbf{q}} f(\mathbf{q}) + \lambda \nabla_{\mathbf{q}} g(\mathbf{q}) + \nu \nabla_{\mathbf{q}} h(\mathbf{q}) = \mathbf{0}$ . During the implementation of the aforementioned cable model, it was found that the serial connectivity of the links would not converge properly with the numerical gradient approximation using finite difference methods. An analytical solution for the gradient of the cost,  $(\nabla_{\mathbf{q}} f(\mathbf{q}))$  and constraint  $(\nabla_{\mathbf{q}} g(\mathbf{q}), \nabla_{\mathbf{q}} h(\mathbf{q}))$  functions has been implemented in the planar manipulation case and leads to a smoother and more consistent static cable configuration. The gradients of the optimization problem require calculation and are outlined for a single link segment in Eq. 11. The gradients follow a similar form for each  $\mathbf{q}_i$  as defined in Eq. 3. The gradient of  $h(\mathbf{q})$  is all zeros, due to the origin point not being dependent on any of the optimization variables.

$$\begin{aligned} \nabla_{\mathbf{q}_i} f(\mathbf{q}) &= \begin{bmatrix} k_r \theta_i + m_{link} g \sum_{j=i}^n \frac{\partial}{\partial \theta_i} y_j & k_l \delta x_i + m_{link} g \sum_{j=i}^n \frac{\partial}{\partial \delta x_i} y_j \end{bmatrix} \\ \nabla_{\mathbf{q}_i} g(\mathbf{q}) &= \nabla \left[ \mathbf{T}_n \mathbf{T}_{link} \mathbf{T}_{spr,n+1}^{link} (\theta_i, \delta x_i) \mathbf{e}_4 \right] = \nabla \mathbf{T}_{end}(\theta_i, \delta x_i) \mathbf{e}_4 \\ &= \begin{bmatrix} \frac{\partial}{\partial \theta_i} \mathbf{T}_{end} \mathbf{e}_4 & \frac{\partial}{\partial \delta x_i} \mathbf{T}_{end} \mathbf{e}_4 \end{bmatrix} \end{aligned} \quad (11)$$

The gradients, built using partial derivatives of the transformation matrices for both  $\mathbf{p}_n$  and  $y_i$ , has one term in the serially linked transformation matrices dependent on any given variable set  $q_i$ . The partial derivatives can therefore be obtained via iteration, as illustrated in Eq. 12.

$$\begin{aligned} \frac{\partial}{\partial \theta_1} \mathbf{T}_{end} &= \left( \frac{\partial}{\partial \theta_1} \mathbf{T}_{spr,1} \right) \mathbf{T}_{link} \mathbf{T}_{spr,2}^{link} \dots \mathbf{T}_{spr,n+1}^{link} \\ \frac{\partial}{\partial \theta_2} \mathbf{T}_{end} &= \mathbf{T}_{spr,1} \mathbf{T}_{link} \left( \frac{\partial}{\partial \theta_2} \mathbf{T}_{spr,2}^{link} \right) \dots \mathbf{T}_{spr,n+1}^{link} \\ \frac{\partial}{\partial \theta_1} y_i &= \mathbf{e}_2 \left( \frac{\partial}{\partial \theta_1} \mathbf{T}_{spr,1} \right) \mathbf{T}_{link} \mathbf{T}_{spr,2}^{link} \dots \mathbf{T}_{spr,i}^{link} \mathbf{T}_{COM} \mathbf{e}_4 \\ \frac{\partial}{\partial \theta_2} y_i &= \mathbf{e}_2 \mathbf{T}_{spr,1} \mathbf{T}_{link} \left( \frac{\partial}{\partial \theta_2} \mathbf{T}_{spr,2}^{link} \right) \dots \mathbf{T}_{spr,i}^{link} \mathbf{T}_{COM} \mathbf{e}_4 \end{aligned} \quad (12)$$



**Fig. 5** Static Cable results for varying number of links; horizontal and vertical axes represent unit distances for the  $x$  and  $y$  directions relative to the global frame, respectively.

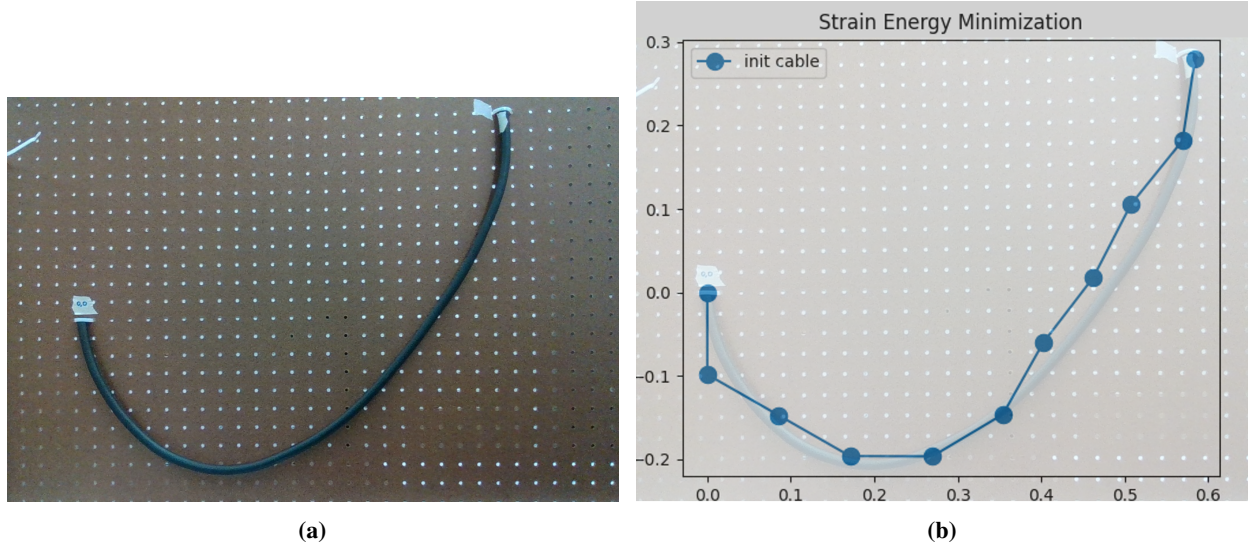
## V. Model Implementation

The cable model detailed in Section IV.A has been implemented in simulation, verifying the internal math and parametric nature of the model formulation. The simulation results can also be qualitatively compared to a hardware experiment with a fixed position planar cable.

### A. Simulation Results

Simulations to demonstrate the parametric nature of the multibody cable model were conducted to evaluate model behavior and change mechanical properties such as axial and bending stiffness, mass, and length in addition to the number of nodes. Fig. 5 illustrates the results of the static optimization problem for different discretization levels (number of link segments). These simulation results illustrate the changes in the predicted cable shape based on the number of links present, given identical constraints and material properties. Utilizing the static solution as a starting point, a quasi-static case can be evaluated at fixed time steps, and/or the cable dynamics can be applied. It is also possible to use a low-link version of the cable for an initial guess, and then increase the fidelity to improve the accuracy of the model (for static or dynamic solutions).

Evaluating the simple case outlined here, there is a point where increasing the number of nodes in the model (moving towards a more continuous system) does not significantly change the shape of the overall cable. In the current model formulation, there is opportunity for local minima to appear in the form of small kinks in the cable, as is the case in high  $n$  cases in Fig. 5. In a cable routing scenario, it may not be necessary to provide the most information to the highest fidelity possible, but rather utilize a lower number of links, therefore reducing the state vector necessary to track the cable in the environment.



**Fig. 6 (a) Planar hardware experiment, (b) simulation with low node count ( $n=10$ ) overlaid on hardware test stand**

## B. Hardware Experiments

The simulation results can be compared to a hardware system, with system ID procedures to determine the spring constants if not available for a specific DLO element. Figure 6b overlays the results of a static hardware experiment overlaid with the simulation results after preliminary model tuning for a discretization of 10 links. This experiment also demonstrates the additional constraints on the cable model, constraining the orientation of the starting node in addition to position to match the specific scenario, demonstrated by the vertical first node in Fig. 6b. The hardware experiment in Fig. 6 consists of a standard off-the-shelf cable mounted to a pegboard with known, measurable end points to align with the model simulation. Further work is needed to fully isolate the most accurate spring constant parameters, a task achievable with machine learning methods and large data sets.

## VI. Conclusion

This paper presents an architecture to incorporate robotic outfitting into space assembly and construction operations, leveraging prior work in robotic in-space assembly and DLO manipulation. The overall framework is described and proposed components of the architecture that utilize a multibody approach for modeling cable elements and with an emphasis on the trade off between model accuracy and computational complexity. As indicated in the comparison between the simulation data and the hardware test setup, depending on application a system may not need a high complexity to achieve the desired routing configuration.

Incorporating a data-driven model to generate the ideal parameter set for the spring stiffnesses, trained on cables generated using high-fidelity simulation techniques in addition to hardware data is a promising avenue to optimally model DLOs with a reduced state space. Utilizing an additional machine learning approach allows for the DLO-MACE model to train with an understanding of the parametrically discretized multibody DLO model, and generate a model for planning and control that best trades off model accuracy and state space size (based on number of links in cable). While Cosserat rods are common in the area of DLO manipulation due to their accuracy, for real-time operations in cluttered environments the number of states and required computations is more significant when there is a feasible solution. Comparing the parametric multibody model to a Cosserat Rod approach will indicate if the increased complexity of the rod model would be worth the trade for accuracy and responsiveness. The next steps in developing the DLO-MACE architecture include implementing dynamics into the cable model, and evaluating how the different methods of path planning, control, and state estimation perform with the DLO and cluttered environment scenarios. Each tool will need to be developed and tested independently before the full DLO-MACE framework can come to fruition.

As the DLO-MACE architecture matures, there are a variety of milestones that will contribute to the advancement of in-space robotic construction and maintenance. Treating cable routing as an optimization problem, with constraints

on cable bend angle, maximum stress, and clearance from hazardous features without a strict rigid body assumption allows for outfitting tasks for both in-space and terrestrial applications to be completed robotically. Cable manipulation, among other outfitting tasks, is an essential part of facility construction and maintenance. Robotic assembly of such facilities will not be viable until the challenging area of outfitting tasks is investigated and discussed.

## VII. Acknowledgments

This work was supported by a NASA Space Technology Graduate Research Opportunity (80NSSC21K1286).

## References

- [1] NASA, “MOON TO MARS OBJECTIVES,” Sep. 2022. URL <https://www.nasa.gov/sites/default/files/atoms/files/m2m-objectives-exec-summary.pdf>.
- [2] NASA, “Gateway Overview,” accessed 3/22/2023, 2022. URL <https://www.nasa.gov/gateway/overview>, <https://www.nasa.gov/gateway/overview>.
- [3] She, Y., Li, S., Liu, Y., and Cao, M., “In-orbit robotic assembly mission design and planning to construct a large space telescope,” Vol. 6, No. 01, 2020, p. 1. <https://doi.org/10.1117/1.jatis.6.1.017002>.
- [4] Waltz, W. J., Grande, M. L., and Moses, R. W., “Autonomous System Operations for Lunar Safe Haven Establishment and Sustainment,” *AIAA SCITECH 2022 Forum*, American Institute of Aeronautics and Astronautics, 2022. <https://doi.org/10.2514/6.2022-2075>.
- [5] Govindaraj, S., Gancet, J., Urbina, D., Brinkmann, W., Aouf, N., Lacroix, S., Wolski, M., Colmenero, F., Walshe, M., Ortega, C., and Bodo, B., “PRO-ACT: Planetary Robots Deployed For Assembly And Construction Of Future Lunar ISRU And Supporting Infrastructures,” 2019. URL [https://www.researchgate.net/publication/344809638\\_PRO-ACT\\_Planetary\\_Robots\\_Deployed\\_For\\_Assembly\\_And\\_Construction\\_Of\\_Future\\_Lunar\\_ISRU\\_And\\_Supporting\\_Infrastructures](https://www.researchgate.net/publication/344809638_PRO-ACT_Planetary_Robots_Deployed_For_Assembly_And_Construction_Of_Future_Lunar_ISRU_And_Supporting_Infrastructures).
- [6] Rembala, R., and Ower, C., “Robotic assembly and maintenance of future space stations based on the ISS mission operations experience,” *Acta Astronautica*, Vol. 65, No. 7-8, 2009, pp. 912–920. <https://doi.org/10.1016/j.actaastro.2009.03.064>.
- [7] Doggett, W., Dorsey, J., Jones, T., Mikulas, M., Teter, J., and Paddock, D., “TriTruss: A New and Novel Structural Concept Enabling Modular Space Telescopes and Space Platforms,” *70th International Astronautical Congress (IAC)*, 2019.
- [8] NASA, “Spotlight on Lessons Learned: Cable Harness Wiring and Connector Anomalies Caused by Induced Damage in Human Spaceflight Vehicles,” Accessed 3/13/2023, Jan. 2022. URL <https://appel.nasa.gov/2022/01/31/spotlight-on-lessons-learned-cable-harness-wiring-and-connector-anomalies-caused-by-induced-damage-in-human-spaceflight-vehicles/>, <https://appel.nasa.gov/2022/01/31/spotlight-on-lessons-learned-cable-harness-wiring-and-connector-anomalies-caused-by-induced-damage-in-human-spaceflight-vehicles/>.
- [9] NASA, “CABLE AND HARNESS GENERAL REQUIREMENTS,” Accessed 3/13/2023, Apr. 2002. URL <https://workmanship.nasa.gov/lib/insp/2books/links/sections/401GeneralRequirements.html>, <https://workmanship.nasa.gov/lib/insp/2>
- [10] NASA, “ISS National Laboratory,” Mar. 2023. URL [https://www.nasa.gov/mission\\_pages/station/research/nlab/index.html](https://www.nasa.gov/mission_pages/station/research/nlab/index.html), accessed 22 March 2023.
- [11] NASA, “Europa Clipper Propulsion Module,” Online, Aug. 2021. URL <https://europa.nasa.gov/resources/207/europa-clipper-propulsion-module/?category=images>, accessed 14 September 2023.
- [12] Belvin, W. K., Doggett, W. R., Watson, J. J., Dorsey, J. T., Warren, J. E., Jones, T. C., Komendera, E. E., Mann, T., and Bowman, L. M., “In-Space Structural Assembly: Applications and Technology,” *3rd AIAA Spacecraft Structures Conference*, American Institute of Aeronautics and Astronautics, San Diego, California, USA, 2016. <https://doi.org/10.2514/6.2016-2163>, URL <http://arc.aiaa.org/doi/10.2514/6.2016-2163>.
- [13] Karumanchi, S., Edelberg, K., Nash, J., Bergh, C., Smith, R., Emanuel, B., Carlton, J., Koehler, J., Kim, J., Mukherjee, R., Kennedy, B., and Backes, P., “Payload-centric autonomy for in-space robotic assembly of modular space structures,” *Journal of Field Robotics*, Vol. 35, No. 6, 2018, pp. 1005–1021. <https://doi.org/10.1002/rob.21792>, URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21792>, eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/rob.21792>.
- [14] Cline, J., Vaughan, M., Waltz, W. J., and Wong, I. M., “LANDO: Developing Autonomous Operations for Planetary Surfaces,” *AIAA SCITECH 2022 Forum*, American Institute of Aeronautics and Astronautics, 2022. <https://doi.org/10.2514/6.2022-0625>.

- [15] Kelley, B. N., Cooper, J. R., Navarro, J. P., Vaughan, M., Waltz, W. J., Allen, B. D., Doggett, W., Avila, T. V., McQuarry, A. K., Shazly, S. A., Slick, R. M., and Williams, R. A., "Designing a Software Architecture for the Precision Assembly of Space Structures," *AIAA SCITECH 2022 Forum*, American Institute of Aeronautics and Astronautics, 2022. <https://doi.org/10.2514/6.2022-2077>.
- [16] ICONTechnology, "ICON To Develop Lunar Surface Construction System With \$57.2 Million NASA Award," Accessed 9/8/2023, Nov. 2022. URL 'www.iconbuild.com/newsroom/icon-to-develop-lunar-surface-construction-system-with-57-2-million-nasa-award'.
- [17] Everson, H., Moser, J., Quartaro, A., Glassner, S., and Komendera, E., "Autonomous Multi-Robot Assembly of Solar Array Modules: Experimental Analysis and Insights," *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2020. <https://doi.org/10.1109/iros45743.2020.9341298>.
- [18] Anderson, M. R., Pulling, C., and Komendera, E., "Robotic Detection of Damage in Autonomously Assembled Structures Via Synthetically Trained Classification Models," *Lecture Notes in Mechanical Engineering*, Springer Nature Singapore, 2022, pp. 299–310. [https://doi.org/10.1007/978-981-16-6490-8\\_25](https://doi.org/10.1007/978-981-16-6490-8_25).
- [19] Muthumanickam, N. K., Pless, S., Rothgeb, S., and Davis, J., "Industrialized and Robotic Construction Advances in Terrestrial Construction and Opportunities in Space Construction," 2022. URL <https://www.osti.gov/biblio/1880799>.
- [20] Prater, T. J., Nickens, T., Watts, J., Edmunson, J., Hanson, I., Bardsley, R., Lanigan, E., Harris, D., and Fiske, M., "NASA's Habitat Outfitting Portfolio: Technology Development to Support Future Habitation Systems," *ASCEND 2022*, American Institute of Aeronautics and Astronautics, 2022. <https://doi.org/10.2514/6.2022-4289>.
- [21] Lv, N., Liu, J., Xia, H., Ma, J., and Yang, X., "A review of techniques for modeling flexible cables," *Computer-Aided Design*, Vol. 122, 2020, p. 102826. <https://doi.org/10.1016/j.cad.2020.102826>.
- [22] Grégoire, M., and Schömer, E., "Interactive simulation of one-dimensional flexible parts," *Computer-Aided Design*, Vol. 39, No. 8, 2007, pp. 694–707. <https://doi.org/10.1016/j.cad.2007.05.005>.
- [23] Iskandar, M., van Ommeren, C., Wu, X., Albu-Schaffer, A., and Dietrich, A., "Model Predictive Control Applied to Different Time-Scale Dynamics of Flexible Joint Robots," *IEEE Robotics and Automation Letters*, Vol. 8, No. 2, 2023, pp. 672–679. <https://doi.org/10.1109/lra.2022.3229226>, citations for more MPC Robotic applications.
- [24] Carabis, D. S., and Wen, J. T., "Trajectory Generation for Flexible-Joint Space Manipulators," *Frontiers in Robotics and AI*, Vol. 9, 2022. <https://doi.org/10.3389/frobt.2022.687595>.
- [25] Kolachalama, S., and Lakshmanan, S., "Continuum Robots for Manipulation Applications: A Survey," *Journal of Robotics*, Vol. 2020, 2020, pp. 1–19. <https://doi.org/10.1155/2020/4187048>.
- [26] Rao, P., Peyron, Q., Lilge, S., and Burgner-Kahrs, J., "How to Model Tendon-Driven Continuum Robots and Benchmark Modelling Performance," *Frontiers in Robotics and AI*, Vol. 7, 2021. <https://doi.org/10.3389/frobt.2020.630245>.
- [27] Zhu, J., Navarro, B., Fraise, P., Crosnier, A., and Cherubini, A., "Dual-arm robotic manipulation of flexible cables," *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2018. <https://doi.org/10.1109/iros.2018.8593780>.
- [28] Suzuki, R., Okada, Y., Yokota, Y., Saijo, T., Eto, H., Sakai, Y., Murano, K., Ohno, K., Tadakuma, K., and Tadokoro, S., "Cooperative Towing by Multi-Robot System That Maintains Welding Cable in Optimized Shape," *IEEE Robotics and Automation Letters*, Vol. 7, No. 4, 2022, pp. 11783–11790. <https://doi.org/10.1109/lra.2022.3183529>.
- [29] Caporali, A., Galassi, K., Zanella, R., and Palli, G., "FASTDLO: Fast Deformable Linear Objects Instance Segmentation," *IEEE Robotics and Automation Letters*, Vol. 7, No. 4, 2022, pp. 9075–9082. <https://doi.org/10.1109/lra.2022.3189791>.
- [30] Liu, F., Su, E., Lu, J., Li, M., and Yip, M. C., "Robotic Manipulation of Deformable Rope-Like Objects Using Differentiable Compliant Position-Based Dynamics," *IEEE Robotics and Automation Letters*, Vol. 8, No. 7, 2023, pp. 3964–3971. <https://doi.org/10.1109/lra.2023.3264766>.
- [31] Lv, N., Liu, J., and Jia, Y., "Dynamic Modeling and Control of Deformable Linear Objects for Single-Arm and Dual-Arm Robot Manipulations," *IEEE Transactions on Robotics*, Vol. 38, No. 4, 2022, pp. 2341–2353. <https://doi.org/10.1109/tro.2021.3139838>.
- [32] Hermansson, T., Bohlin, R., Carlson, J. S., and Söderberg, R., "Automatic assembly path planning for wiring harness installations," *Journal of Manufacturing Systems*, Vol. 32, No. 3, 2013, pp. 417–422. <https://doi.org/10.1016/j.jmsy.2013.04.006>.

- [33] Suberkub, F., Laezza, R., and Karayiannidis, Y., “Feel the Tension: Manipulation of Deformable Linear Objects in Environments with Fixtures using Force Information,” *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2022. <https://doi.org/10.1109/iros47612.2022.9982065>.
- [34] Yang, Y., Stork, J. A., and Stoyanov, T., “Online Model Learning for Shape Control of Deformable Linear Objects,” *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2022. <https://doi.org/10.1109/iros47612.2022.9981080>.
- [35] Schwenzer, M., Ay, M., Bergs, T., and Abel, D., “Review on model predictive control: an engineering perspective, Planning,” *The International Journal of Advanced Manufacturing Technology*, Vol. 117, No. 5-6, 2021, pp. 1327–1349. <https://doi.org/10.1007/s00170-021-07682-3>.
- [36] Thrun, S., and Montemerlo, M., “The Graph SLAM Algorithm with Applications to Large-Scale Mapping of Urban Structures,” *The International Journal of Robotics Research*, Vol. 25, No. 5-6, 2006, pp. 403–429. <https://doi.org/10.1177/0278364906065387>.
- [37] Hommels, A., Murakami, A., and Nishimura, S., “Comparison of the Ensemble Kalman filter with the Unscented Kalman filter: application to the construction of a road embankment,” *GEO International*, 2009.
- [38] Wang, T., and Yamakawa, Y., “Real-Time Occlusion-Robust Deformable Linear Object Tracking With Model-Based Gaussian Mixture Model,” *Frontiers in Neurorobotics*, Vol. 16, 2022. <https://doi.org/10.3389/fnbot.2022.886068>, strain Energy Model, uses spring/damper at EDGES and no beams, 3D petrahedrons.
- [39] FBGS, “Shape Sensing,” Accessed 9/6/2023, 2023. URL <https://fbgs.com/solutions/shape-sensing/>.
- [40] Universal.Robotics, “THE UR10e,” Accessed 3/22/2023, 2023. URL <https://www.universal-robots.com/products/ur10-robot/>, <https://www.universal-robots.com/products/ur10-robot/>.
- [41] Corke, P., and Haviland, J., “Not your grandmother’s toolbox—the Robotics Toolbox reinvented for Python,” *2021 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2021, pp. 11357–11363.
- [42] Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and SciPy 1.0 Contributors, “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python,” *Nature Methods*, Vol. 17, 2020, pp. 261–272. <https://doi.org/10.1038/s41592-019-0686-2>.
- [43] Nocedal, J., and Wright, S., *Numerical Optimization, Optimal*, Springer, 2006. URL <https://link.springer.com/book/10.1007/978-0-387-40065-5>.