TEL AVIV UNIVERSITY
The Iby and Aladar Fleischman Faculty of Engineering
The Zandman-Slaner School of Graduate Studies

DUAL-ARM ROBOTIC MANIPULATION OF WIRES USING SOLELY FORCE/TORQUE FEEDBACK

A thesis submitted toward the degree of
Master of Science in Engineering

by

Itamar Mishani

June 2022

TEL AVIV UNIVERSITY

The Iby and Aladar Fleischman Faculty of Engineering
The Zandman-Slaner School of Graduate Studies

DUAL-ARM ROBOTIC MANIPULATION OF WIRES USING SOLELY FORCE/TORQUE FEEDBACK

A thesis submitted toward the degree of
Master of Science in Engineering

by

Itamar Mishani

This research was carried out at Tel Aviv University
in the School of Mechanical Engineering
Faculty of Engineering
under the supervision of Dr. Avishai Sintov

June 2022

Acknowledgments

I would like to thank my supervisor Dr. Avishai Sintov, your support and guidance, your availability and willingness to be there for me at every step of the way were a key element in creating my academic path. Your creativity, vision and enabling atmosphere were an inspiration to me.

In addition, I would like to acknowledge the best group I could ask to work with, my dear friends and lab co-workers at ROBTAU Lab: Noam Nahum, Osher Azulay, Nadav Kahanowich, Anton Gurevich, Eran Bamani, Omer Keinan, Maxim Monastirsky, Alon Mizrahi and Inbar Ben-David.

Thanks to my family: Yasmin, Eyal, Ilil, Eylon and Omri, for your endless support along the way.

My final, biggest thanks go to my wife, Efrat, for always supporting, believing and motivating me to achieve my goals.

Abstract

Manipulation of wires has been a challenging task and main interest for many decades. There have been many attempts to use visual perception and image segmentation to perform wire manipulation. However, wire manipulation in a cluttered environment where there are many visual occlusions and uncertainties (due to poor lighting or shadows) is hard. Giving a robot the ability to manipulate wires with high certainty is necessary and requires rapid reasoning of its shape in real-time. Furthermore, after having this ability, planning and control of wire manipulations is required. However, there is no efficient ability to do so without visual perception.

Recent work has shown that the shape of an elastic wire can be defined by a very simple representation. This representation can be interpreted as forces and torques at one end of the wire. To begin with, we experimentally analyzed the theoretical foundation. We deployed a dual-arm robotic system able to accurately manipulate an elastic wire. The system does not require complex visual perception and is able to reason about the shape of the wire by solely sensing forces and torques on one arm. Furthermore, we proposed a full framework in which the mechanical properties of the wire are rapidly approximated in real-time. Then, a simple control rule based on Force/Torque (F/T) feedback is used to manipulate the wire to some goal or track a planned path.

However, the model used to develop the system relies on assumptions that may not be met in real-world wires and does not take gravity into account. Therefore, the model cannot be applied to any wire with accurate shape estimation. Additionally, as the model does not consider the non-linearity of the F/T sensor, it is necessary to solve the non-linear non convex inverse problem, i.e. from wire shape to the F/T state, which is computationally expensive. As a consequence, we investigated the learning of a model to estimate the shape of a wire solely from measurements of the F/T state. We propose to train a novel learning model which can both act as a descriptor of the wire where F/T states can be mapped to its shape and as a solver of the inverse problem where a desired goal shape can be mapped to an F/T state. Additionally, we trained a different model, with the same dataset, which gives the robot the ability to execute a planned path.

Table of Contents

List of Figures

List of Tables

1 Introduction

The manipulation of thin elastic wires has been of interest for centuries [34, 28, 38, 46, 49]. Wire manipulation is considered a difficult task to operate in industrial environments. The common approach to manipulate deformable objects is using two robotic arms [28, 48]. However, in order to efficiently and safely manipulate wires, one requires sufficiently accurate models and control schemes. Once achieved, wire manipulation abilities for robots would be practical in many applications. For instance, cable routing is still operated manually in automotive production lines [21]. Other applications include knot tying [10, 42, 43, 52], surgical suturing [19], hot wire carving recently demonstrated by a robotic system [14] or manipulation of cables using autonomous aerial vehicles [9]. Progress in the robotic manipulation of elastic wires can also advance research in other manipulation tasks such as the handling and automation of flexible material (e.g., sheet metal parts) [39, 41], virtual reality and animation [27].

The configuration space describing the shape of an elastic wire has infinite dimension. Moreover, a multitude of wire shapes exists for a single pose of the robot arms holding it by the tips. These challenges have made the manipulation planning of a wire a challenging problem. The literature on path planning for elastic wires suggests exploring the set of equilibrium configurations indirectly, by sampling displacements of grippers and using numerical simulations to approximate their effect on the wire. This approach was developed in the work of Lamiraux and Kavraki [29] and was applied by Moll and Kavraki [37] to the manipulation of deformable linear objects. Hermansson et al. [17] relaxed gripping points constraints along an elastic harness while planning a collision-free path for a sphere around a predefined central grip point. These methods and similar like in [8, 51] use computationally expensive numerical methods that make them hard to perform well. A different approach used in [15, 22] simplified the model of the deformed object by reducing it to a sequence of rigid masses with springs. In such approach, the solution is highly sensitive to the approximation which in turn affects the quality of the planning.

In all the cited approaches, a feasible procedure to derive the free configuration space was not clear at that time. Seminal work by Bretl and McCarthy [13, 12] relied on a description of elastic rods in equilibrium as local solutions to a geometric optimal control problem and showed that the configuration space of the wire is a six-dimensional smooth manifold. The configurations space was also shown to be represented by the Force and Torque (F/T) at the base of the rod. Such revelation enabled the use of sampling-based planning algorithms to plan stable and collision-free paths [40, 44]. The work of Takano et al. [47] uses a Force/Torque sensor to estimate the shape of a thin strip based on a discrete model. Mechanical properties are not estimated and assumed to be known. While reducing the problem to a finite-dimensional space, these approaches are highly dependent on the resolution of the discretization and directly proportional to computation time. In addition, this method focuses mostly on shape estimation and do not provide an efficient ability to plan and control wire motions.

Estimating the shape of an elastic rod has also been discussed with visual perception approaches. In [20], a simulated discrete elastic rod model is fitted on data obtained from camera images. Similarly, a Fourier series was used to parameterize a cable segmented from an image [55]. Borum et al. [11] tracked fiducial markers on a planar wire with a camera and fitted them to the model of Bretl and McCarthy to estimate the wire's shape. In a more recent work, the instabilities of a rod were analyzed by identifying markers with a camera and compared to a numerical simulation [50]. However, visual perception and image segmenting of thin objects such as a wire in a cluttered environment is a challenging task. Moreover, relying on continuous visual feedback limits the performance of various tasks in which visual uncertainty (e.g., poor lighting or shadows) or occlusion may occur. This may include manipulating the wire within a confined space such as a vehicle frame.

In this work, the goal was to use F/T feedback to perform a full wire manipulation tasks, from estimation of wire shape, to planning a path and control it. The work was divided into two parts described in the following subsections. First, we relied on the work of Bretl and McCarthy and were able to propose a full framework for elastic wire manipulation tasks. Second, we addressed issues from part one, regarding model assumptions, by utilizing the learning methods.

1.1 Model based framework

In this part of the research, we propose a full framework to identify, plan and control the motion of an elastic wire using a dual-arm robot without visual perception. Theoretical

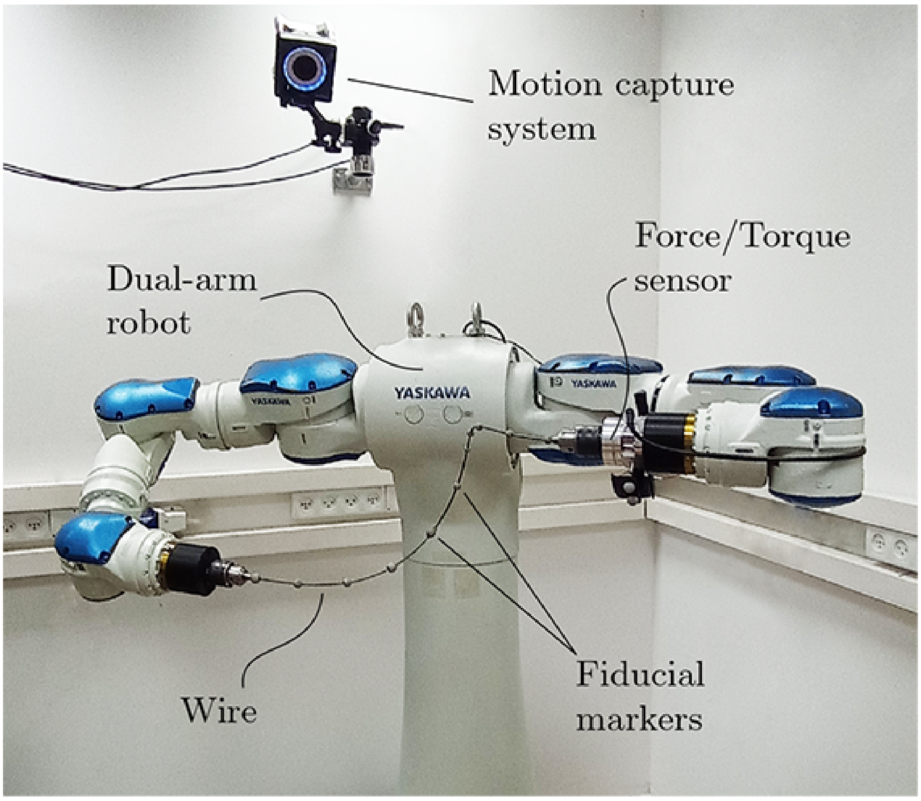

Figure 1.1: A dual-arm robot manipulating an elastic wire. The shape of the wire is estimated solely using Force/Torque sensing on one gripper.

analysis in [13] has shown that the configuration of a wire is, in fact, the force and torque exerted at one end of the wire. Hence, we utilized a F/T sensor to measure the load exerted on one gripper by the wire (Figure 1.1). We explored the sole use of the load measurement to accurately estimate the shape of wire. Our framework first estimates the mechanical parameters of the wire in real-time based only on F/T measurements and gripper poses. The accuracy of the shape estimation based only on load sensing is then analyzed over various wires. Furthermore, we propose a simple and novel control scheme to reach a desired configuration and track a planned path. Inaccurate tracking of a planned path may lead to wire instability and collision with obstacles. During a collision, the model deviates from the basic assumptions of the model (i.e., two fixture points) and predictions are not possible. Hence, accurate tracking of a path in a confined space is essential. This was the first implementation and experimental analysis of the theoretical foundation developed by Bretl and McCarthy [13] for shape estimation of spatial elastic wires using F/T measurements with a full scale dual-arm robotic system.

1.2 Learning based framework

After the development of a model based framework for real-time shape estimation and control of a wire solely using a F/T sensor and without any visual feedback [36], we propose a different approach to manipulate wires. Since the previous work relied on the analytical model devised by Bretl and McCarthy and since the model does not take into account gravitation, non-linearity of the sensors or other uncertainties, a Neural-Network (NN) was included to calibrate the F/T sensor to map between real sensed loads and theoretical ones defined in the model. However, such process requires solving the inverse problem for each sample where the theoretical load is computed given a measured shape. The computational complexity of such process is high and may take a very long time. Furthermore and once calibrated the F/T sensor, real-time estimation requires repeated solving of a system of ordinary differential equations to find the corresponding shape of the rod. Each solution is computationally expensive and the update frequency remains low [44].

To cope with the computational complexity of computing the shape of a wire, previous work has proposed to pre-compute a roadmap within the free configuration space of the wire as part of a path planning problem [44]. However, the roadmap, acting as a descriptor of the wire, represented only a small subset of possible wire configurations. In this part of the research, we explore data-based approaches to estimate the shape of the wire given an F/T measurement. A trained Neural-Network can be an higher-capacity descriptor of the wire enabling rapid solutions for its shape.

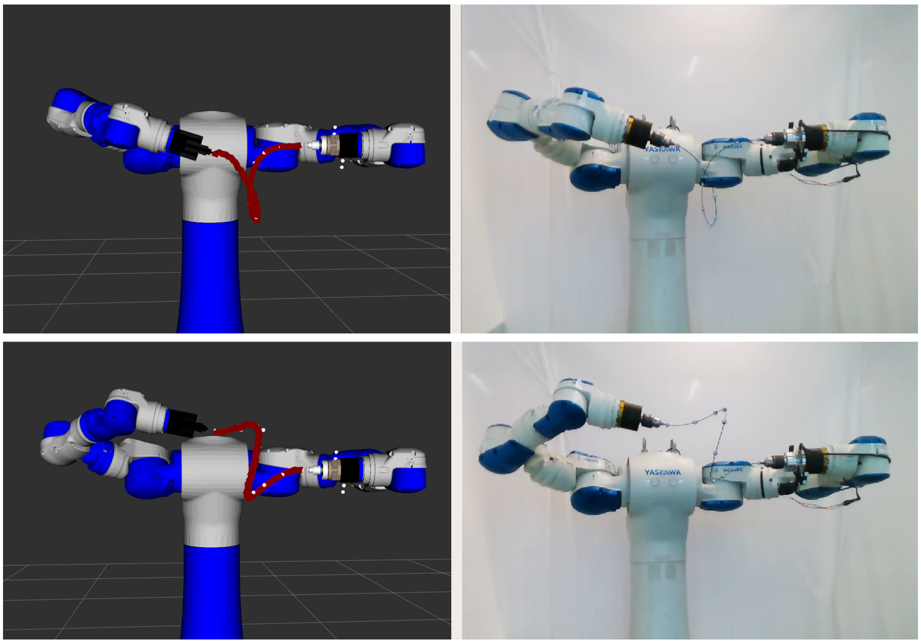

Figure 1.2: Shape estimation of a Nitinol elastic wire in two different configurations using only a Force/Torque sensor on one gripper. The wire configuration on the right is approximated and visualized in simulation (red curve), and compared to markers tracked (white markers) with a motion capture system.

We propose the Supervised and Convolutional Autoencoder (SnCAE) to learn wire shapes. We rely on the revelation of the work by Bretl and McCarthy where the shape of the wire can be represented in a lower-dimension space. An F/T measurement, therefore, is assumed to be an encoded state representation of the wire and can potentially be mapped to its explicit shape. Hence, we train the SnCAE which is a Supervised Autoencoder [32] constructed with convolutional layers. The SnCAE reconstructs the shape of the wire, based on collected data, while also supervising the latent space to match corresponding F/T states. Convolutional layers are included in order to embed the spatial shape of the wire and ease the learning. Once trained, the decoder of the SnCAE is the shape estimator and can rapidly map F/T states to wire shapes. Figure 1.2 shows an example where a trained decoder is used for real-time estimation of the spatial shape of a wire based on measured F/T and without visual perception. We test our approach on Nitinol and standard electrical wires.

In addition to the decoder, the encoder of the SnCAE provides the solution of the inverse problem where an F/T state of the wire can be extracted from a desired shape. Hence, planning in the space of F/T states can be performed while the goal state is extracted using the encoder from a measured shape. To demonstrate planning with the learned model, the same training data is used to train an NN to map a F/T state to the corresponding pose of one gripper relative to the first. Then, a motion planner is implemented where the encoder identifies goal F/T states and the decoder acts as a collision checker. To the best of the authors' knowledge, this is the first attempt to fully describe the shape of the wire using a data-based model and based on F/T measurements. While additional work must be done for generalization, our approach is a first step towards accurate description of wires without dependence on limited analytical models.

2 Background

2.1 Mathematical foundation

In the work of Bretl and McCarthy [13], it was shown that an elastic rod in static equilibrium is a local solution to a geometric optimal control problem. The energy of an elastic rod can be formulated as a cost function to be minimized in the form of a Lagrange problem (integral part only). With some constraints, the problem can be solved using the Pontryagin's maximum (minimum) principle (Pontryagin et al. 1962 [26]). Since the description of the shape of the wire is in a 3D workspace and will be described in $SE(3)$, our problem will become a problem of optimal control on manifolds.

2.1.1 Manifolds

As a first step, we will define manifolds and smooth manifolds that will be used throughout this work [33].

Definition 1. A ***manifold*** is a topological space which locally resembles an Euclidean space $\mathbb{R}^n$.

Intuitively, it is a generalization of curves and surfaces to higher dimensions.

Definition 2. A ***smooth manifold*** is a type of *manifold* which locally acts as a vector space and thus, one can apply calculus on it, i.e. the notion of differentiability exists on it.

In particular, graphs of smooth maps between Euclidean spaces are *smooth manifolds*.

2.1.2 Diffeomorphism

The following definitions are fundamental to understanding manifolds and play a vital role in our research.

Definition 3. [53] A map $F : U \to V$ between topological spaces is ***homeomorphism*** if F is invertible and both F and F^{-1} are continuous.

Definition 4. [33] If U and V are smooth manifolds and let $F : U \to V$ be a bijective map that has a smooth inverse, then F is ***diffeomorphism***.

In practice, diffeomorphism is isomorphism of smooth maniflods.

Definition 5. [33] If U and V are smooth manifolds, $F : U \to V$ is called ***local diffeomorphism*** if every point $p \in U$ has a neighborhood N such that $F(N)$ is open in V and $F|_N : N \to F(N)$ is *diffeomorphism*.

Definition 6. [33] If U and V are smooth manifolds and $F : U \to V$ is a smooth map, for each $p \in U$ we define a map

$$dF_p : T_pU \to T_{F(p)}V \tag{2.1}$$

called the ***differential of F at p***.

Where T represents the *tangential space* (2.1).

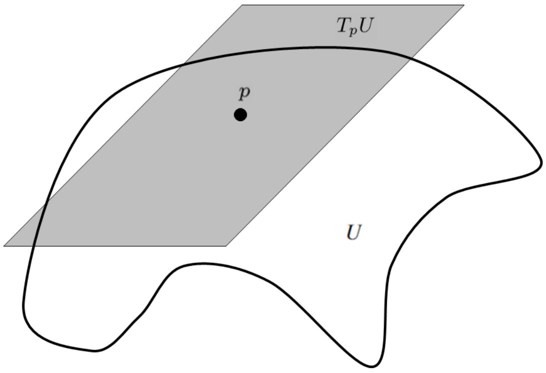

Figure 2.1: Tangent space at point $p \in U$

Lemma 1. [33] Let U, V be smooth manifolds, let $F : U \to V$ be smooth map and let $p \in U$:

- $dF_p : T_pU \to T_{F(p)}V$ is *linear*.
- If F is a *diffeomorphism*, then dF_p is an *isomorphism*

Lemma 2. [33] Suppose U and V are smooth manifolds, and $F : U \to V$ is a smooth map. If $p \in U$ is a point such that dF_p is invertible, then there are connected neighborhoods U_0 of point p and V_0 of $F(p)$ such that $F|_{U_0} : U_0 \to V_0$ is *diffeomorphism*.

Based on these *lemmas*, we can now define two *theorems* that will assist us in the following work.

Theorem 1. Let U and V be a smooth manifolds and let $F : U \to V$ be a map. F is a *local diffeomorphism* if and only if the *differential* map of F is a ***linear isomorphism***.

Proof. Suppose first that F is a local diffeomorphism. Given $p \in U$, there is a neighborhood N of p such that F is a diffeomorphism from N to $F(N)$. It then follows from 1 that dF_p is *linear isomorphism*. Conversely, if dF_p is *linear isomorphism* at each $p \in U$, then from 2 p has a neighborhood on which F restricts to a diffeomorphism into its image.

Theorem 2. dF_p is represented in coordinate bases by the ***Jacobian*** matrix of F.

Proof. See [33, p. 62]

2.2 Wire representation

In this section, we briefly present the theoretical background from Bretl and McCarthy [13]. Their work showed that each equilibrium configuration of a Kirchhoff elastic rod [7] corresponds to a unique point in a subset of $\mathbb{R}^6$. Furthermore, it was shown that a configuration is, in fact, the force and torque exerted at the base of the wire.

2.2.1 The configuration space of an elastic wire

We assume a wire of length L that is straight in the undeformed configuration with high enough stiffness so that the effects of gravity can be neglected. Using $t \in [0, L]$ to denote arc-length along the wire, the position and orientation of the wire at arc-length t are described by an element $\mathbf{q}(t)$ of the special Euclidean group $SE(3)$. The wire's shape is described by a continuous map $\mathbf{q}\colon [0,L] \to SE(3)$. given by

$$\mathbf{q}(t) = \begin{bmatrix} R(t) & \mathbf{p}(t) \\ \mathbf{0} & 1 \end{bmatrix}, \tag{2.2}$$

where $R(t) : [0,L] \to SO(3)$ and $\mathbf{p}(t) : [0,L] \to \mathbb{R}^3$ are curvature and position functions, respectively. In the Kirchhoff model, the wire is allowed to twist and bend, but is unshearable and inextensible [7]. These constraints are enforced by requiring q to satisfy the differential equation

$$\dot{\mathbf{q}} = \mathbf{q}\,(u_1 X_1 + u_2 X_2 + u_3 X_3 + X_4) \tag{2.3}$$

for some $\mathbf{u}\colon [0,L] \to \mathbb{R}^3$, where

$$X_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, X_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, X_3 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$X_4 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, X_5 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, X_6 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

is a basis for $\mathfrak{se}(3)$, the Lie algebra of $SE(3)$.

We assume that each end of the wire is held by a robotic gripper. The position and orientation of each point $q(t)$ on the wire is represented relative to the gripper at $t = 0$ (referred to as the *base gripper*) such that $q(0) = I$, where $I \in SE(3)$ is the identity matrix. This establishes the initial condition for differential equation (2.3). Furthermore, let $\mathcal{B} \subset SE(3)$ denote the space of boundary conditions at $\mathbf{q}(L)$, the configuration of the gripper holding the wire at $t = L$ (referred to as the *second gripper*) is denoted by $\mathbf{b} \in \mathcal{B}$. Under these conditions, the total elastic energy of the wire is

$$\frac{1}{2}\int_0^1 (c_1u_1^2 + c_2u_2^2 + c_3u_3^2)dt \tag{2.4}$$

where $c_1 > 0$ is the torsional stiffness of the wire and $c_2 > 0$ and $c_3 > 0$ are the bending stiffnesses of the wire. Therefore, an equilibrium configuration of the wire is a local minimum of the following problem:

$$\begin{aligned} \min_{q,u} \quad & \frac{1}{2}\int_0^1 (c_1u_1^2 + c_2u_2^2 + c_3u_3^2)dt \\ \text{s.t.} \quad & \dot{q} = q\,(u_1X_1 + u_2X_2 + u_3X_3 + X_4) \\ & q(0) = I, \;\; q(L) = \mathbf{b}, \;\; \mathbf{b} \in SE(3). \end{aligned} \tag{2.5}$$

We define the set $\mathcal{A} \subset \mathbb{R}^6$ by

$$\mathcal{A} = \{\mathbf{a} \in \mathbb{R}^6 : (a_2, a_3, a_5, a_6) \neq (0,0,0,0)\} \tag{2.6}$$

The set $\mathcal{A}$ is simply $\mathbb{R}^6$ with a two-dimensional plane removed. Each point in $\mathcal{A}$ corresponds to an equilibrium configuration of the wire and a local minimum of the total elastic energy of the wire. Proof for this can be viewed in Theorem 5 of Bretl and McCarthy [13].

Thus, one can solve the following six ordinary differential equations

$$\begin{aligned}
\frac{d\mu_1}{dt} &= \frac{\mu_3\mu_2}{c_3} - \frac{\mu_2\mu_3}{c_2} & \frac{d\mu_4}{dt} &= \frac{\mu_3\mu_5}{c_3} - \frac{\mu_2\mu_6}{c_2} \\
\frac{d\mu_2}{dt} &= \mu_6 + \frac{\mu_1\mu_3}{c_1} - \frac{\mu_3\mu_1}{c_3} & \frac{d\mu_5}{dt} &= \frac{\mu_1\mu_6}{c_1} - \frac{\mu_3\mu_4}{c_3} \\
\frac{d\mu_3}{dt} &= -\mu_5 + \frac{\mu_2\mu_1}{c_2} - \frac{\mu_1\mu_2}{c_1} & \frac{d\mu_6}{dt} &= \frac{\mu_2\mu_4}{c_2} - \frac{\mu_1\mu_5}{c_1}
\end{aligned} \tag{2.7}$$

on the interval $t \in [0, L]$ with the initial condition $\mu(0) = \mathbf{a}$ for $\mathbf{a} \in \mathcal{A}$. Next, functions $u_1 \colon [0, L] \to \mathbb{R}$ and $u_2, u_3 \colon [0, L] \to \mathbb{R}$ are the twisting and bending strains along the wire, respectively, such that $\mathbf{u} = (u_1, u_2, u_3)^T$ and $u_i = \mu_i / c_i$ for $i = 1, 2, 3$. Solving (2.3) with the resulting $\mathbf{u}$ produces an equilibrium shape of the wire, denoted by the pair of functions $(\mathbf{q}, \mathbf{u})$. Each $(\mathbf{q}, \mathbf{u})$ and the corresponding μ are completely defined by the choice of $\mathbf{a} \in \mathcal{A}$. Therefore and in practice, $\mathcal{A}$ serves as the configuration space of the wire. The resulting map is defined by $\mathcal{C} = \Phi(\mathcal{A})$. The map Φ is homeomorphism and thus, a bijective, i.e. for each $(\mathbf{q}, \mathbf{u}) \in \mathcal{C}$ there exists a unique $\mathbf{a} \in \mathcal{A}$ such that $(\mathbf{q}, \mathbf{u}) = \Phi(\mathbf{a})$ and vice versa. Furthermore, one may solve for matrix $\mathbf{J} : [0, L] \to \mathbb{R}^{6\times 6}$, the following linear arc-length-varying matrix differential equations

$$\dot{\mathbf{M}} = \mathbf{F}(\mu(t))\mathbf{M} \qquad \dot{\mathbf{J}} = \mathbf{G}\mathbf{M} + \mathbf{H}(\mu(t))\mathbf{J} \tag{2.8}$$

with initial conditions $\mathbf{M}(0) = I$ and $\mathbf{J}(0) = 0$. Definitions for $\mathbf{G}$, $\mathbf{F}(\cdot)$ and $\mathbf{H}(\cdot)$ are given in [13]. Here also, the matrices $\mathbf{M}$ and $\mathbf{J}$ are completely defined by the choice of $\mathbf{a} \in \mathcal{A}$.

We denote the set of all $\mathbf{a} \in \mathcal{A}$ that correspond to stable equilibrium configurations by $\mathcal{A}_{stable}$. A configuration $(\mathbf{q}, \mathbf{u})$ is a stable equilibrium configuration if $\det(\mathbf{J}(t)) \neq 0$ for all $t \in (0, L]$. Hence, we have a numerical test for each configuration $\mathbf{a} \in \mathcal{A}$ to determine which equilibrium configurations of the wire is in $\mathcal{A}_{stable}$. We define the free configuration space $\mathcal{A}_{free} \subset \mathcal{A}_{stable}$ to be the set of all $\mathbf{a} \in \mathcal{A}$ that correspond to stable equilibrium configurations of the wire and do not contain self-intersections. We next define the map $\Psi : \mathcal{C} \to \mathcal{B}$ such that a configuration $(\mathbf{q}, \mathbf{u})$ is mapped to $\mathbf{q}(L)$. Given a path of the wire in $\mathcal{C}_{stable}$, the function Φ can be used to find the path of the robotic gripper that causes the wire to follow the path in $\mathcal{C}_{stable}$. In particular, the map

$$\Gamma : \mathcal{A}_{stable} \to \mathcal{B}_{stable}, \tag{2.9}$$

where $\Gamma = \Phi \circ \Psi$, takes $\mathbf{a} \in \mathcal{A}_{stable}$ to the corresponding pose of the second gripper $\mathbf{b} = \mathbf{q}(L) \in \mathcal{B}_{stable}$. Map Γ is a *local diffeomorphism* [13].

2.2.2 Physical meaning of $\mathbf{a} \in \mathcal{A}$

The resulting function $\mu\colon [0,L] \to \mathbb{R}^6$ can be interpreted as the vector of internal forces and torques along the wire. Therefore, we can describe the force and torque at point t along the wire as

$$\mathbf{f}(t) = (\mu_4(t), \mu_5(t), \mu_6(t))^T \tag{2.10}$$

$$\tau(t) = (\mu_1(t), \mu_2(t), \mu_3(t))^T\,, \tag{2.11}$$

respectively, where $\mu_j(t)$ is the j^{th} component of $\mu(t)$ [13]. Consequently and since Φ is injective, any equilibrium configuration $\mathbf{a} = \mu(0)$ is completely defined by the force $\mathbf{f}(0)$ and torque $\tau(0)$ at the base gripper. In other words, by solely measuring the load exerted on the gripper using a F/T sensor, one can directly acquire the configuration $\mathbf{a}$ and, using $\Phi(\mathbf{a})$, solve for the shape of the wire $\mathbf{q}$. Similarly, the load measurement provides the expected position of the second gripper $\mathbf{b}$.

2.2.3 Dual-Arm Manipulation Planning Problem

Since we can now consider $\mathcal{A}$ as the configuration space of the wire during quasi-static motion, motion planning can be done in this space. The natural approach would be to implement sampling-based planning algorithms [24, 30]. However, practical applications require planning for two robotic arms to manipulate the wire from one pose and shape to another. In other words, the configuration space of the system is now the product space of the robot's joint space and $\mathcal{A}$.

Let $\mathcal{Q}$ be the configuration space of the two arms formed by their joint space product, and $\mathcal{A}$ be the configuration space of the wire as defined in (2.6). Thus, the configuration space of the system is defined as $\mathcal{Z} = \mathcal{A} \times \mathcal{Q}$. At any time instant and for computational purposes, a desired wire configuration $a \in \mathcal{A}$ can be considered as a rigid object. Thus, the two robotic arms holding the wire by its endpoints must impose a closed kinematic chain constraint

$$C(a, \phi) = 0 \tag{2.12}$$

for $\phi \in \mathcal{Q}$. Further, let $\mathcal{Z}_b \subset \mathcal{Z}$ be a restricted region due to obstacles, joint limits and the wire's infeasible set $\mathcal{A} \setminus \mathcal{A}_{free}$. Therefore, the free configuration set of the system is $\mathcal{Z}_o = \mathcal{Z} \setminus \mathcal{Z}_b$. Thus, we define the free configuration set $\mathcal{Z}_{free}$ as follows:

Definition 7. [45] Let $\mathcal{Z}_{free}$ be the free configuration set of the robotic arms and wire such that

$$\mathcal{Z}_{free} = \{(a, \phi) \in \mathcal{Z} : (a, \phi) \in \mathcal{Z}_o,\ C(a, \phi) = 0\} \tag{2.13}$$

is the set of configurations that satisfy the closed kinematic chain constraint, satisfy joint limits and are collision-free.

We now define the ”classical” motion planning problem as generally addressed in literature:

Problem 1. Given start and goal configurations $(\mathbf{a_s}, \phi_s) \in \mathcal{Z}_{free}$ and $(\mathbf{a_g}, \phi_g) \in \mathcal{Z}_{free}$, find a continues path

$$\gamma : [0,1] \to \mathcal{Z}_{free}$$

such that $\gamma(0) = (\mathbf{a_s}, \phi_s)$ and $\gamma(1) = (\mathbf{a_g}, \phi_g)$.

The above motion planning problem is a generic formulation to reach from one configuration to another while avoiding collisions and in-stability of the wire.

3 System

3.1 Hardware

Our system is composed of a Dual-arm robot manipulator, elastic wire, F/T sensor, custom grippers and motion capture system as shown in Figure 1.1. We use a fifteen degrees-of-freedom (DOF) Yaskawa Motoman SDA10F robot along with FS100 controller. Each arm has a seven DOF with an additional DOF for rotating the body of the robot. Each arm is a separated kinematic chain capable of maximum ten kilogram of payload. Both robot arms can synchronously work together or simultaneous perform tasks independently.

On the robot left arm, we have mounted a six-axis F/T sensor, Bota SensONE. To integrate the sensor and the robot arm, we designed and printed a connector to which the sensor can be mounted on. The end-effectors (Figure 3.1), were custom designed to grasp a circular rod using drill chucks. A set of $V = 11$ fiducial markers were placed along the installed wire so that the OptiTrack motion capture system is able to provide ground-truth measurements of its shape in real-time.

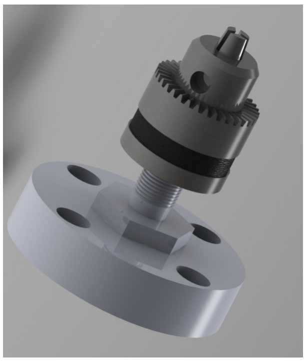

Figure 3.1: A chuck gripper has been added to each arm, allowing it to hold a circular rod without slippage.

3.2 Software

Data acquisition, control and communication were implemented using the Robot Operating System (ROS) over an Ubuntu machine. Action, perception and cognition are the three principal modules of the system. To begin with, we need to be able to control robot actions so that we can manipulate wires. Next, we must possess sensing capabilities in order to communicate with the environment. Once we have both, we can develop algorithms to manipulate wires.

3.2.1 Robot

To control the robot, we first needed to utilize two packages from [6] in order to stream data and have a driver to the robot - `motoman_sda10f_support` and `motoman_driver`. In order to include sensors and grippers in the transformation tree of the robot, we built a description package that includes the robot and all of its additional components [3.2]. It is important to notice that, in both arms, we defined the last joint as the contact point with the cable, i.e. in the base gripper (where the F/T sensor is mounted) this frame is the reference frame for the models [35]. After obtaining streaming, driver, and description, we could visualize the robot in Rviz and perform manipulation using the driver and MoveIt.

3.2.2 Motion Capture System

Along with the wire markers, we have also added four fixed markers on the base gripper to be tracked by the motion capture system as a rigid body. The rigid body was configured in the Motive software provided by OptiTrack so its center of mass coincides with the location of the reference frame described above and has the same orientation. Since a wire is not a rigid body, all wire mounted fiducial markers were not classified as rigid bodies. The data from the software was transmitted over the network. In order to read this data from the network, we updated a package that captures all transmitted data and automatically publishes the rigid bodies and individual markers in ROS [18]. This data was added to the robot transformation tree by defining the four markers rigid body fixed to the reference frame. As a ground truth for the wire shape, the positions of the fiducial markers, measured relative to the rigid body and therefore to the reference frame, were extracted from the data. This data was also visualized in Rviz.

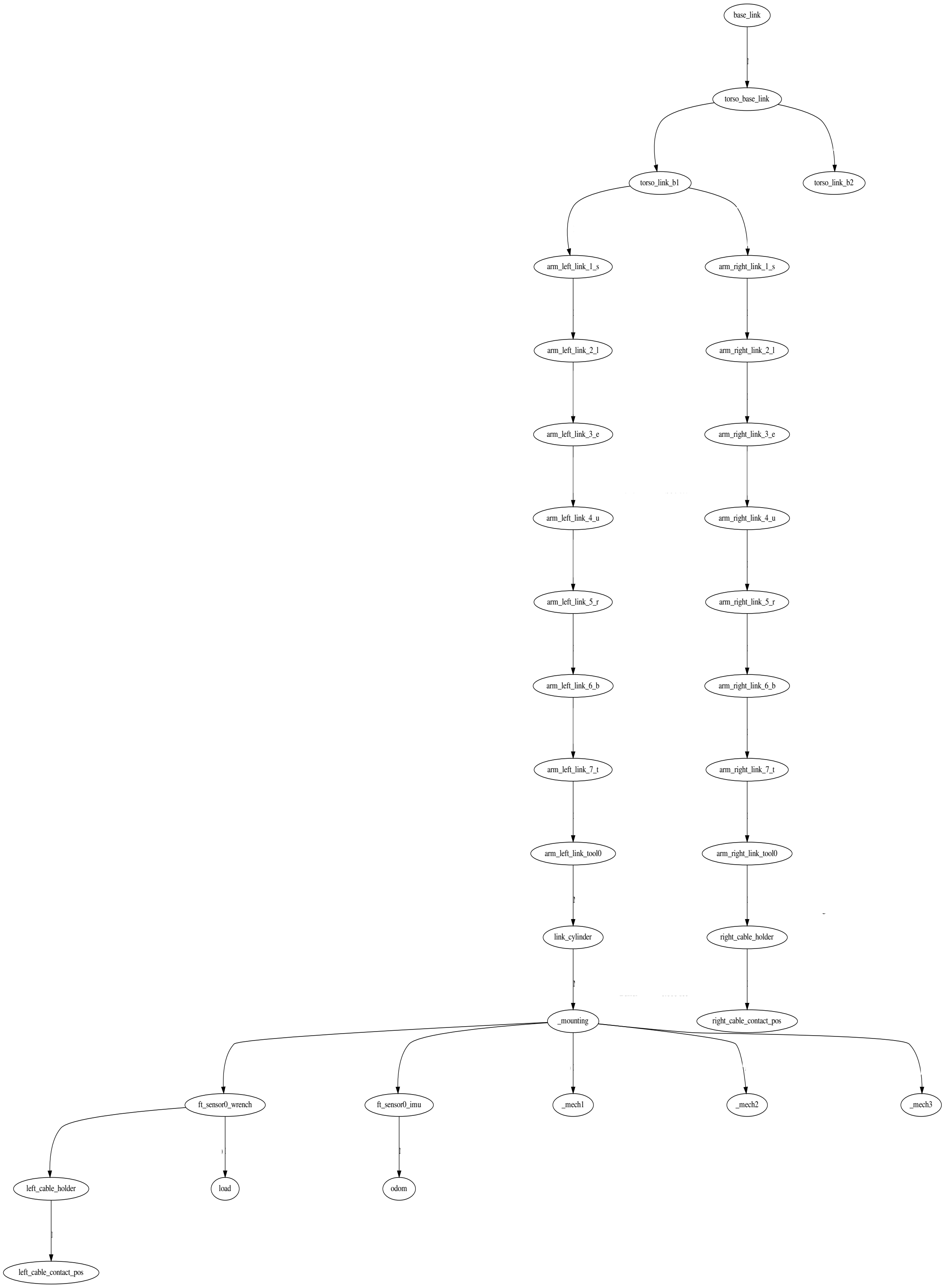

Figure 3.2: Robot transformation tree - each circle represents a frame, and the arrows indicate the connections.

3.2.3 Force/Torque Sensor

The F/T sensor needed to be calibrated in order to accommodate a gripper attached to it. We used Bota SensOne provided open source code [1] with adjustments, to perceive F/T and IMU readings. Additionally, we adjusted calibration package based on [2] ROS package. With the F/T sensor calibrated, we were able to publish filtered data at up to 100 Hz.

3.2.4 Algorithms

The algorithms have been developed in three separate packages. The first one [5] is responsible for all robot motions (path and motion planning and control) and visualization in Rviz. As the second package [4], there are models scripts that include optimization algorithms and analytical wire shape estimation scripts. Finally, the third package [3] consists of the development of all learning methods. figure 3.3 illustrates the data flow between the packages.

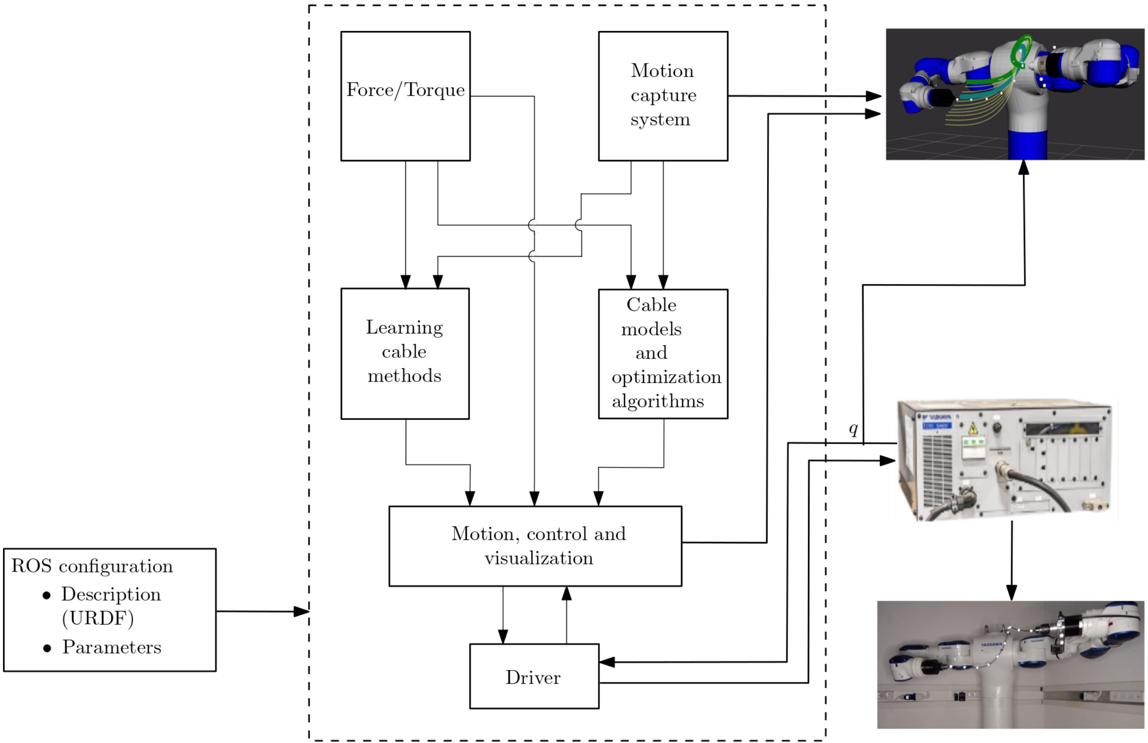

Figure 3.3: Data flow between system packages where each box in the diagram represents a package, while the arrows indicate the flow of data between them.

4 Model Based Framework

In this chapter we describe the development of the manipulation framework while deploying the analytical foundation presented previously. Using the presented method we conducted a various experiments to analyze wire properties approximations, pose estimations and control accuracies. **The work in this chapter was published in the *IEEE Robotics & Automation Letters* [36] and presented at the *IEEE Conference on Robotics and Automation*, 2022.**

4.1 Method

Given two robotic arms with a Force/Torque (F/T) sensor mounted on one arm. The arms hold a wire of length L and mechanical coefficients $\mathbf{c} = (c_1, c_2, c_3)^T$ by its end-tips. We aim to use the measurement $\tilde{\mathbf{a}} \in \mathcal{A}$ of the F/T sensor to approximate the shape of the wire $\tilde{\mathbf{q}}$. Motion of the wire is considered to be quasi-static in order to maintain equilibrium configurations. In this section, we present the proposed framework accurately manipulate a given wire. A scheme of the framework is illustrated in Figure 4.1. We first describe the process to estimate the length and coefficients of a new wire solely using F/T measurements and second gripper pose $\mathbf{b}$. Then, we propose an easy to implement rule to control the shape of the wire.

4.1.1 Perturbation mapping from $\mathcal{A}$ to $\mathcal{B}$

Given $\mathbf{a}_i \in \mathcal{A}_{stable}$ and its corresponding end-tip pose $\mathbf{b}_i = \Gamma(\mathbf{a}_i)$. We define an homogeneous transformation matrix $M \in SE(3)$ with $\delta\mathbf{x} \in \mathbb{R}^3$ and exponential coordinates $\mathbf{w} \in \mathbb{R}^3$ and $\delta\theta \in [0, \pi)$ such that

$$M(\delta\mathbf{b}) = \begin{bmatrix} e^{\mathrm{w}\delta\theta} & \delta\mathbf{x} \\ \mathbf{0} & 1 \end{bmatrix}, \tag{4.1}$$

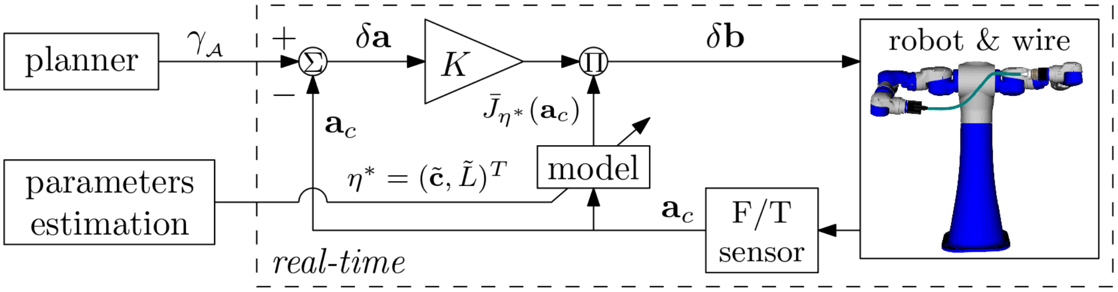

Figure 4.1: Illustration of the control scheme. In an off-line operation, parameters estimation will estimate c and L, and update the model. Also, in an offline process, a planner plans a path to some goal. In the online process and given a goal configuration $\mathbf{a}_g$ or a path $\gamma_{\mathcal{A}}$ to track, the controller will provide the required perturbations of the second gripper. Feedback is acquired using a force/torque sensor on the base arm whom provides information about the current configuration $\mathbf{a}_c$ of the wire.

where

$$\delta \mathbf{b} = \begin{pmatrix} \mathbf{w}\delta\theta \\ \delta \mathbf{x} \end{pmatrix}. \tag{4.2}$$

Matrix M is defined to map between two configurations in $\mathcal{B}$ such that

$$\mathbf{b}_{i+1} = \mathbf{b}_i M(\delta \mathbf{b}) \tag{4.3}$$

where perturbation to $\mathbf{b}_{i+1}$ will result in wire configuration $\mathbf{a}_{i+1}$. Map Γ is a local diffeomorphism being smooth and has a non-singular Jacobian matrix $\mathbf{J}(L)$. From Theorem 7 and equation (37) in [13], we get that

$$\delta \mathbf{b} \approx \mathbf{J}(L)\delta \mathbf{a} \tag{4.4}$$

where $\delta \mathbf{a} = \mathbf{a}_{i+1} - \mathbf{a}_i$. Equation (4.4) states that matrix $\mathbf{J}(L)$ contains information about the relationship between small changes in $\mathcal{A}$ and small changes in $\mathcal{B}$. Therefore and given a desired $\mathbf{a}_{i+1}$ in the vicinity of $\mathbf{a}_i$, the required perturbation $\delta \mathbf{a}$ in $\mathcal{A}$ can be obtained. Then, by solving (4.4), one can use map (4.3) to compute the required perturbation in $\mathcal{B}$ in order to move a wire from configuration $\mathbf{a}_i$ to $\mathbf{a}_{i+1}$.

4.1.2 Real-time approximation of **c** *and L*

Given a measurement $\tilde{\mathbf{a}}_j$ of the F/T sensor, one can solve $(\mathbf{q}_j, \mathbf{u}_j) = \Phi(\tilde{\mathbf{a}}_j)$ with (2.7) to acquire the current shape q_j of the wire. In order to solve (2.7), however, one must first estimate the mechanical coefficients vector $\mathbf{c}$ of the wire. Given a wire picked-up by two robotic arms, the information available includes measurements of $\tilde{\mathbf{a}}$ and $\tilde{\mathbf{b}}$. We aim to rapidly compute an estimation $\tilde{\mathbf{c}}$ of $\mathbf{c}$ and the wire length $\tilde{L}$. We first formulate the cost function to be minimized in order to estimate $\mathbf{c}$. Recall that matrix $\mathbf{J}(L)$ is dependent on the choice of $\mathbf{c}$. Hence, (4.4) implies that differential measurements of $\mathbf{a}$ and $\mathbf{b}$ are required in order to estimate $\mathbf{c}$ and $\mathbf{L}$. We propose a real-time model identification process based on such notion.

While moving the robot arms, a path $\mathcal{B}_c = \{\tilde{\mathbf{b}}_1, \tilde{\mathbf{b}}_2, \ldots, \tilde{\mathbf{b}}_m\}$ of the second gripper relative to the base one is being recorded along with its corresponding vectors of force and torque $\mathcal{A}_c = \{\tilde{\mathbf{a}}_1, \tilde{\mathbf{a}}_2, \ldots, \tilde{\mathbf{a}}_m\}$ exerted on the base gripper. Sets $\mathcal{A}_c$ and $\mathcal{B}_c$ are then used to compute the differential sets $\mathcal{A}'_c = \{\delta\tilde{\mathbf{a}}_1, \delta\tilde{\mathbf{a}}_2, \ldots, \delta\tilde{\mathbf{a}}_{m-1}\}$ and $\mathcal{B}'_c = \{\delta\tilde{\mathbf{b}}_1, \delta\tilde{\mathbf{b}}_2, \ldots, \delta\tilde{\mathbf{b}}_{m-1}\}$, respectively, where $\delta\tilde{\mathbf{a}}_i = \tilde{\mathbf{a}}_{i+1} - \tilde{\mathbf{a}}_i$ and $\delta\tilde{\mathbf{b}}_i$ is computed according to (4.2)-(4.3). Also, let $\eta = (\mathbf{c}, L)^T \in \mathbb{R}^4$ be the vector of unknown parameters. Furthermore, we define matrix $\bar{\mathbf{J}}_\eta(\mathbf{a})$ to be the Jacobian $\mathbf{J}(L)$ of configuration $\mathbf{a}$ and computed with η. An approximation of η is the solution of the following problem

$$\begin{aligned} \eta^* = arg\min_{\eta} \quad & \sum_{i=1}^{m-1} \|\bar{\mathbf{J}}_\eta(\tilde{\mathbf{a}}_i)\delta\tilde{\mathbf{a}}_i - \delta\tilde{\mathbf{b}}_i\|^2 \\ \text{s.t.} \quad & \eta > 0. \end{aligned} \tag{4.5}$$

Problem (4.5) is non-linear and non-convex with regards to η and, therefore, requires a global minimization algorithm. We employ a wire model identification process based on the Particle Swarm Optimization (PSO) [25] algorithm. PSO is a meta-heuristic global optimization algorithm that minimizes some cost function by iteratively improving candidate solutions. The algorithm maintains a population of particles in the search space, i.e., candidate solutions, each having its own cost value. Their position is iteratively updated according to simple mathematical rules. The rules consider the momentary best local and global cost, and are aimed to both explore new regions and exploit local information. Since our goal is to estimate η in real-time, convergence to the best solution given some amount of data must be fast. Thus, a small population size is used. The particles are continuously updated when new measurements are acquired to refine η^*. Moreover, to avoid convergence to local minima, we randomly sample new particles over the search space.

4.1.3 Control in $\mathcal{A}_{free}$

Given a wire goal configuration $(\mathbf{q},\mathbf{u})_g \in \mathcal{C}$, directly controlling the path of the wire in $\mathcal{C}$ requires continuous visual feedback of its shape while the policy for moving the arms is not clear [29]. Alternatively, we propose to control the motion in $\mathcal{A}$ with no visual perception and based solely on F/T measurements. Given the current start configuration of the wire $\mathbf{a}_s \in \mathcal{A}_{free}$ and goal destination $\mathbf{a}_g \in \mathcal{A}_{free}$, we aim for a control rule for the grippers to manipulate the wire from $\mathbf{a}_s$ to $\mathbf{a}_g$. From perturbation mapping (4.4), we can formulate an iterative correction rule for the second gripper pose relative to the base gripper as

$$\delta\mathbf{b}_i = K\bar{\mathbf{J}}_{\eta^*}(\tilde{\mathbf{a}}_i)(\mathbf{a}_g - \tilde{\mathbf{a}}_i) \tag{4.6}$$

where $\tilde{\mathbf{a}}_0 = \mathbf{a}_s$. The control gain $K > 0$ is chosen to constrain motion of the wire to small steps in order to enable frequent corrections of deviations. In addition, a small value for K prevents fast manipulations that could hinder the quasi-static motion required to maintain static equilibrium. According to (4.3), applying perturbation $\delta\mathbf{b}_i$ to the current configuration $\mathbf{b}_i$ of the second gripper (relative to the base gripper) yields

$$\mathbf{b}_{i+1} = \mathbf{b}_i M(\delta\mathbf{b}_i). \tag{4.7}$$

Note that since both arms share the same base frame, the relative motion is possible when transforming it from base gripper frame to the shared base frame of the arms. The value for $\tilde{\mathbf{a}}_{i+1}$ is acquired once the second gripper is moved to $\mathbf{b}_{i+1}$. The process is repeated for $i = 0,1,\ldots,k$ until satisfying $\|\mathbf{a}_g - \tilde{\mathbf{a}}_i\| < \varepsilon$ where $\varepsilon > 0$ is an accuracy distance threshold in $\mathcal{A}$.

When no uncertainties exist, the motion will be on a straight line $\bar{\mathbf{a}}(\sigma) = \sigma(\mathbf{a}_g - \mathbf{a}_s) + \mathbf{a}_s$ for $\sigma \in [0,1]$. If $\mathbf{a}_s$ and $\mathbf{a}_g$ are $\mathcal{A}$-connected [13], i.e., $\bar{\mathbf{a}}(\sigma) \in \mathcal{A}_{free}$ is satisfied for all $\sigma \in [0,1]$, motion to $\mathbf{a}_g$ along $\bar{\mathbf{a}}(\sigma)$ is feasible. However, controller (4.6)-(4.7) ignores the kinematics of the dual-arm system and does not guarantee it can provide the required motion. Hence, tracking a planned path where these are constrained is presented next.

4.1.4 Path tracking

Path $\gamma(s)$ with $s \in [0,1]$ [Problem 1] can be divided into $\gamma_{\mathcal{A}}(s) \in \mathcal{A}$ and $\gamma_{\mathcal{Q}}(s) \in \mathcal{Q}$. Planning in $\mathcal{Z}_{free}$ ensures that the path is feasible in terms of wire stability, obstacles and kinematics of the arms.

While the planning of path γ considers motion both in $\mathcal{Q}$ and $\mathcal{A}$, the tracking accuracy of the wire is prioritized for successful task completion. In open-loop control, however, tracking would be conducted by solely moving the arms along path $\gamma_{\mathcal{Q}}$ without the ability to reason about accuracy along $\gamma_{\mathcal{A}}$. Therefore, positioning inaccuracies and uncertainties may deviate the wire from the desired path $\gamma_{\mathcal{A}}$ in $\mathcal{A}_{free}$. Correcting deviations along $\gamma_{\mathcal{A}}$ is not possible in such approach. Alternatively, we propose to ignore $\gamma_{\mathcal{Q}}$ and control the motion to solely track $\gamma_{\mathcal{A}}$ using (4.6)-(4.7). Tracking would be performed on a set of N discrete points $\{\bar{\mathbf{a}}_0, \bar{\mathbf{a}}_1, \ldots, \bar{\mathbf{a}}_N\} \in \gamma_{\mathcal{A}}(s)$ where $\bar{\mathbf{a}}_i = \gamma_{\mathcal{A}}(\frac{i}{N})$. The path in $\mathcal{A}_{free}$ is now piecewise linear and motion between each two points is controlled as described in Section 4.1.3. While we ignore the planned path $\gamma_{\mathcal{Q}}$, if the model of the wire is sufficiently accurate, the true path $\tilde{\gamma}_{\mathcal{Q}}$ of the dual-arm system with (4.6)-(4.7) is expected to be approximately similar such that $\tilde{\gamma}_{\mathcal{Q}}(s) \approx \gamma_{\mathcal{Q}}(s)$.

4.2 Experiments

4.2.1 Force/Torque sensor Calibration

Direct measurements of the F/T sensor do not reflect the pure force and torque that are exerted by the wire due to gripper fabrication uncertainties and intrinsic non-linearity of the sensor. Hence, we employ a machine learning based calibration process of the F/T sensor. We use a Nitinol wire with known length and mechanical coefficient vector $\mathbf{c}$ to collect labeled data. The wire is manipulated through various configurations while recording for each the F/T measurement $\bar{\mathbf{a}}_j$ and the corresponding set of marker locations $\mathcal{P}_j = \{\mathbf{p}_{j,1}, \ldots, \mathbf{p}_{j,V}\}$ where $\mathbf{p}_{j,k} \in \mathbb{R}^3$ is the spatial position of marker k relative to the base gripper. For each sample $\{\bar{\mathbf{a}}_j, \mathcal{P}_j\}$, we compute the theoretical wire configuration $\mathbf{a}_j$ by solving the following minimization problem

$$\mathbf{a}_j = arg \min_{\mathbf{a}} \sum_{k=1}^{V} \|\mathbf{p}_{j,k} - \mathbf{x}_k(\mathbf{a})\|^2 \tag{4.8}$$

where $\mathbf{x}_k \in \mathbb{R}^3$ is the closest point to $\mathbf{p}_{j,k}$ on a wire $(\mathbf{q}, \mathbf{u}) = \Phi(\mathbf{a})$. Problem (4.8) is solved off-line for each sample using PSO. Note that the solution of (4.8) is the inverse of Γ where the respected $\mathbf{a}$ is found based on a desired shape of the wire. The final product is a dataset comprised of K input samples $\{\bar{\mathbf{a}}_1, \ldots, \bar{\mathbf{a}}_K\}$ and output labels $\{\mathbf{a}_1, \ldots, \mathbf{a}_K\}$. The data is then used to train an Artificial Neural-Network (ANN) to map a perceived F/T measurement to the respected model-based configuration $\mathbf{a}$ of the wire. This process is done once. Then, the trained ANN can be deployed to acquire wire configurations in real-time and approximate **c** as described next. Videos of the experiments and demonstration can be viewed online. [1]

Table 4.1: Accuracy results for mechanical coefficients and length estimation of a Nitinol wire

Diameter	Coefficient	True	Estimated	Relative Error (%)
2 mm	$\mathbf{c}_1$ (Nm^2)	0.042	0.043 ± 0.006	2.4
	$\mathbf{c}_2$ (Nm^2)	0.055	0.056 ± 0.008	1.8
	$\mathbf{c}_3$ (Nm^2)	0.055	0.056 ± 0.008	1.8
	L (m)	0.630	0.635 ± 0.012	0.8
3 mm	$\mathbf{c}_1$ (Nm^2)	0.214	0.215 ± 0.008	0.5
	$\mathbf{c}_2$ (Nm^2)	0.278	0.279 ± 0.011	0.4
	$\mathbf{c}_3$ (Nm^2)	0.278	0.279 ± 0.011	0.4
	L (m)	0.820	0.811 ± 0.013	1.1

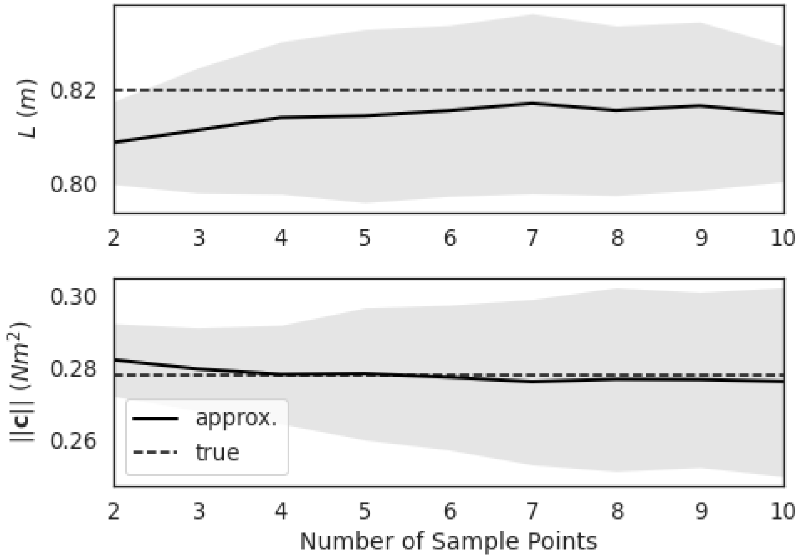

Figure 4.2: Mean error and standard deviation for approximating c and L for the 3 mm diameter wire with regards to the number of sample points.

4.2.2 Approximation of **c** *and L*

We have conducted an approximation of two Nitinol wires of 2 mm and 3 mm diameter with known mechanical coefficients. The second gripper was randomly moved while adding poses to $\mathcal{B}_c$ and F/T measurements to $\mathcal{A}_c$. The sampling frequency is 10 Hz. Using

[1] https://www.youtube.com/watch?v=1nOaoQ92zoofeature=youtu.be

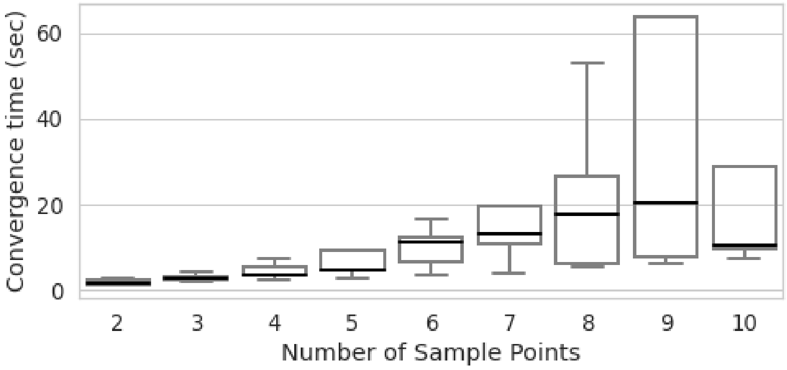

Figure 4.3: Time to estimate c and L for the 3 mm diameter wire with regards to the number of samples taken.

PSO with 20 particles, vector η was approximated by solving (4.5) while collecting data in real-time. Table 4.1 presents the average of the approximated values and their relative error after 10 repeated trials and for only three sampled points in $\{\mathcal{A}_c, \mathcal{B}_c\}$. Figure 4.2 presents the approximated values for the 3 mm diameter wire with regards to the number of recorded data points in $\{\mathcal{A}_c, \mathcal{B}_c\}$. The results show that with only two samples, the approximation accuracy is high. Adding more data points does not provide significant accuracy improvement. In addition, Figure 4.3 reports the convergence time to reach a solution given a number of samples. The results show that an accurate real-time approximation is available after only three seconds of motion while sensing only F/T and second gripper poses.

4.2.3 Wire shape estimation

In this section, we analyze the accuracy of shape estimation using the F/T measurements. Four wires are tested including three Nitinol wires of different lengths and diameters, and an electric cable made of copper with a polymeric insulator. For each wire, the mechanical properties were estimated. Then, pose estimation was averaged for 50 configurations in $\mathcal{A}_{free}$. Table 4.2 presents the mean and standard deviation shape errors for the tested wires. The results show that Nitinol wires acquire high shape estimation accuracy since their mechanical properties comply with the assumption described in Section 2.2.1. On the other hand, the electric wire is not straight in the relaxed form and highly affected by gravitation which is not included in the model. Hence, the electric cable does not comply with the assumptions of the model and large errors are exhibited. Figure 4.4 shows an example of one pose estimation with Nitinol wire number 3.

Table 4.2: Accuracy results for shape estimation of various wires

Num.	Wire	Diameter	length	Mean Error
	type	(mm)	(mm)	(mm)
1	Nitinol	2	630	5.72 ± 2.58
2	Nitinol	3	910	7.42 ± 1.33
3	Nitinol	3	820	6.55 ± 0.87
4	Electric	3	920	52.3 ± 11.5

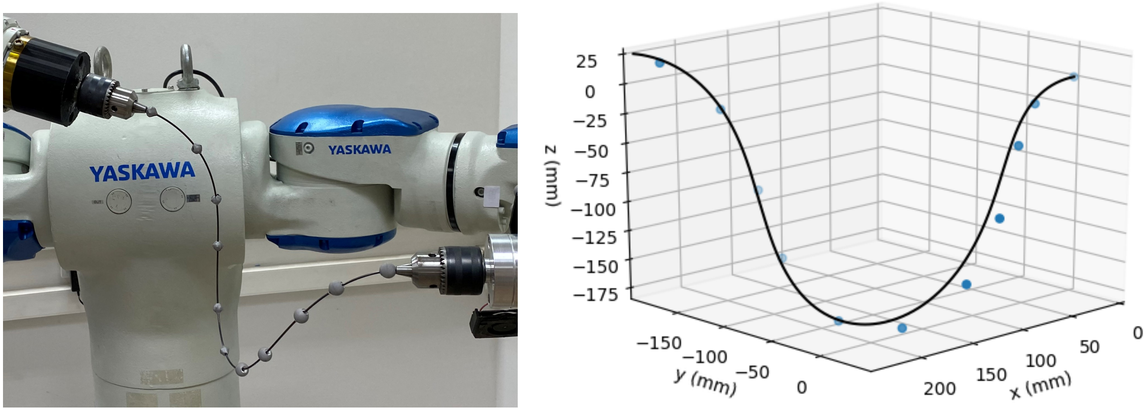

Figure 4.4: Shape estimation of a wire configuration using only F/T measurement. The circular blue markers are the ground-truth measured using a motion capture system and the solid curve is the estimation. Average shape estimation error along the wire is 5.81 mm.

4.2.4 Control

In the first control experiment, we test the accuracy when manipulating the wire to given goals using (4.6). Hence, we sample 30 random start and goal configurations in $\mathcal{A}_{free}$ which are known to be $\mathcal{A}$-connected. Once the system is at a start configuration, the controller is applied to reach the corresponding goal. The controller gain was set to $K = 0.2$. We compare the control to an open-loop setting in which a straight line path from $\mathbf{a}_s$ to $\mathbf{a}_g$ is computed along with the corresponding gripper poses. Then, the gripper poses are rolled-out without any feedback. Furthermore, we test the performance in a case where the positioning of the grippers is inaccurate. Thus, we add random normal noise $\mathcal{N}(0,10)$ *mm* and $\mathcal{N}(0,3^\circ)$ to the position and orientation, respectively, of the control output $\delta\mathbf{b}_i$ in (4.6). Hence, we add inaccuracies to the relative pose between the two grippers. Table 4.3 presents results of the mean shape error along the wire for open and closed loop control with and without adding noise. Without noise, the high accuracy of the robotic arms provides low errors in open-loop. However, inaccuracy of the arms (with noise) leads to

Table 4.3: Mean accuracy for reaching goals

	Open-loop (mm)	Closed-loop (mm)
without noise	4.80 ± 1.19	9.67 ± 3.02
with noise	17.35 ± 9.54	9.11 ± 3.62

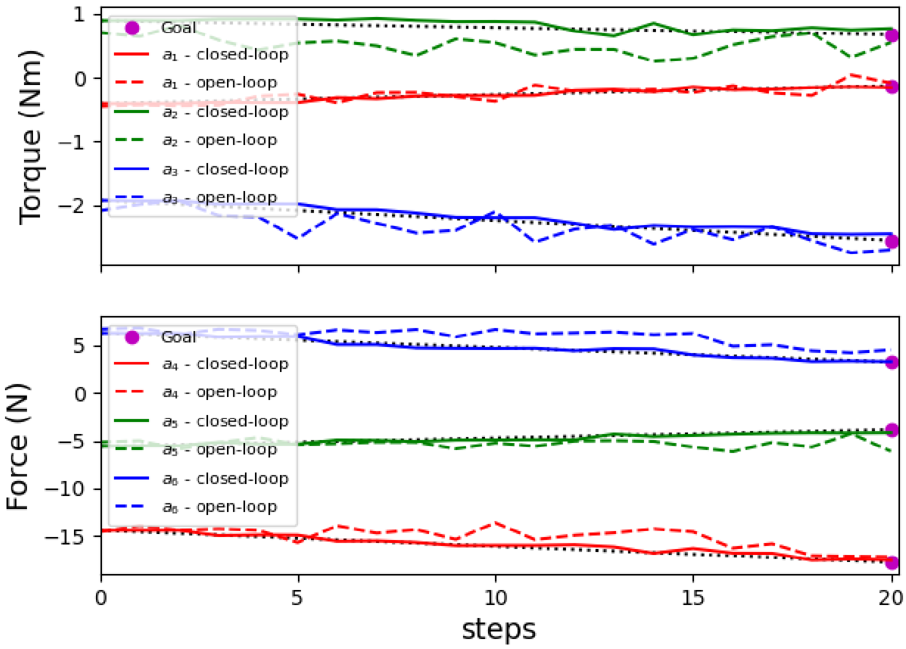

Figure 4.5: Wire manipulation towards the goal in the $\mathcal{A}$ with open (dashed) and closed-loop (solid) control. Dotted lines illustrate the nominal straight line path in $\mathcal{A}$.

poor goal reach. On the other hand, the proposed closed-loop control over $\mathcal{A}$ shows the ability to compensate these inaccuracies and maintain accuracy with and without noise. Figure 4.5 shows an example of motion towards a goal in the $\mathcal{A}$ in open and closed-loop control. Closed-loop control exhibits smoother motion along a straight line and better accuracy at the final configuration. Figure 4.6 shows the corresponding wire shape errors in $\mathcal{C}$ relative to the shape $(\mathbf{q}, \mathbf{u})_g = \Phi(\mathbf{a}_g)$. The figure also shows (in dotted lines) the error for moving along a straight line in $\mathcal{B}$ without considering F/T measurements. Closed-loop control clearly reaches closer to the desired shape. It is important to note that, while the control converges to the desired configuration in $\mathcal{A}$, the error in $\mathcal{C}$ did not converge to zero due to inaccuracies of the F/T sensor calibration model described in Section 4.2.1. Overall, the results exhibit good accuracy in reaching various goals with control regardless of arm inaccuracies.

We now experiment the tracking of a path with open and closed loop control as dis-

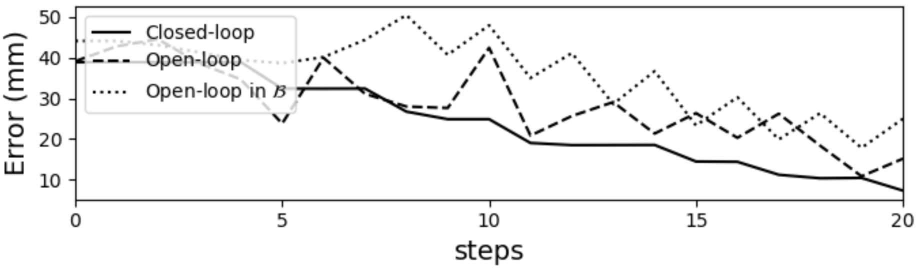

Figure 4.6: Wire shape error during manipulation to the goal $\mathbf{a}_g$ with open (dashed) and closed-loop (solid) control. Dotted curve show the error for an open-loop while tracking a straight line path in $\mathcal{B}$.

Table 4.4: Accuracy of path tracking

	Open-loop (mm)	Closed-loop (mm)
Mean	11.01 ± 5.21	6.22 ± 1.83
Max.	31.51	11.62

cussed in Section 4.1.4 while including the same noise. We have implemented the asymptotically optimal variant of the Rapidly-exploring Random Tree, i.e., RRT* [23]. The RRT* planner ensures that the path from $\mathbf{a}_s$ to $\mathbf{a}_g$ is in $\mathcal{A}_{free}$ while minimizing path length. For a set of ten arbitrary start and goal configurations, we have planned and rolled-out the paths in open and closed loop control. Table 4.4 presents the mean and maximum shape error along the paths. Furthermore, Figures 4.7 and 4.8 show an example of tracking one path. The fine tracking along the planned path with closed-loop control is shown compared to non-smooth and erroneous tracking with open-loop. Snapshots of the motion with pose estimation are shown in Figure 4.9. Here also, the small tracking errors in closed-loop are imposed by accuracy of the F/T sensor calibration. Nevertheless, these experiments validate the ability of the proposed closed-loop control to maintain good tracking along planned paths.

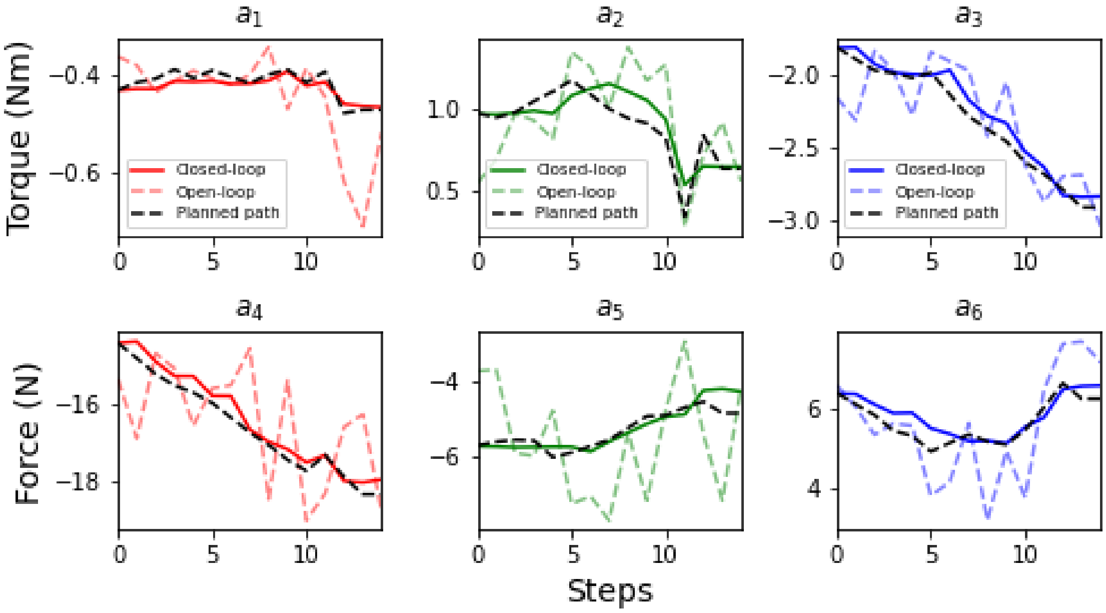

Figure 4.7: Tracking of a path in $\mathcal{A}$ planned using RRT* with open (dashed) and closed-loop (solid) control. Black dashed lines illustrate the planned path to track in $\mathcal{A}$.

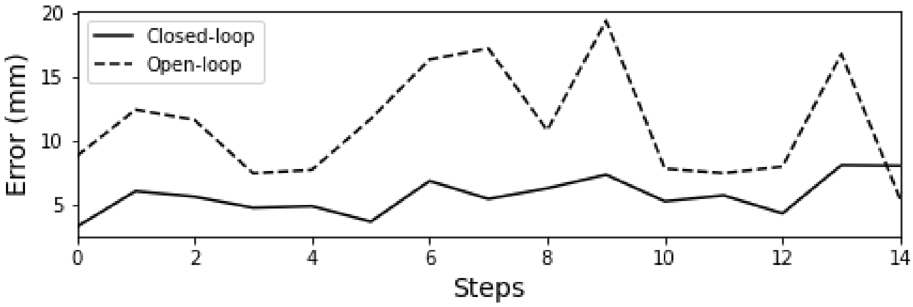

Figure 4.8: Average shape error of the wire during path tracking with open and closed-loop control.

4.2.5 Demonstration

We have conducted a demonstration in which the robot must manipulate the wire through a narrow passage formed by two obstacles. The width of the narrow passage is 50 *mm*. Two start and goal configurations were chosen and a path between them was planned using RRT*. Then, the path was rolled-out with open and closed-loop control in real-time for ten attempts each. Here also, normal noise was included similar to previous sub-section.

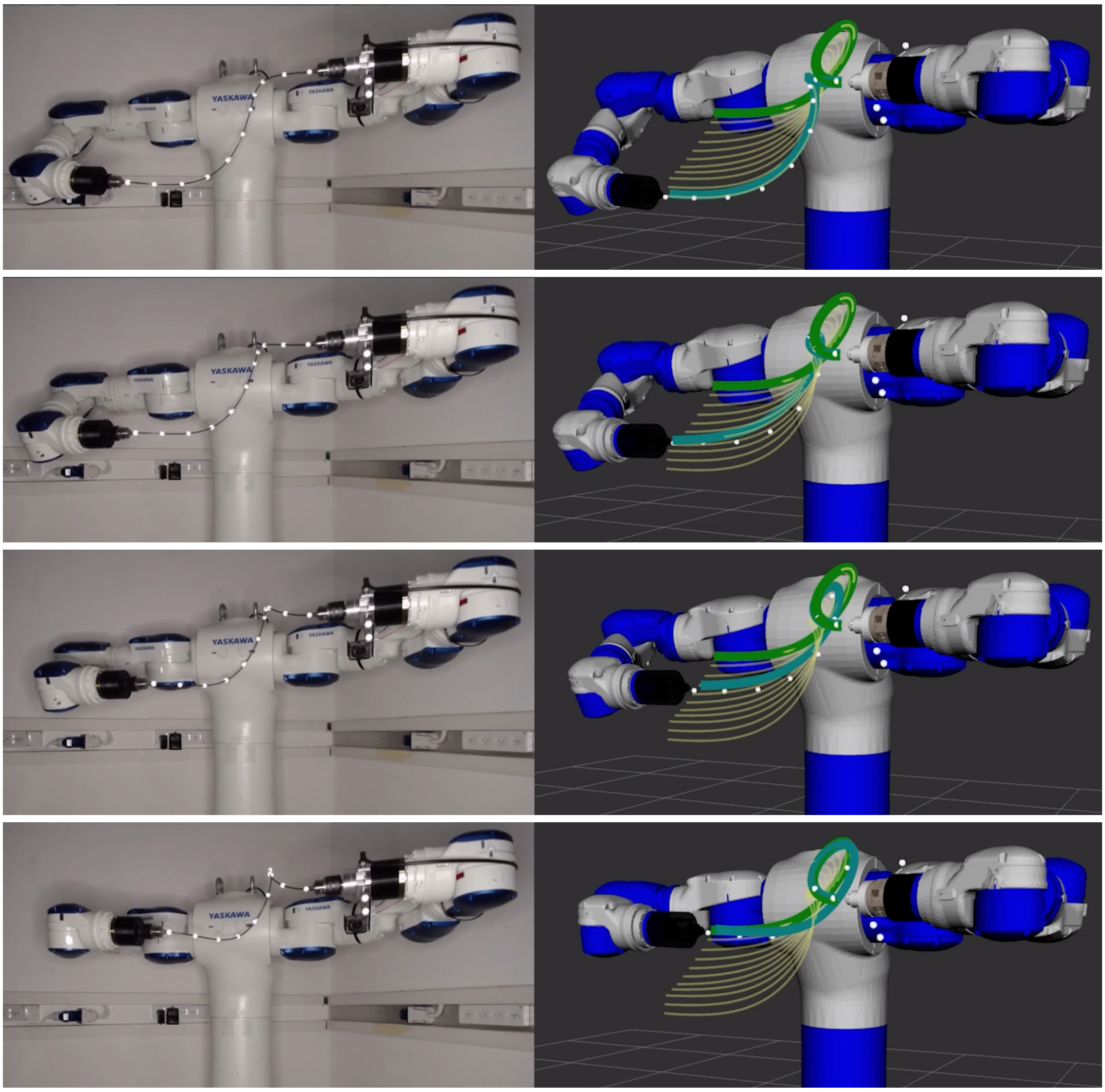

Figure 4.9: The wire is controlled to track a path in $\mathcal{A}_{free}$ planned with RRT*. White circles are the measured markers along the wire and the cyan curve shows the current pose estimation based on F/T sensing. Yellow curves indicate intermediate configurations to pass along the motion while the green curve illustrates the goal. Mean shape error across the manipulation is $6.2mm$.

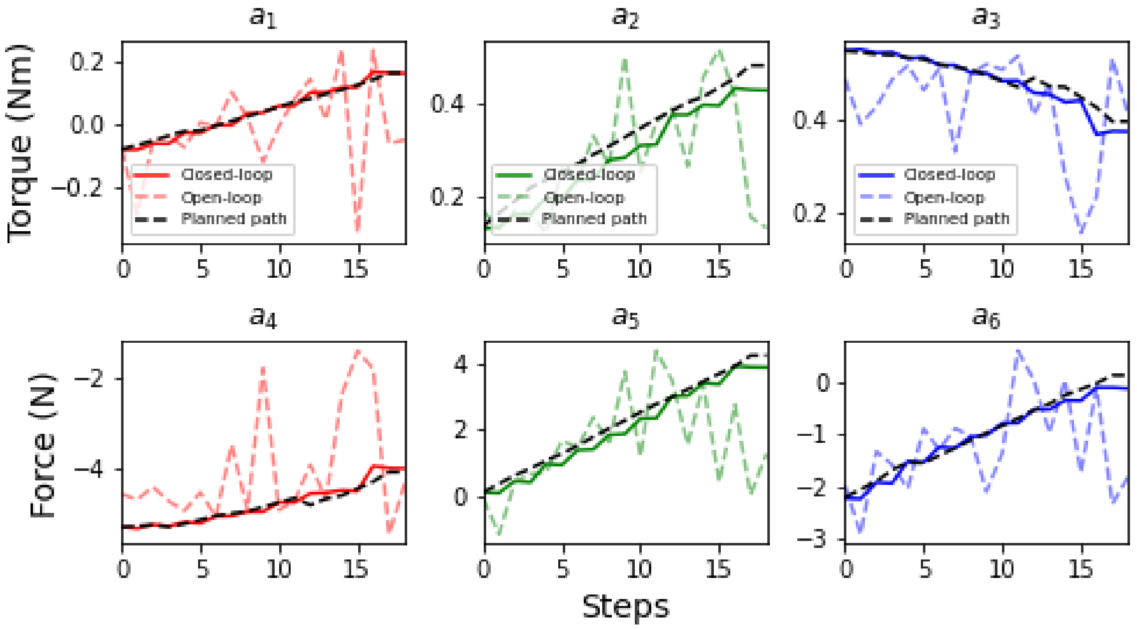

Figure 4.10: Roll-out of the path between the obstacles in open (dashed) and closed-loop (solid) control. Black dashed lines illustrate the planned path to track in $\mathcal{A}$. The roll-out in open-loop collided and the F/T measurements significantly deviated from the planned path in $\mathcal{A}$.

Failure is declared when the wire collides with an obstacle and can be detected when large deviations in the F/T measurements occur in a short period of time. Figures 4.10-4.11 show the path tracking of roll-outs in open-loop with a collision and closed-loop control. Once the wire has collided, the F/T measurements include contact loads and it is no longer possible to estimate the shape of the wire in order to return to the path. Due to better tracking with control, the success rates for tracking the path with open- and closed-loop control are 30% and 90%, respectively. Figure 4.12 shows snapshot of one successful roll-out with closed-loop control. The demonstration results validate the accurate tracking and emphasize the importance of accurate path tracking in cluttered environments.

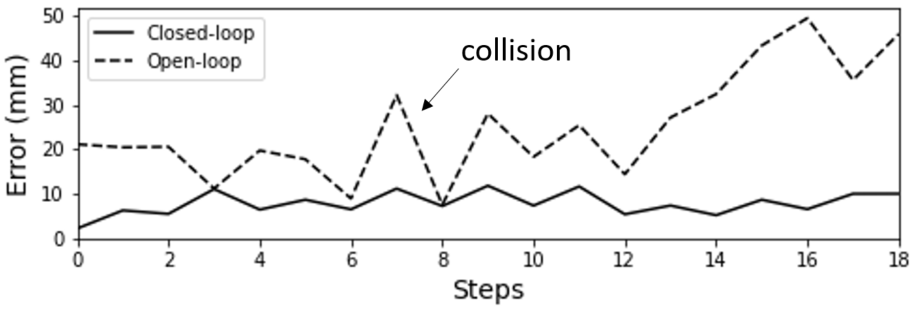

Figure 4.11: Average shape error of the wire during path tracking between the obstacles with open and closed-loop control. The roll-out in open-loop collided and the motion significantly deviated from the planned path.

4.3 Practical considerations

Throughout this study, we drew upon the analytical model developed by Bretl and McCarthy. As the model does not incorporate gravitation, non-linearity of the sensors, or other uncertainties, a Neural-Network (NN) was employed to calibrate the F/T sensor to map between the real sensed loads and the theoretically defined ones. However, such a process has the disadvantage of requiring the solution of the inverse problem for each sample, given a measured shape, to compute the theoretical load. Such a process is computationally complex and may take a very long time to complete. Additionally, once the F/T sensor has been calibrated, it is necessary to repeat the solution of an ordinary differential equations system in order to determine the corresponding shape of the rod. Each solution is computationally expensive and the update frequency remains low [44]. Furthermore, the model assumes super elastic wire, so the system cannot deal with real-world wires, such as the electric wire. In the following chapter, these issues are addressed.

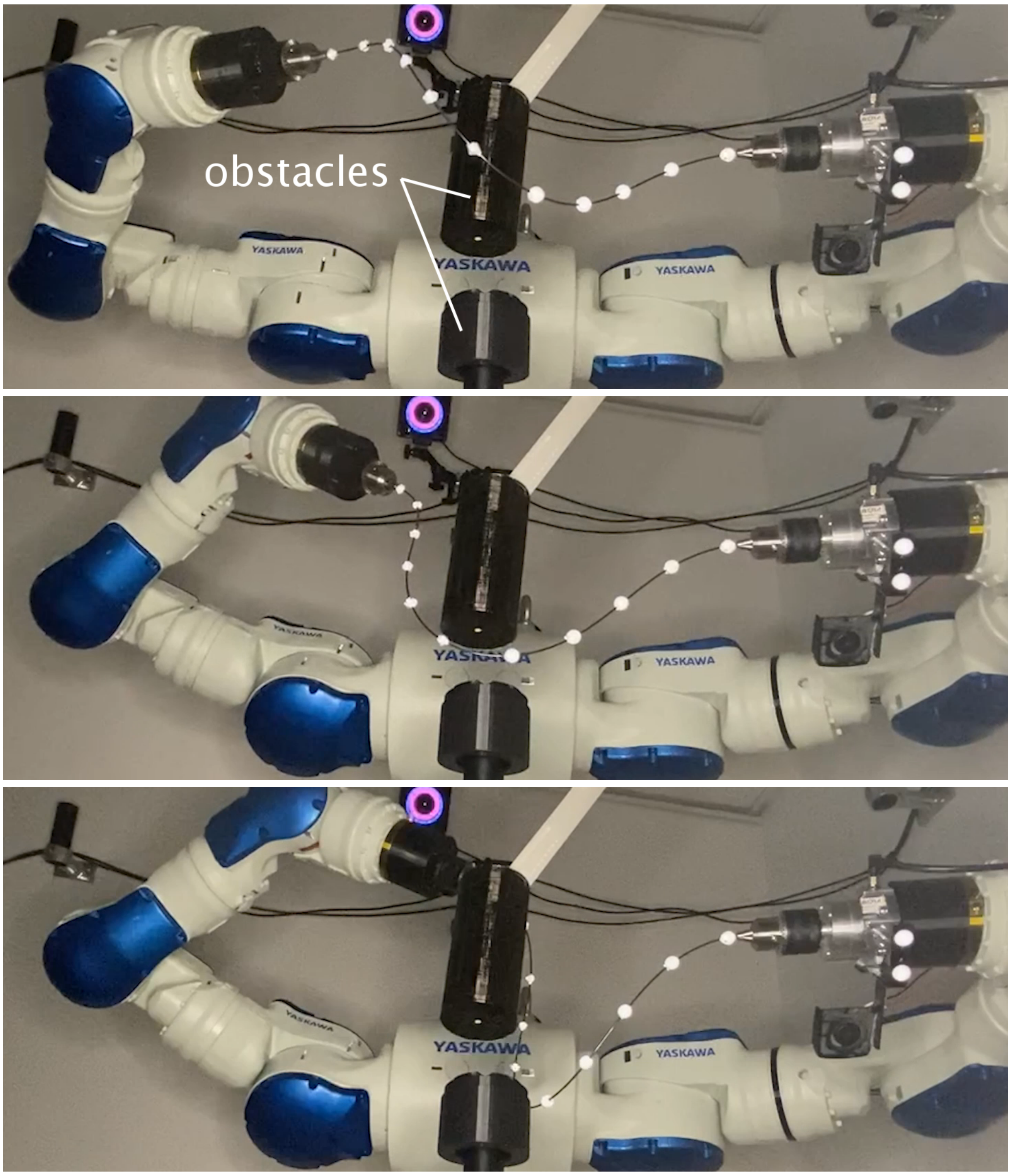

Figure 4.12: Snapshots of the robot manipulating the wire between obstacles in closed-loop control with F/T feedback.

5 Learning Based Framework

In this chapter, we will continue addressing the issues discussed in previous chapter, by using Learning methods to manipulate various wires. We begin by highlighting the challenges we face and describing our methods for addressing them. In addition, we present the results of a series of experiments in which we analyzed the performances of our models. **The work in this chapter was submitted for publication in the *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022.**

5.1 Method

An elastic wire of length L and mechanical coefficients $\mathbf{c} = (c_1, c_2, c_3)^T$ is held by a dual-arm robotic system. Furthermore, a Force/Torque (F/T) sensor is mounted on one arm and measures the load $\overline{\mathbf{a}} \in \mathcal{A}_{FT}$ exerted by the wire where $\mathcal{A}_{FT}$ is the space of measured wire F/T states. We wish to estimate a wire shape given the load. In other words, we explore the learning of a discrete map $\Gamma : \mathcal{A}_{FT} \to \mathbb{R}^3 \times \ldots \times \mathbb{R}^3$. Hence, given a measurement $\overline{\mathbf{a}} \in \mathcal{A}_{FT}$ of the F/T sensor, the map will output a set of m points $\Gamma(\overline{\mathbf{a}}) = \{\overline{\mathbf{p}}_1, \ldots, \overline{\mathbf{p}}_m\}$ along the wire where $\overline{\mathbf{p}}_j = \mathbf{p}(\frac{j}{m}) \in \mathbb{R}^3$.

As discussed in previous chapter, a measurement $\overline{\mathbf{a}}$ of the F/T sensor cannot directly be applied to extract the shape with model $\Phi(\overline{\mathbf{a}})$ (4.2.1). Solving inverse problem (4.8) for each sample is computationally expensive. Furthermore, in order to acquire a fine calibration model, one must collect a large amount of samples and with high variance. Such process may take very long time while insufficient data yield low accuracy as will be demonstrated in the experiments. In this work, we take a different approach where we directly map F/T measurements to wire shapes and, by that, incorporate all uncertainties in the model. Therefore, explicit knowledge of coefficients $\mathbf{c}$ is not required nor the tedious solution of inverse problem (4.8).

5.1.1 Data Collection

Training data is collected by sampling M wire shape along with their corresponding F/T measurements. Sampling a continuous representation of the wire shape can be done in several ways including shape segmentation from RGB cameras or from a depth camera if the wire is thick enough. In this work, a motion capture system tracks V reflective markers fixed along the wire. While the approach provides a coarse resolution along the wire, the marker positions are acquired with high accuracy. Hence, sample j taken from the system is in the form $(\bar{\mathbf{a}}_j, \mathcal{F}_j)$ where $\mathcal{F}_j = \{\bar{\mathbf{p}}_{j,1}, \dots, \bar{\mathbf{p}}_{j,V}\}$.

To provide a finer representation of the wire given $\mathcal{F}_j$, we search for a parametric curve function $\mathbf{f} : \mathbb{R} \to \mathbb{R}^3$ that would represent the wire with higher resolution. In practice, parametric curve

$$\mathbf{f}(t) = (f_x(t), f_y(t), f_z(t))^T \tag{5.1}$$

could be represented by polynomial functions of degree h such that

$$f_x(t) = \sum_{k=0}^{h} s_{x,k} t^k,\ f_y(t) = \sum_{k=0}^{h} s_{y,k} t^k,\ f_z(t) = \sum_{k=0}^{h} s_{z,k} t^k$$

where $s_{j,k}$ are coefficients to be optimized. Curve function $\mathbf{f}(t)$ that best fits points $\mathcal{F}_j$ can be obtained by the least-squares method and is the solution of

$$\min_{\upsilon} \sum_{k=1}^{V} \|\bar{\mathbf{p}}_{j,k} - \mathbf{f}(t)\|^2 \tag{5.2}$$

where υ is the vector of all polynomial coefficients in $\mathbf{f}(t)$. Problem (5.2) can be solved rapidly with parametric curve fitting [16]. In brief, an iterative algorithm associates value t_i to $\bar{\mathbf{p}}_i$ by locally minimizing $\|\bar{\mathbf{p}}_i - \mathbf{f}(t_i)\|^2$ and ensuring that $t_i < t_{i+1}$. With the acquired parametric curve fit, we generate m equally spaced points $\mathcal{P}_j = \{\mathbf{p}_1, \dots, \mathbf{p}_m\}$ along the wire ($m \gg V$). These m points are considered an higher resolution representation of the sampled wire and further used for learning map Γ. In order to acquire this parametric fit, the initial points must be in order, which is not necessarily true. Using the K-Nearest Neighbors algorithm with $K = 1$, we iteratively search for the next marker along the wire that satisfies the distance constraints starting at the base gripper marker. Afterward, we use the arranged markers as inputs for the parametric fit. Finally, the generated training data consists of N samples in the form $\mathcal{D} = \{(\bar{\mathbf{a}}_i, \mathcal{P}_i)\}_{i=1}^{M}$. For each $\bar{\mathbf{a}}_i$ sample, we also record the corresponding pose of the second gripper $\bar{\mathbf{b}}_i$. Hence, we have another dataset $\mathcal{V} = \{(\bar{\mathbf{a}}_i, \bar{\mathbf{b}}_i)\}_{i=1}^{M}$ for motion planning as will be discussed later.

5.1.2 Learning Model: Supervised and Convolutional Autoencoder

A Fully-Connected NN (FC-NN) model can be directly trained with $\mathcal{D}$ to acquire an approximation of Γ. In such case, the input would be six-dimensional while the output is the flattening of $\mathcal{P}$ to a vector of dimension $3m$. However, such vector representation loses the spatial relationship between coordinates along the wire and may affect accuracy. Alternatively, we propose to incorporate convolutional layers to allow the model spatial understanding of the data and parameter sharing. On top of that, we train a Supervised Autoencoder (SAE) model [32] to augment the learning and acquire an inverse solution Γ^{-1} along the way. Hence, we further describe the architecture of the proposed Supervised and Convolutional Autoencoder (SnCAE) to learn wire shape representation.

A standard Autoencoder (AE) is a neural-network aimed to find a lower-dimensional embedding of some data, i.e., dimensionality reduction. AE is trained to reconstruct the input at the output through an encoder and a decoder. The encoder is used to identify embedded information in the data and compress it to a latent representation $\mathbf{z} \in \mathbb{R}^d$ where dimension d is lower than the one of the input data. The decoder, on the other hand, reconstructs the original data from the latent representation. AE is normally trained to reconstruct input data X by minimizing the objective function $\|X - X'\|^2$ where X' is the output of the decoder. AE is capable of learning complex non-linear relations where simpler models of dimensionality reduction under-perform.

As mentioned, we preserve the spatial representation of the wire by having the input and reconstructed output to the encoder and decoder, respectively, as $m \times 3$ arrays (each row is a point along the wire). Therefore, the input to the encoder passes through a convolution with an $m_c \times 1$ kernel yielding a convolutional layer of size $m \times 3 \times m_a$. The data then passes through a set of fully-connected layers as seen in Figure 5.1. The latent space is six-dimensional to match the size of the F/T state. The encoder and decoder are mirrored while the output of the decoder goes through a de-convolutional layer of size $m_c \times 1$. m_a, m_b and m_c are hyper-parameters to be further optimized. Therefore, the reconstruction loss is given by

$$\mathcal{J}_r = \|\mathcal{P} - \mathcal{P}'\|^2. \tag{5.3}$$

While the AE is an unsupervised method, SAE is a variation of AE where the model also supervises the representation of the latent space. Based on prior work discussed above, the latent representation of a wire shape is six-dimensional. Hence, we set the latent layer in the AE to be $d = 6$, i.e. $\mathbf{z} \in \mathbb{R}^6$. Furthermore, we add a soft constraint on the latent space to minimize the distance to $\bar{\mathbf{a}} \in \mathcal{A}_{FT}$. Hence, we formulate a latent loss value

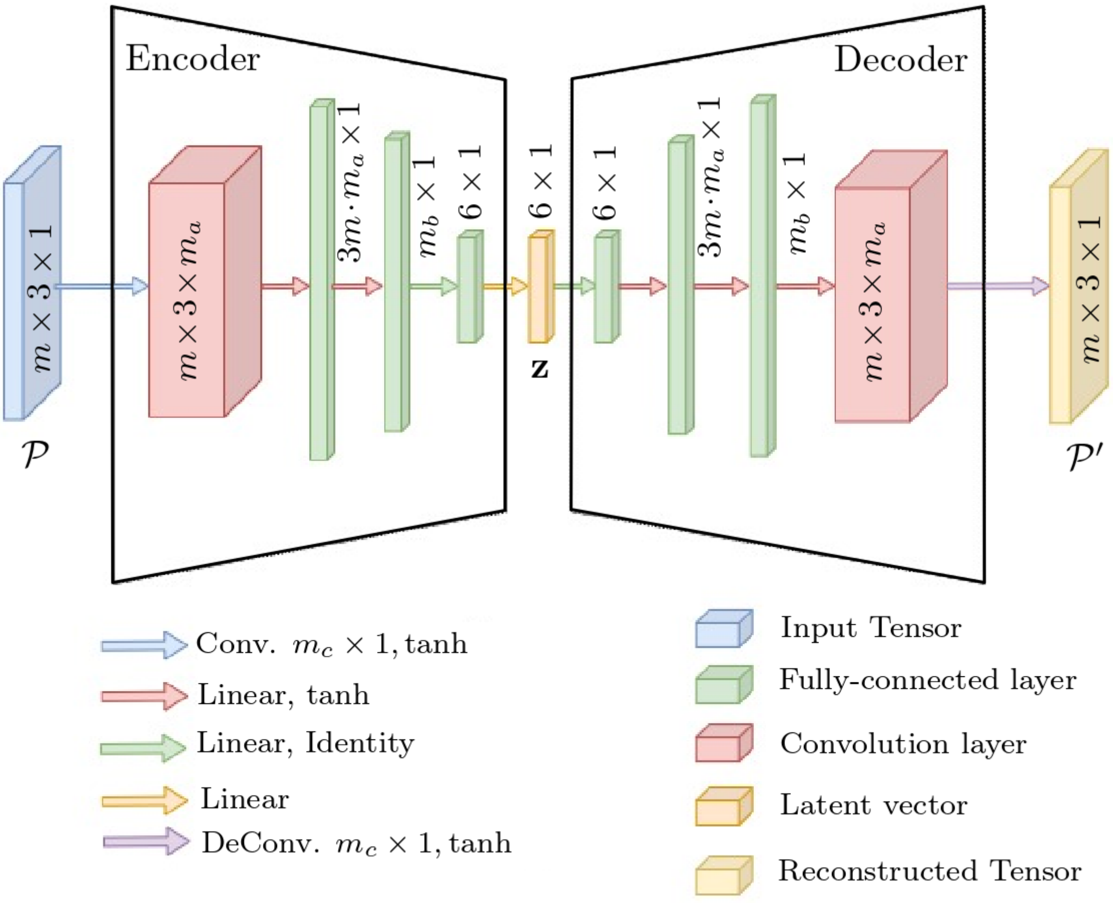

Figure 5.1: Convolutional layers constructing the encoder and decoder

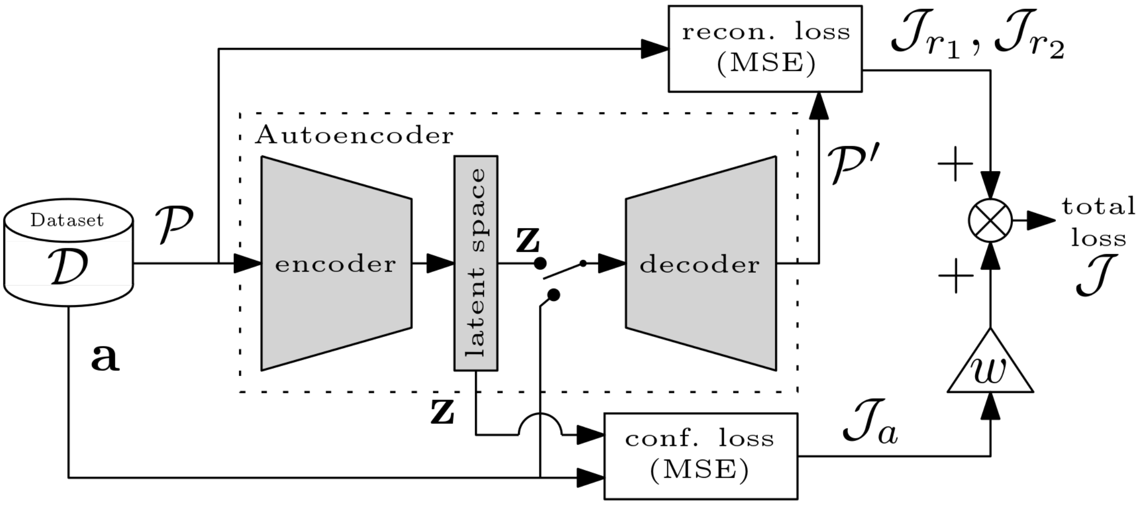

Figure 5.2: Framework of the Supervised and Convolutional Autoencoder (SnCAE).

in the form of

$$\mathcal{J}_a = \|\mathbf{z} - \bar{\mathbf{a}}\|^2. \tag{5.4}$$

Additionally and for regularization, for each batch of data, we compare the reconstruction of $\mathcal{P}$ with either using $\mathbf{z}$ or $\bar{\mathbf{a}}$ in the decoder. Consequently, we train the SnCAE to minimize a combined loss function

$$\mathcal{J} = \mathcal{J}_{r_1} + \mathcal{J}_{r_2} + w\mathcal{J}_a \tag{5.5}$$

where $\mathcal{J}_{r_1}$ and $\mathcal{J}_{r_2}$ are the reconstruction loss when applying $\mathbf{z}$ and $\bar{\mathbf{a}}$ to the decoder, respectively. Scalar $w > 0$ is a tunable weight. An illustration of the SnCAE is given in Figure 5.2. Preliminary analysis has shown that adding $\mathcal{J}_{r_2}$ to the loss improves accuracy by approximately 25%. A trained SnCAE with minimal loss $\mathcal{J}$ can reconstruct wire shape data while giving a physical and practical meaning to the latent space.

A trained SnCAE has two usages. First, the decoder is the approximated mapping Γ which maps a measured $\bar{\mathbf{a}}_i$ to the spatial shape of the wire $\mathcal{P}_i$. In other words, we can use the decoder as a shape estimator based on F/T sensing at the base gripper. Furthermore, the encoder provides a fast inverse solution instead of problem (4.8), i.e., $\bar{\mathbf{a}}_i = \Gamma^{-1}(\mathcal{P}_i)$. Therefore, the encoder can be exploited to estimate the F/T load at the base gripper, i.e., wire state in $\mathcal{A}_{FT}$, based on a measured shape of the wire. We note that an NN architecture similar to the decoder, termed Decoder CNN (D-CNN) can be trained independently. However, the SnCAE provides both shape estimation and inverse model within the same training. The D-CNN will also be analyzed in the experiments.

5.1.3 Motion planning over F/T states

Given $\mathbf{a}_i \in \mathcal{A}$ and its corresponding second gripper pose $\mathbf{b}_i \in \mathcal{B}$. We define an homogeneous transformation matrix $M \in SE(3)$ with $\delta\mathbf{x} \in \mathbb{R}^3$ and exponential coordinates $\mathbf{y} \in \mathbb{R}^3$ and $\delta\theta \in [0, \pi)$ such that

$$M(\delta\mathbf{b}) = \begin{bmatrix} e^{\mathbf{y}\delta\theta} & \delta\mathbf{x} \\ \mathbf{0} & 1 \end{bmatrix} \text{ for } \delta\mathbf{b} = \begin{pmatrix} \mathbf{y}\delta\theta \\ \delta\mathbf{x} \end{pmatrix}. \tag{5.6}$$

Matrix M is defined to map between two configurations in $\mathcal{B}$ such that

$$\mathbf{b}_{i+1} = \mathbf{b}_i M(\delta\mathbf{b}) \tag{5.7}$$

where perturbation to $\mathbf{b}_{i+1}$ will result in wire configuration $\mathbf{a}_{i+1}$. From Theorem 7 and equation (37) in [13], we get that

$$\delta\mathbf{b} \approx \mathbf{J}(L)\delta\mathbf{a} \tag{5.8}$$

where $\delta\mathbf{a} = \mathbf{a}_{i+1} - \mathbf{a}_i$. Matrix $\mathbf{J}(L)$ is the Jacobian relying on the solution of the above ordinary differential equations and is detailed in [44, 36]. Equation (5.8) states that matrix $\mathbf{J}(L)$ contains information about the relationship between small changes in $\mathcal{A}$ and small changes in $\mathcal{B}$. Therefore and given a desired $\mathbf{a}_{i+1}$ in the vicinity of $\mathbf{a}_i$, the required perturbation $\delta\mathbf{a}$ in $\mathcal{A}$ can be obtained. Then, by solving (5.8), one can use map (5.7) to compute the required perturbation in $\mathcal{B}$ in order to move a wire from configuration $\mathbf{a}_i$ to $\mathbf{a}_{i+1}$.

A data-based approach and a learned map Γ do not enable the extraction of an explicit representation of the Jacobian. Furthermore, the non-linearity of the F/T sensor does not guarantee that perturbation $\delta\bar{\mathbf{a}}$ can be mapped to the same $\delta\bar{\mathbf{b}}$ for any $\bar{\mathbf{a}}_i \in \mathcal{A}_{FT}$. Alternatively, we propose to directly learn mapping from $\bar{\mathbf{a}}_i$ to $\bar{\mathbf{b}}_i$, i.e., $\Lambda : \mathcal{A}_{FT} \to \mathcal{B}$. Once having map Λ, we can extract the required gripper perturbation $\delta\bar{\mathbf{b}}_i$ from $\bar{\mathbf{b}}_i = \Lambda(\bar{\mathbf{a}}_i)$ to $\bar{\mathbf{b}}_{i+1} = \Lambda(\bar{\mathbf{a}}_{i+1})$ according to (5.6)-(5.7). With dataset $\mathcal{V}$, an FC-NN is trained to learn map $\bar{\mathbf{b}}_i = \Lambda(\bar{\mathbf{a}}_i)$.

Matrix $\bar{\mathbf{b}}_i \in SE(3)$ contains a rotation matrix $R_{\bar{\mathbf{b}}_i}$ and the position vector. While the latter is easy to encode, direct encoding of rotation matrices for NN training cannot be done while maintaining orthonormality. Hence, we encode $\bar{\mathbf{b}}_i$ with a nine-dimensional vector by flattening the position vector along with two columns $(\mathbf{v}_1, \mathbf{v}_2)$ of the matrix. Reconstruction of the rotation matrix $R_{\bar{\mathbf{b}}_i}$ given output $(\mathbf{v}_1, \mathbf{v}_2)$ from the NN is done using the GramâSchmidt process [54] according to

$$\begin{aligned}
\mathbf{w}_1 &= \frac{\mathbf{v}_1}{\|\mathbf{v}_1\|} \\
\mathbf{u}_2 &= \mathbf{v}_2 - (\mathbf{w} \cdot \mathbf{v}_2)\mathbf{w}_1 \\
\mathbf{w}_2 &= \frac{\mathbf{u}_2}{\|\mathbf{u}_2\|} \\
R_{\bar{\mathbf{b}}_i} &= [\mathbf{w}_1 \;\; \mathbf{w}_2 \;\; \mathbf{w}_1 \times \mathbf{w}_2] \,.
\end{aligned}$$

Model Λ could be used to plan motion to a desired goal as follows. Given a wire shape goal $\mathcal{P}_g$, the F/T state goal is the solution of the inverse problem $\bar{\mathbf{a}}_g = \Gamma^{-1}(\mathcal{P}_g)$ computed with the encoder of the SnCAE. A motion planner would output a continuous path $\gamma : [0,1] \to \mathcal{A}_{FT}$ from the current start $\gamma(0) = \bar{\mathbf{a}}_s \in \mathcal{A}_{FT}$ to a desired one $\gamma(1) = \bar{\mathbf{a}}_g \in \mathcal{A}_{FT}$. Furthermore, decoder mapping Γ is used as a collision checker where shapes of candidate F/T states are validated to be collision free. We check wire collision with obstacles or with the robot and wire self-collision. A sampling-based motion planner can then be used for finding a collision-free path [31]. In this work, we employ the asymptotically optimal variant of the Rapidly-exploring Random Tree, i.e., RRT* [23]. The RRT* planner finds a path from $\bar{\mathbf{a}}_s$ to $\bar{\mathbf{a}}_g$ while minimizing path length in $\mathcal{A}_{FT}$. Once path γ is acquired, a step from $\bar{\mathbf{a}}_i$ to $\bar{\mathbf{a}}_{i+1}$ along the path is translated to perturbation command $\delta\bar{\mathbf{b}}_i$ of the second gripper from $\bar{\mathbf{b}}_i = \Lambda(\bar{\mathbf{a}}_i)$.

5.2 Experiments

5.2.1 Shape estimation analysis

We evaluate shape estimation on two wires seen in Figure 5.3: a Nitinol wire of 2 mm diameter and 820 mm length, and a standard electrical wire of 3 mm diameter and 500 mm length. The electrical wire does not meet the ground assumptions of [13] and was shown in [36] to yield large approximation errors when using the analytical model. Data was collected as discussed in Section 5.1.1 while sampling $M = 26,187$ and $M = 15,400$ Nitinol and electric wire shapes, respectively, and their corresponding F/T measurements. For each sample, a $h = 6$ degree polynomial function was fitted and $m = 100$ equally spaced points were generated along it. Training set $\mathcal{D}$ is, therefore, comprised of F/T measurements and their corresponding polynomial approximation of the shape. In addition, test sets were collected independent of the training set and included approximately $1,200$ samples. Shape estimation error is defined as the Root-Mean-Square-error (RMSE) between

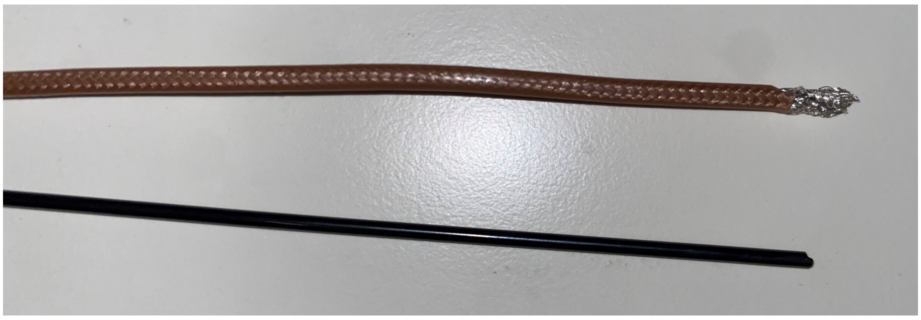

Figure 5.3: Electrical (top) and Nitinol (bottom) wires used in the experiments.

Table 5.1: Results for wire shape estimation using various models

Model	Mean error (mm)	Comp. time (msec)
Analytical model [36]	37.14±16.48	72.22±11.40
FC-NN	12.16±5.26	0.18±0.19
SnCAE (with Decoder)	10.70±0.25	1.83±5.26
D-CNN	11.20±0.17	1.72±5.07

measured marker positions of a test shape and the closest points of the predicted polynomial shape (based on corresponding F/T measurement).

We first analyze the shape estimation of the Nitinol wire. Using dataset $\mathcal{D}$, we train the proposed SnCAE model along with FC-NN and D-CNN models for comparison. In addition, we also provide results for the analytical model as implemented in [36]. Hence, optimization problem (4.8) was solved for each sample in $\mathcal{D}$ taking approximately 48 hours in total. With the solutions, we train NN model ψ to map F/T measurements to theoretical wire configurations enabling the solution of $\mathbf{p}(t) = \Phi(\psi(\bar{\mathbf{a}}_i))$. On the other hand, the other NN models were optimized to yield the lowest shape estimation error. The optimal FC-NN model is composed of one hidden layer of 119 nodes, Rectified Linear Unit (ReLU) activation function and a regularizer of 8.8×10^{-7}. The optimal hyper-parameters of the SnCAE model are $m_a = 16$, $m_b = 480$ and $m_c = 10$.

Table 5.1 summarizes the results of all methods for the Nitinol wire including shape estimation accuracy and mean computation time of an individual wire shape estimation. Computation time was evaluated on an Intel-Core i7-8700 Ubuntu machine with 16GB of RAM. First, the analytical model not only takes a long period of time to process data, it provides inferior results in terms of accuracy. For the model to provide better accuracy as demonstrated in [36], much more data is required for training calibration model ψ with the cost of days more of computation. Also, the average computation time for one shape estimation is rather large, fits to the time reported in [44] and is limited in real-time

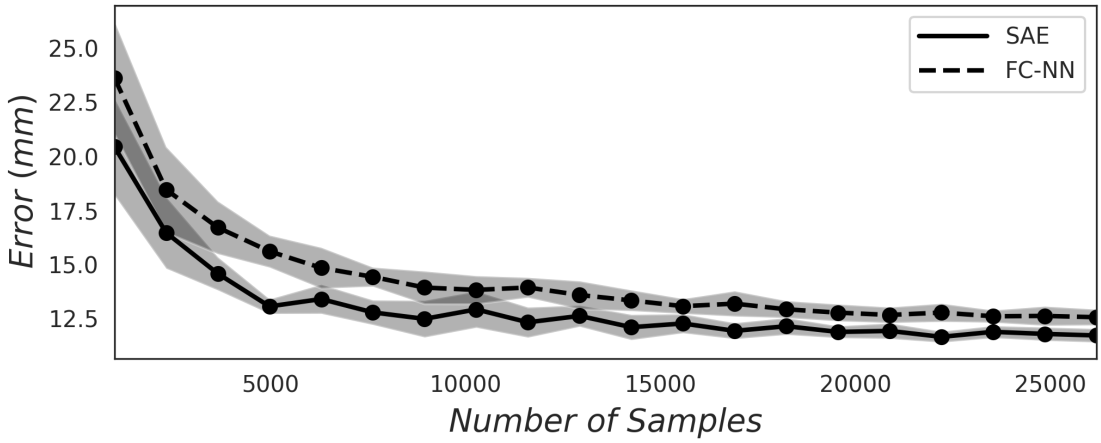

Figure 5.4: Shape estimation accuracy with regards to the number of collected samples and for FC-NN and SnCAE.

motion planning. On the other hand, with the same amount of data, a NN can provide a much lower error as the results indicate. Furthermore, SnCAE is shown to provide lower errors in average compared to the FC-NN with a much smaller standard deviation. D-CNN also provides a fairly good accuracy. Computation times of shape estimation for all data-based models are much faster by at least an order of magnitude compared to the analytical model. Thus, they are far more suitable for real-time applications.

Figures 5.4 and 5.5 show the shape estimation error of the Nitinol and Electrical wires, respectively, for SnCAE and FC-NN with regards to the number of samples in $\mathcal{D}$. The error for each number of samples was cross validated over 20 sequential data batches taken randomly from the entire training set. While the improvement of SnCAE over FC-NN is marginal for FC-NN, the improvement is much larger for the electrical wire. Overall, the results show that SnCAE outperforms FC-NN while enabling low errors for a relatively small amount of data. The electrical wire is softer than the Nitinol and, therefore, the magnitude of the F/T signals are smaller and more affected by noise making it harder to learn. Nevertheless, the mean error of SnCAE is rather small (approximately 22 mm). In addition, one can settle for half of the data and acquire almost the same accuracy. In a collision checker and motion planning setting, a safety distance would be taken from the wire that is larger than such accuracy. Figures 1.2 and 5.6 show examples of shape estimation with SnCAE in real-time for the Nitinol and electrical wires, respectively.

Table 5.2 presents the F/T errors of the inverse problem solved with the Encoder of the SnCAE. The encoder is evaluated over the test data mapping measured wire shapes to F/T states. For reference, maximal absolute force and torque measured with the Nitinol wire

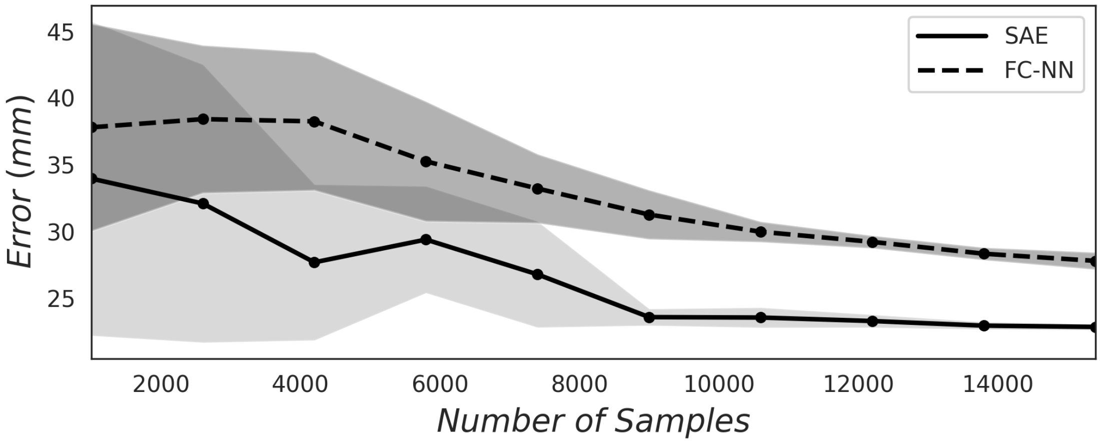

Figure 5.5: Shape estimation accuracy with regards to the number of collected samples and for FC-NN and SnCAE.

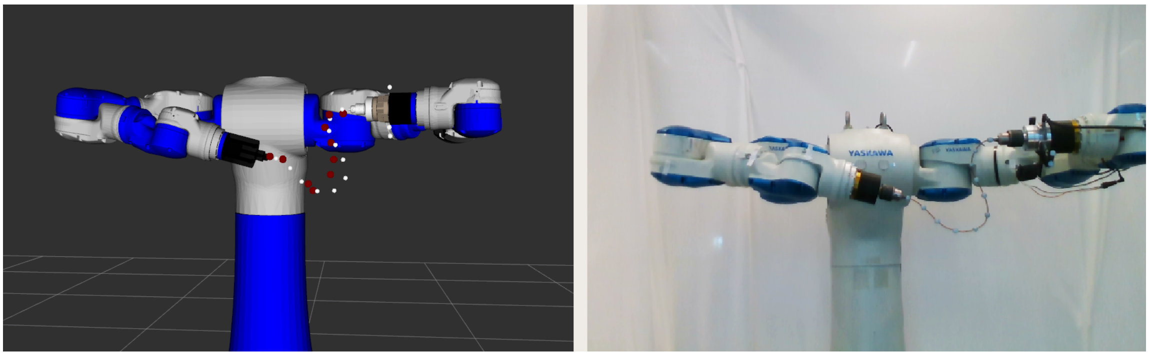

Figure 5.6: Shape estimation of a standard electrical wire using only a Force/Torque sensor on one gripper. The wire configuration on the right is approximated and visualized in simulation (red markers), and compared to markers (white markers) tracked with a motion capture system..

Table 5.2: Accuracy of inverse solution with the S*n*CAE encoder

Wire	Force error (N)	Torque error (Nm)
Nitinol	0.34±0.28	0.038±0.032
Electric	0.52±0.11	0.051±0.016

Table 5.3: Roll-out errors along planned paths

Error	Shape (mm)	Force (N)	Torque (Nm)
Goal reach	4.5±2.9	0.23±0.18	0.041±0.027
Path tracking	6.1±3.3	0.36±0.29	0.066±0.055

are 6.27 N and 1.06 Nm, respectively, and for the electrical wire, 3.10 N and 0.3 Nm. The results, therefore, show good F/T estimation allowing mapping desired shapes to states for motion planning, as discussed in the next section.

5.2.2 Path planning

We conduct an experiment in which the robot must manipulate the Nitinol wire to some desired shape. Map Λ is implemented by training a standard fully-connected NN as discussed in Section 5.1.3. Furthermore, goal shapes $\mathcal{P}_g$ are randomly sampled by manually placing the cable without the robot along desired shapes and recording using the motion capture system. A goal state is computed with the encoder according to $\bar{\mathbf{a}}_g = \Gamma^{-1}(\mathcal{P}_g)$. An RRT* is then implemented to plan in $\mathcal{A}_{FT}$ while using Γ for collision checking and Λ for moving the robot along the path.

We analyze roll-outs along ten planned paths to various goals. At each trial, we plan motion from the current pose to a chosen goal shape. Once planned, the path is rolled-out in open-loop by exerting the computed sequence of gripper poses. Table 5.3 summarizes the results for roll-out accuracy both for tracking the path and reaching the desired goal. Note that a shape tracking error refers to the RMSE of the markers relative to the planned shapes of the corresponding steps along the path. Therefore, the results show high accuracy tracking and goal reaching. A path was planned with the accuracy of the SnCAE decoder. However, roll-outs are dependent on the accuracy of the corresponding gripper pose $\bar{\mathbf{b}}$ acquired by learned model Λ. Hence, the accurate model Λ enabled the good tracking of paths without regards to the accuracy of the decoder that planned them. Figure 5.7 shows snapshots of one roll-out while the corresponding path tracking in $\mathcal{A}_{FT}$ is illustrated in Figure 5.8. Tracking is seen to be accurate both in shapes and in $\mathcal{A}_{FT}$.

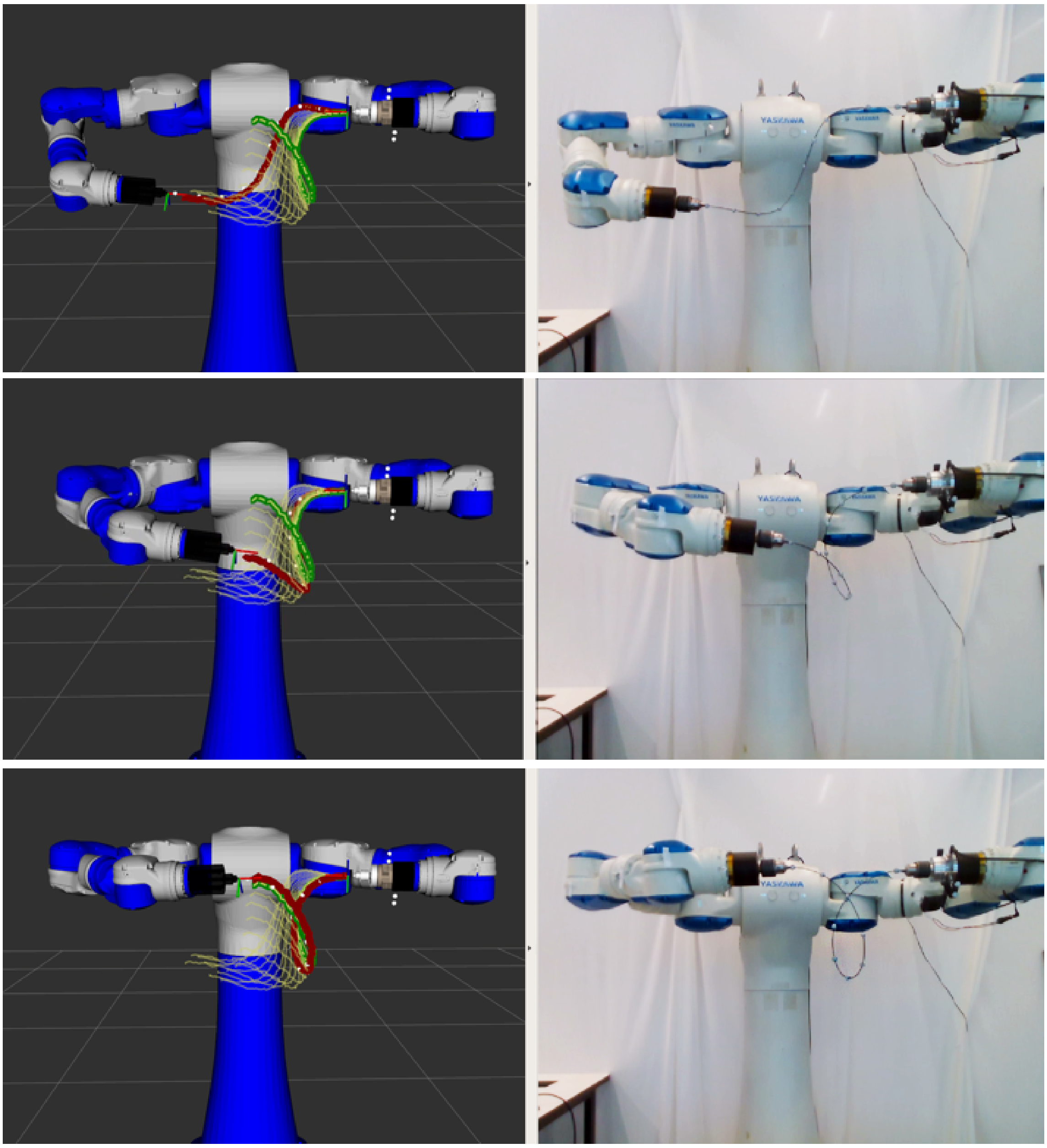

Figure 5.7: The wire is manipulated along a planned path (yellow curves) in $\mathcal{A}_{FT}$ from (top) start to (bottom) goal. The shape of the wire is estimated in real-time (red curve) using the decoder of the SnCAE and motion of the gripper is determined according to model Λ.

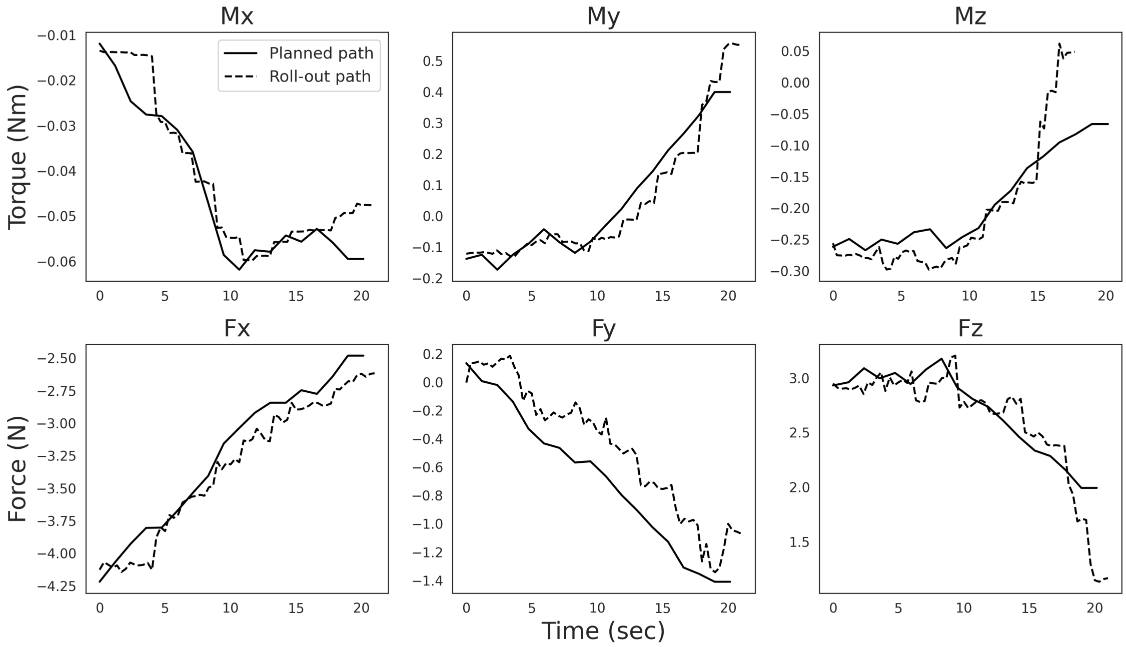

Figure 5.8: Roll-out of a planned path in $\mathcal{A}_{FT}$ corresponding to the motion in Figure 5.7.

Figure 5.9 shows a demonstration of planning a path in the presence of a cylindrical obstacle. The location of the obstacle was detected with markers and a motion capture system. Motion to the same goal was conducted five times while starting from different states. All roll-outs were successful and the wire did not collide.

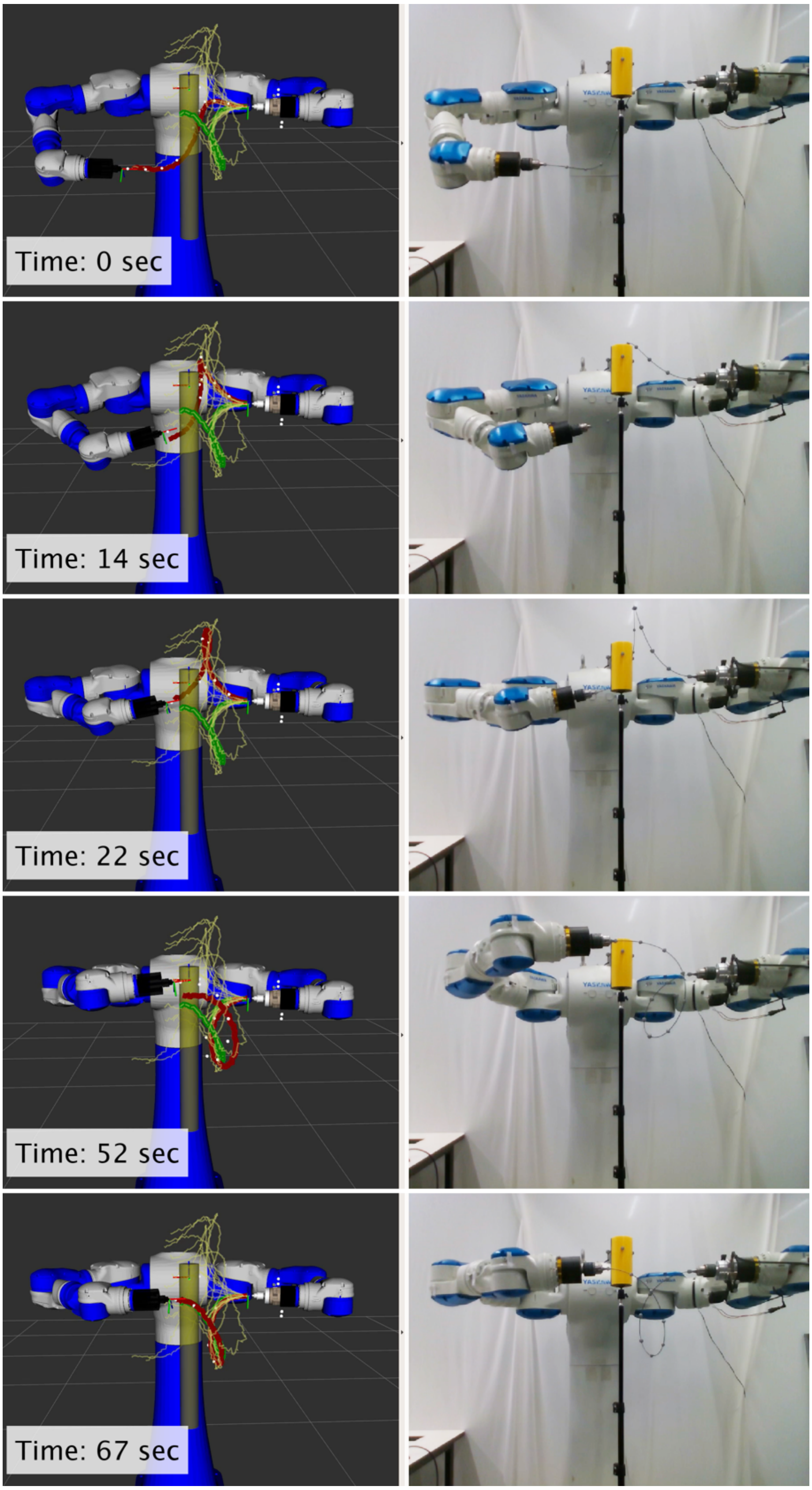

Figure 5.9: Roll-out of a planned path (yellow curves) across an obstacle (yellow cylinder).

6 Conclusion

We have addressed the problem of estimating the shape of a wire solely based on F/T measurements at the base gripper. At first, a complete framework has been proposed where the mechanical properties of the wire are rapidly approximated and the wire can be accurately manipulated with closed-loop control. The shape of the wire is controlled in the configuration space of the wire rather then in space of gripper poses. In such way, the manipulation can be performed without explicit information on the spatial shape of the wire. The experiments have shown that, indeed, accurate manipulations can be performed without complex visual perception. We have also evaluated the control of the wire in $\mathcal{A}$. Open-loop control has been shown to be feasible when the robotic arms are accurate. Nevertheless, closed-loop control maintains accurate manipulations even when the accuracy of the arms is low.

To overcome issues when using the analytical model, such as computational complexity and wires generalization, we proposed training a NN model based on F/T measurements exerted on a robot arm by a wire. An autoencoder based model, termed SnCAE, was presented where convolutional layers were used to maintain the spatial shape of the wire and the latent space was forced to resemble the space of F/T measurements. Then, the trained encoder and decoder are used for mapping wire shape to F/T state and vice versa, respectively. Generally, the evaluated data-based models (decoder of SnCAE and FC-NN) outperformed the analytical approach proposed in prior work for shape estimation, as well as being much more sources efficient. While the SnCAE gained moderate accuracy improvement over the FC-NN for the two cables, it has provided an added value with the encoder. The encoder was shown to be able to solve the inverse problem and identify goal F/T states from given shapes. Furthermore, the results show that sufficient accuracy can be achieved with a relatively small amount of samples. Along with a Nitinol wire, we have demonstrated, for the first time, the ability to predict the shape of an electrical wire with an accuracy that is feasible for real applications. With another NN model trained with the same collected data, we can map F/T states to desired gripper poses. Hence, a motion planner was implemented to plan and roll-out collision-free paths. A set of experiments was shown to validate pose estimation and planning accuracy.

Our proposed method indeed provides a model for the trained wire. The model cannot be applied, for instance, to a wire of different length. Therefore, future work may involve the generalization of a NN model to wires of various lengths, materials and stiffnesses. In addition, some data augmentation can be performed to reduce the number of real samples required. Alternatively, data collected from a physics engine along with domain randomization may provide sufficient generalization to various wires. Future work may also combine visual perception and F/T measurement to reduce estimation inaccuracies. Additionally, an algorithmic solution is required in order to detect the location of collisions in order to retract properly.

References

[1] Bota systems gitlab. URL https://gitlab.com/botasys/bota_driver.

[2] Force torque calibration utilitize package. URL http://wiki.ros.org/force_torque_sensor_calib.

[3] Learning models development. URL https://gitlab.com/imishani/mish_motoman/-/tree/master/learn_rod_path.

[4] Analytical models and development, . URL https://gitlab.com/imishani/mish_motoman/-/tree/master/motoman_cable.

[5] Robot motions and visualization, . URL https://gitlab.com/imishani/mish_motoman/-/tree/master/motoman_gazebo.

[6] Motoman github in ros-industrial, . URL https://github.com/ros-industrial/motoman.

[7] S.S. Antman. *Nonlinear Problems of Elasticity*. Springer, 2005.

[8] Y. Asano, H. Wakamatsu, E. Morinaga, E. Arai, and S. Hirai. Deformation path planning for manipulation of flexible circuit boards. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5386–5391, Oct 2010. doi: 10.1109/IROS.2010.5651132.

[9] Federico Augugliaro, Emanuele Zarfati, Ammar Mirjan, and Raffaello D'Andrea. Knot-tying with flying machines for aerial construction. In *IEEE/RSJ Inter. Conference on Intelligent Robots and Systems*, pages 5917–5922, 2015.

[10] M. Bell and D. Balkcom. Knot tying with single piece fixtures. In *Proc. IEEE Int. Conf. on Rob. and Aut.*, pages 379–384, May 2008.

[11] Andy Borum, Dennis Matthews, and Timothy Bretl. State estimation and tracking of deforming planar elastic rods. In *IEEE International Conference on Robotics and Automation*, pages 4127–4132, September 2014.

[12] Timothy Bretl and Zoe McCarthy. *Equilibrium Configurations of a Kirchhoff Elastic Rod under Quasi-static Manipulation*, pages 71–87. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. ISBN 978-3-642-36279-8.

[13] Timothy Bretl and Zoe Mccarthy. Quasi-static manipulation of a kirchhoff elastic rod based on a geometric analysis of equilibrium configurations. *Int. Jou. of Robotics Research*, 33(1):48–68, 2014. ISSN 0278-3649.

[14] Simon Duenser, Roi Poranne, Bernhard Thomaszewski, and Stelian Coros. Robocut: Hot-wire cutting with robot-controlled flexible rods. *ACM Transactions on Graphics*, 39(4), 2020.

[15] Russell Gayle, Paul Segars, Ming C. Lin, and Dinesh Manocha. *Path planning for deformable robots in complex environments*, volume 1, pages 225–232. MIT Press Journals, 2005. ISBN 9780262701143.

[16] M. Grossman. Parametric curve fitting. *The Computer Journal*, 14(2):169–172, 01 1971.

[17] Tomas Hermansson, Robert Bohlin, Johan S. Carlson, and Rikard SÃ¶derberg. Automatic assembly path planning for wiring harness installations. *Journal of Manufacturing Systems*, 32(3):417 – 422, 2013.

[18] A. Sintov I. Mishani. Motion capture system communication package. URL `https://gitlab.com/imishani/mish_motoman/-/blob/master/natnet_ros/natnet_ros/scripts/client`.

[19] R. C. Jackson and M. C. ÃavuÅoÄlu. Needle path planning for autonomous robotic surgical suturing. In *Proc. IEEE Int. Conf. on Rob. and Aut.*, pages 1669–1675, 2013.

[20] Shervin Javdani, Sameep Tandon, Jie Tang, James F. O'Brien, and Pieter Abbeel. Modeling and perception of deformable one-dimensional objects. In *IEEE Inter. Conf. on Robotics and Automation*, pages 1607–1614, 2011.

[21] Xin Jiang, Kyong-Mo Koo, Kohei Kikuchi, Atsushi Konno, and Masaru Uchiyama. Robotized assembly of a wire harness in a car production line. *Advanced Robotics*, 25(3-4):473–489, 2011.

[22] Ilknur Kabul, Russell Gayle, and Ming C. Lin. Cable route planning in complex environments using constrained sampling. In *Proc. of the ACM Symposium on Solid and Physical Modeling*, pages 395–402, NY, 2007. ISBN 978-1-59593-666-0.

[23] Sertac Karaman and Emilio Frazzoli. Sampling-based algorithms for optimal motion planning. *International Journal of Robotics Research*, 30(7):846–894, June 2011. ISSN 0278-3649.

[24] L. E. Kavraki, P. Svestka, J. C. Latombe, and M. H. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Tran. Rob. and Aut.*, 12(4):566–580, 1996.

[25] J. Kennedy and R. Eberhart. Particle swarm optimization. In *International Conference on Neural Networks*, volume 4, pages 1942–1948, 1995.

[26] Donal E. Kirk. *Optimal Control Theory*. Dover Publications Inc., 1998.

[27] Ryo Kondo, Takashi Kanai, and Ken-ichi Anjyo. Directable animation of elastic objects. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '05, pages 127–134, New York, NY, USA, 2005. ACM. ISBN 1-59593-198-8.

[28] K. Kosuge, H. Yoshida, T. Fukuda, M. Sakai, and K. Kanitani. Manipulation of a flexible object by dual manipulators. In *Proc. IEEE Int. Conf. on Rob. and Aut.*, volume 1, 1995.

[29] Florent Lamiraux and Lydia E. Kavraki. Planning paths for elastic objects under manipulation constraints. *International Journal of Robotics Research*, 20:188–208, 2001.

[30] S.M. LaValle and Jr. Kuffner, J.J. Randomized kinodynamic planning. In *Proc. IEEE Int. Conf. on Rob. and Aut.*, volume 1, pages 473–479, 1999.

[31] Steven M. LaValle. *Planning Algorithms*. Cambridge University Press, 2006.

[32] Lei Le, Andrew Patterson, and Martha White. Supervised autoencoders: Improving generalization performance with unsupervised regularizers. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31, 2018.

[33] John M. Lee. *Introduction to smooth manifolds*. Springer, second edition, 2013.

[34] Raph Levien. The elastica: a mathematical history. Technical report, EECS Department, University of California, Berkeley, Aug 2008.

[35] I. Mishani. Robot description package. URL `https://gitlab.com/imishani/mish_motoman/-/tree/master/motoman_description/urdf_for_cable`.

[36] Itamar Mishani and Avishai Sintov. Real-time non-visual shape estimation and robotic dual-arm manipulation control of an elastic wire. *IEEE Robotics and Automation Letters*, 7(1):422–429, 2022.

[37] M. Moll and L. E. Kavraki. Path planning for deformable linear objects. *IEEE Trans. on Robotics*, 22(4):625–636, 2006. ISSN 1552-3098.

[38] D. Navarro-Alarcon, H. M. Yip, Z. Wang, Y. H. Liu, F. Zhong, T. Zhang, and P. Li. Automatic 3-d manipulation of soft objects by robotic arms with an adaptive deformation model. *IEEE Transactions on Robotics*, 32(2):429–441, April 2016. ISSN 1552-3098. doi: 10.1109/TRO.2016.2533639.

[39] W. Nguyen and J. K. Mills. Multi-robot control for flexible fixtureless assembly of flexible sheet metal auto body parts. In *Proc. IEEE Int. Conf. on Rob. and Aut.*, volume 3, pages 2340–2345 vol.3, Apr 1996. doi: 10.1109/ROBOT.1996.506513.

[40] O. Roussel, A. Borum, M. TaÃ¯x, and T. Bretl. Manipulation planning with contacts for an extensible elastic rod by sampling on the submanifold of static equilibrium configurations. In *Proc. IEEE Int. Conf. on Rob. and Aut.*, pages 3116–3121, 2015.

[41] Mozafar Saadat and Ping Nan. Industrial applications of automatic manipulation of flexible materials. *Ind. Rob.*, 29:5:434–442, 2002.

[42] M. Saha and P. Isto. Manipulation planning for deformable linear objects. *IEEE Transactions on Robotics*, 23(6):1141–1150, Dec 2007. ISSN 1552-3098.

[43] John Schulman, Jonathan Ho, Cameron Lee, and Pieter Abbeel. *Learning from Demonstrations Through the Use of Non-rigid Registration*, pages 339–354. 2016.

[44] A. Sintov, S. Macenski, A. Borum, and T. Bretl. Motion planning for dual-arm manipulation of elastic rods. *IEEE Robotics and Automation Letters*, 5(4):6065–6072, 2020.

[45] A. Sintov, S. Macenski, A. Borum, and T. Bretl. Motion planning for dual-arm manipulation of elastic rods. *IEEE Robotics and Automation Letters*, 5(4):6065–6072, 2020.

[46] Dong Sun, James K. Mills, and Yunhui Liu. Position control of robot manipulators manipulating a flexible payload. *The International Journal of Robotics Research*, 18 (3):319–332, 1999.

[47] Ryo Takano, Hiromi Mochiyama, and Naoyuki Takesue. Real-time shape estimation of kirchhoff elastic rod based on force/torque sensor. In *IEEE Inter. Conf. on Robotics and Automation (ICRA)*, pages 2508–2515, 2017.

[48] Masaru Takizawa, Shunsuke Kudoh, and Takashi Suehiro. Implementation of twisting skill to robot hands for manipulating linear deformable objects. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 945–950, 2016.

[49] H. G. Tanner. Mobile manipulation of flexible objects under deformation constraints. *IEEE Trans. on Rob.*, 22:179–184, 2006.

[50] Dezhong Tong, Andy Borum, and Mohammad Khalid Jawed. Automated stability testing of elastic rods with helical centerlines using a robotic system. *IEEE Robotics and Automation Letters*, 7(2):1126–1133, 2022.

[51] Hidefumi Wakamatsu, Kousaku Takahashi, and Shinichi Hirai. Static modeling of linear object deformation based on differential geometry. In *International Journal of Robotics Research*, volume 23, pages 293–311, 2004.

[52] Hidefumi Wakamatsu, Eiji Arai, and Shinichi Hirai. Knotting/unknotting manipulation of deformable linear objects. *International Journal of Robotics Research*, 25(4): 371–395, April 2006. ISSN 0278-3649.

[53] Min Yan. *Introduction to Topology*. De Gruyter, 2016. ISBN 9783110378160. doi: doi:10.1515/9783110378160.

[54] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5738–5746, 2019.

[55] Jihong Zhu, Benjamin Navarro, Philippe Fraisse, AndrÃ© Crosnier, and Andrea Cherubini. Dual-arm robotic manipulation of flexible cables. In *IEEE/RSJ Inter. Conference on Intelligent Robots and Systems*, pages 479–484, 2018.

תקציר

מניפולציה של כבלים באמצעות רובוטים מהווה זה מכבר תחום עניין המעורר אתגרים ושאלות רבות. אחד האתגרים הקיימים בתחום הוא ביצוע מניפולציה של כבלים בסביבה סבוכה ; בסביבה שכזו קיים חוסר ודאות הנובע מהסתרות או תאורה בעייתית, דבר אשר מקשה על מצלמה להבחין בכבל ולמעשה הופך את השימוש בראייה ממוחשבת ובסגמנטציה של תמונה לפחות יעילים. כמו כן, ביצוע מניפולציה של כבל מצריך שערוך של צורת הכבל באופן מדויק בזמן אמת לצד יכולת תכנון ובקרה, אך מלבד השימוש בראייה ממוחשבת, יתר הכלים מתקשים לתת מענה מספק.

בשנים האחרונות התפתחה תיאוריה לפיה צורת כבל אלסטי ניתנת להגדרה על ידי ייצוג די פשוט – כוחות ומומנטים רק בקצה אחד של הכבל. כדי לבחון את התיאוריה באופן אמפירי, יצרנו תחילה מערכת שמטרתה לבצע מניפולציות של כבלים אלסטיים באמצעות רובוט דו-זרועי. מערכת זו מבוססת על חישה של כוחות ומומנטים בזרוע אחת של הרובוט, כך שאינה דורשת שימוש בכלים מבוססי תמונה. לאחר מכן הרחבנו את יכולות המערכת, כך שמלבד היכולת שלה, על בסיס ידע מקדים של תכונות הכבל, לשערך בזמן אמת את צורתו, היא יכולה כעת, באמצעות חיישן כוחות ומומנטים, לאמוד את תכונותיו ללא צורך בידע מקדים, לשערך את צורתו, ועל ידי כך לתכנן מסלול ולבצע בקרה על המסלול בחוג סגור.

מודל זה עליו מבוססת המערכת נשען על הנחות מסוימות שאינן בהכרח מתקיימות עבור כל סוגי הכבלים, ולכן למערכת אין את היכולת לבצע מניפולציה של כל כבל. יתרה מזאת, המודל אינו לוקח בחשבון גרביטציה ואי-לינאריות של חיישן הכוחות והמומנטים, וכדי לפתור את הבעיה הזו יש צורך לפתור את הבעיה ההופכית (שערוך של הכוחות והמומנטים מתוך צילום הכבל), אשר פתרונה יקר מאוד חישובית. בעקבות אלו, חקרנו את הלמידה של מודל המתאר את צורת הכבל על ידי מיפוי הכוחות והמומנטים הנמדדים בחיישן. על בסיס כך, אנו מציעים לאמן מודל אשר יוכל מחד לתאר את צורת הכבל על ידי מיפוי ממרחב הכוחות והמומנטים בחיישן, ומאידך כפתרון לבעיה ההופכית. לבסוף, ועל מנת לאפשר לרובוט לבצע את המניפולציות על הכבלים, ביצענו אימון של מודל נפרד, על אותו סט אימון, אשר ממפה את הכוחות והמומנטים למצב הרובוט.

אוניברסיטת תל אביב

הפקולטה להנדסה ע״ש איבי ואלדר פליישמן

בית הספר לתארים מתקדמים ע״ש זנדמן סליינר

מניפולציה של כבלים באמצעות רובוט דו-זרועי על-ידי חישה של כוחות ומומנטים

חיבור זה הוגש כעבודת מחקר לקראת התואר ״מוסמך אוניברסיטה״ בהנדסה מכנית
על ידי

איתמר משעני

העבודה נעשתה בבית הספר להנדסה מכנית
בהנחיית ד״ר אבישי סינטוב

סיוון תשפ״ב