Closed-loop System for Concentric-Tube Continuum Robots

Izzat Nadzmi Kamarudzaman

Master of Science

Artificial Intelligence

School of Informatics

University of Edinburgh

2026

Abstract

The capabilities of concentric tube robots (CTR) type of build for continuum robots to navigate through confined spaces, and conform to curvilinear paths make the CTR advantageous for minimally invasive surgeries applications. Surgical applications require very high operational accuracy, and optimal real-time performance of the robot. The CTR shape is controlled via rotation and translation of several concentrically arranged, super-elastic pre-curved tubes, that form the robot's backbone. The ever-changing rotation, translation, and shape of the CTR tubes however, cause dissimilar curvature, and stiffness, among others, to different sections of the robot. This results in discontinuities along the length of the robot.

Therefore, a stable real-time position control for this highly non-linear robots like the CTR proves to be a very challenging task. In this paper, we propose a closed-loop model-based distal-end tip position control system design that can ensure that all the boundary conditions of the model are complied to. This will improve the current standard position controller accuracy for the CTR system.

This is achieved by cascading another controller dedicated to only regulating the distal-end boundary condition which is not complied to by current standard controllers. The results from the simulations and experiments show significant difference in position estimation from both the proposed controller and the standard controller. The value of the distal-end z-axis curvature is supposed to stay at the same value as it was initially, in its reference configuration. This value however can vary significantly if not regulated, depending on the rotation, translation, and the shape of the CTR tubes.

Compliance to the model requirements are achieved with the proposed controller design integrated. Therefore, this proposed approach for designing a closed-loop controller for the CTR system could be a viable option in the future. This approach also opens up more opportunity for improvements of the Jacobian-based closed-loop system.

Acknowledgements

I would like to thank Dr. Mohsen Khadem, for allowing me the opportunity to work on the project, and for the constructive guidance, as well as insights throughout the whole project.

Table of Contents

Chapter 1

Introduction

Concentric tube continuum robots is a fairly recent approach for constructing robots. A concentric tube robot is composed of multiple concentric, pre-curved, and elastic tubes inserted into one another. This type of continuum robot, that is using pre-curved tubes construction, allows for the ability to manipulate its tip's position, length, as well as its shape and stiffness, by controlling the translation and rotation of each tubes. A robot construction with this capability, opens up more possibilities for numerous different applications, such as a minimally invasive procedures, which would require a device to be flexible, and also small enough to navigate through tight areas. Furthermore, the device must be able to operate tools and instrumentation while at the same time, accurately maintain its position and, ideally, orientation. Figure 1.1 shows a visualised conception of the concentric tube robot (CTR).

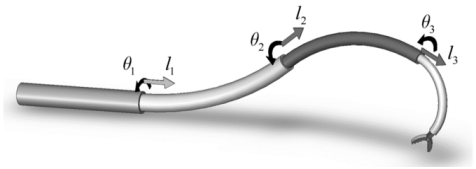

Figure 1.1: Figure from [5] of a concentric tube robot comprising of 4 tubes telescoping out of each other, and can be rotated or translated with respect to each other.

Since the advent of the CTR just less than a decade ago, there has been an increase in popularity on the research and development effort of the CTR in terms of its design, modelling, and control [7]. This trend continue to gain further momentum, given the remarkable potential the CTR construction of continuum robots have, especially in medical applications, namely for minimally invasive surgeries (MIS).

With the promise of manoeuvrability and dexterity, Burgner-Kahrs *et al.* in the

paper [3], show the various different surgical disciplines and procedures that can benefit from CTR continuum-typed robotics construction for surgical devices. Operational accuracy and performance of the CTR are crucial for such applications. However, a stable real-time position control is a challenging task due to the robot's modelling uncertainties, complexity, and mechanical instabilities.

Previous studies on position control have largely focused on either model-free control system, or mechanics-based open-loop control. While some able to achieve considerable results, these types of control models get constrained as the CTR dynamics get more complex, even more so when the controller has to be real-time [7]. There are however some prior works done to achieve better real-time accuracy by designing a closed-loop model-based control system, but the design is rather limiting to the capability of the CTR as a tool. Besides, solutions that are able to generalise better to more complex control requirements are needed instead, if CTR for MIS applications were to be realised. A more in-depth review will be further discussed in the Background section of this paper.

This paper intends to propose a method to improve, and implement a distal-end tip position control of the CTR robot system, by means of a closed-loop model-based approach. Furthermore, with the recent advancements in term of online sensing capabilities such as a stereo cameras [29], x-ray [12], magnetic resonance imaging (MRI) [25], and ultrasound [15] among others, the notion of a real-time and close-loop controller is well fitted. Moreover, the current dynamic models of CTR [4], [19], incorporating sufficient external and internal effects, enabling various applications [7] to be possible, should be utilised.

1.1 Objectives and Contributions

The main objective for the project is to design, and improve a closed-loop and model-based type of control system, for the distal-end tip position of the CTR. The hypothesis here is that the implementation of this type of controller for the CTR will be more accurate, and robust towards internal and external noise, when compared to the open-loop mechanics-based controller, or model-free controller design.

Following the hypothesis, a closed-loop model-based control system seems to be the best bet for developing a CTR that demands foolproof accuracy, especially for critical applications such as the MIS. The intention is that, improving the performance of this type of controller would consequently, be contributing more towards the develop-

ment of CTR as a whole.

The main contribution for this thesis in particular, is proposing a new approach for solving the boundary condition of the CTR system, hence improving the overall controller design and performance compared to the current standard implementation. This allows better implementation of the controller when integrating with actual physical CTR system. Aside from that, an added contribution would be the development of the robot model, controller, as well as a simulation and testing environment in Python 3.7.

This project narrows down its scope to only position control of the tip at the distal end of the CTR tube, and without external load on the CTR to isolate the error source to testing only the boundary value problems (BVP) solution accuracy. The final shape or curvature of the CTR is also not considered as a constraint, and therefore any shape that results in the correct desired position of the distal end tip of the CTR is still considered as one of the potential solutions for the trajectory of the CTR to the desired position.

Even so, the proposed closed-loop control system design will be implemented on the more recent kinematics models for CTR that can generalise to arbitrary number of concentric tubes with varying curvatures and stiffness. The project is fully simulation based. Forward kinematics is currently used as the closed-loop feedback to the control system. This can later be replaced with other sensing technology as previously mentioned, for different applications of the CTR.

The rest of the thesis is organised as such: Chapter 2 delves further into required background knowledge needed in order to solve the BVP of the CTR system. Chapter 2 also includes review of past literature to assist us better in designing and development of the project. Chapter 3 articulates more specifically the methodology and implemetation of the whole project. Chapter 4 discusses the experiments done and evaluate the results. Chapter 5 and 6 then suggest for future work or possible improvements, and conclude the thesis, respectively.

Chapter 2

Background and Literature Review

This chapter discusses the relevant prior background needed and a brief review of present literature related to the subject matter. The chapter is split into components required for this thesis, starting from the model formulation and the model solution approaches, as well as introducing the nomenclatures that will be used throughout. We then briefly review other prior studies on model-less approach, and controller design.

2.1 Model Formulation

There have been significant prior studies regarding the modelling of the CTR dynamics, developed by a few different groups, with incremental progress in development approach, and incorporating different mechanical effects to the model. Simplest model considers only the geometrical transformation, with an assumption that an outer tube having infinite stiffness compared to the ones within it, where an inner tube cannula complies to the shape of its outer tube [6], [26]. The beam bending mechanics are then introduced in [23], and [28] which also included torsional effect in the straight section of the tubes. Torsion for the curved part of the tubes are later considered in [21], and [5]. Finally, [19], and [13] incorporate external loading effects, while also including bending and torsion for an arbitrary number of tubes, for the entire length of the tubes. These models make it possible to develop model-based control strategies. The dynamics model formulation efforts have since become largely stagnant, possibly because these models have reached the “complexity vs. accuracy” balance.

2.1.1 Cosserat Rod Theory

Researches on the model formulation for the concentric tube type of continuum robots have appeared to converge to leveraging the *special Cosserat rods theory* [7]. The theory is special case of the more general Cosserat continua [1], and made way for a more useful, and accurate physical model for simulating thin deformable objects [16]. This physical model allows for the discretisation of thin solids to be solved more efficiently, for many applications.

It is necessary for us here to obtain some relevant background on the notations and formulation concept concerning the Cosserat model, which will be used throughout the thesis. Books by Antman [1] and Rubin [18] provide a more exhaustive formulations and explanations of the Cosserat rod theory.

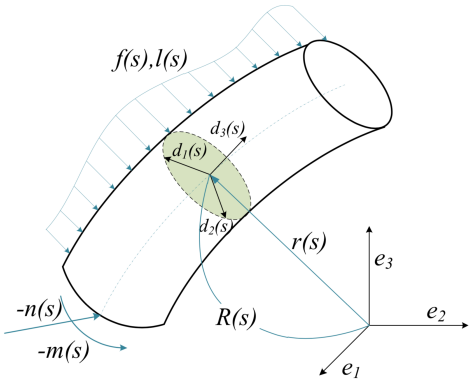

Figure 2.1: Figure from [31] of a Cosserat rod representation.

Figure 2.1 illustrates an arbitrary section of a flexible rod, in 3-D space being subjected to external force $\boldsymbol{f}(s)$ and moment $\boldsymbol{l}(s)$ with respect to its arc length s. Each cross-section of the rod along the arc length, from the proximal end ($s = 0$) to the distal end ($s = l$), also experiences internal force $\boldsymbol{n}(s)$ and moment $\boldsymbol{m}(s)$ exerted by the materials before and after a particular arc length. The position and orientation of a particular cross-section at arc length s is represented by the vector $\boldsymbol{r}(s)$ and rotation matrix $R(s)$ respectively, between with two frames. The first frame, the static world-frame $\{e_1, e_2, e_3\}$, is associated to the cross-section's original position and orientation, or undeformed shape of the rod. Secondly, the body frame $\{d_1(s), d_2(s), d_3(s)\}$, moves together with the particular cross-section of an arc length with the z axis $(d_3(s))$ perpendicular to it, or always tangential to the rod's centre line. For this thesis, we are following the *Bishop frame* convention of framing representation [2], with a nonrestrictive assumption that the rods (or tubes as in our case) do not possess an initial

material torsion. Next frame $F(s)$ is obtained by sliding along the curve from $F(0)$ without rotation about the tangent z-axis.

The classical forms of the equations of equilibrium as given in [1], states the following:

$$\boldsymbol{n}'(s) + \boldsymbol{u}(s) \times \boldsymbol{n}(s) + \boldsymbol{f}(s) = 0 \tag{2.1}$$

$$\boldsymbol{m}'(s) + \boldsymbol{u}(s) \times \boldsymbol{m}(s) + \boldsymbol{v}(s) \times \boldsymbol{n}(s) + \boldsymbol{l}(s) = 0 \tag{2.2}$$

$\boldsymbol{u}(s)$ is the curvature vector $\begin{bmatrix} u_x(s) & u_y(s) & u_z(s) \end{bmatrix}^T$ where the u_x and u_y elements are the bending curvatures, and u_z is the torsional curvature, about the local body frame axes. The vector $\boldsymbol{v}(s) = \begin{bmatrix} v_x(s) & v_y(s) & v_z(s) \end{bmatrix}^T$ contains the physical variables shear strain at the x and y axes, and elongation at the z-axis of a body frame. The $'$ denotes a derivative with respect to the arc length s.

As aptly summarised by [31], the Cosserat rod theory interprets the curvatures as angular rates of change between body frames as we slide along the centre line. Analogous to angular velocity in rigid-body motion, this yields the equation

$$\boldsymbol{u}(s) = \left(R^{\mathrm{T}}(s)R'(s)\right)^{\vee} \tag{2.3}$$

This thesis follows the notation convention used by Rucker *et al.*. The $^{\vee}$ operator is used to denote either the conversion of elements of $\mathfrak{so}(3)$ to its corresponding element in $\mathbb{R}^3$, or $\mathfrak{se}(3)$ to $\mathbb{R}^6$. The inverse operation in both cases is denoted by the $\widehat{\ }$ operator.

Similar analogy can be made for the shear strain and elongation, that is as displacements in rigid-body motion. This gives us the expression

$$\boldsymbol{v}(s) = R^T(s)\boldsymbol{r}'(s) \tag{2.4}$$

The special case of the Cosserat rod theory adapts the classical elastic-rod theory of Kirchoff standard assumptions [1]. These assumptions are that the rods are not extensible, has no shear of the cross-section, and has linear constitutive behaviour for bending and torsion. For long thin rods, inextensibility and shearlessness are generally considered to be an acceptable assumptions [19]. Therefore, the vector $\boldsymbol{v}(s) = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$. The linear constitutive assumption is used to describe the relationship between the strains and the internal moment vector, in world-frame coordinates, as follows,

$$\boldsymbol{m}(s) = R(s)K(s)\left(\boldsymbol{u}(s) - \boldsymbol{u}^*(s)\right) \tag{2.5}$$

where the * denotes variables associated to the original undeformed shape, or precur-

vature; and

$$K(s) = \begin{bmatrix} E(s)I(s) & 0 & 0 \\ 0 & E(s)I(s) & 0 \\ 0 & 0 & G(s)J(s) \end{bmatrix} \tag{2.6}$$

is the stiffness matrix. $E(s)$ is the tube's Young's modulus, $I(s)$ is the second moment of inertia, $G(s)$ is the shear modulus, and $J(s)$ is the polar moment of inertia.

These concepts are built upon to formulate the more recent concentric tube kinematics models in the literature, which will be utilised for our project. The next subsection gives some brief background on the rest of the formulation process for a full CTR kinematics model.

2.1.2 CTR Model Kinematics

Rucker *et al.*[19] built on these concepts to develop a general model for any number of tubes with different precurvatures. Their model also considers the effects of external point and distributed loading. Here we will utilise the model formulated by Rucker *et al.* as theirs is one of the more complete, and recent model. We now briefly summarise the formulation of the CTR kinematics model to get a better understanding with regard to the right approach to solving our problem.

Following the special Cosserat rod theory in Section 2.1.1, we obtain the first two kinematic formulations of a single precurved tube. From the equations (2.3) and (2.4) we can derive:

$$\dot{\boldsymbol{r}}(s) = R(s)\boldsymbol{e}_3 \tag{2.7}$$

$$\dot{R}(s) = R(s)\widehat{\boldsymbol{u}}(s) \tag{2.8}$$

where $\boldsymbol{e}_3 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$.

Generalising to multiple tubes, with the added assumption that the clearance between overlapping tubes are just enough to only allow relative motion of between the tubes, the final deformed tubes must conform to the same final curvature. Here, we will consider the final curvature (for x and y bending components) for each tube to conform to the innermost tube at a given equilibrium time t,

$$r_i(s) = r_1(s,t)$$

where $i = 1, \cdots, n$ is the tube number starting from the innermost tube. Expanding equation (2.7) that $\dot{\boldsymbol{r}}_i(s) = R_i(s)\boldsymbol{e}_3$, and given the previous consideration, we obtain

$$R_i(s) = R_1(s)R_{\theta_i} \tag{2.9}$$

where it shows that the $R_i(s)$ for each tube differ by a rotation R_{θ_i} by some twist angle about the z-axis, which we parameterise here as $\theta_i(s)$. It also implies that the torsional z component can vary independently at different rates for each tube, giving us

$$\dot{\theta}_i(s) = u_{i,z}(s) - u_{1,z}(s) \tag{2.10}$$

Note that the (s) notation is dropped from this point onwards for clarity and simplicity.

We can derive further and solve for $\dot{\boldsymbol{u}}_1$, from the equilibrium equations (2.1) and (2.2), the constitutive law (2.5), and the kinematic relationships (2.7) and (2.8), which give us

$$\begin{aligned}\begin{bmatrix} \dot{u}_{1,x} \\ \dot{u}_{1,y} \end{bmatrix} = &- \left(\sum_{i=1}^{n} K^i\right)^{-1} \sum_{i=1}^{n} R_{\theta_i} \left(K_i \left(\dot{\theta}_i \frac{dR_{\theta_i}^T}{d\theta_i} \boldsymbol{u}_1 - \dot{\boldsymbol{u}}_i^* \right) \right. \\ &\left. + \left(\widehat{\boldsymbol{u}}_i K_i + \dot{K}_i\right)\left(\boldsymbol{u}_i - \boldsymbol{u}_i^*\right)\right)\Big|_{x,y} \\ &- K^{-1} \left(\widehat{e}_3 R_1^T \int_s^{\ell} \boldsymbol{f}(\sigma) d\sigma + R_1^T \boldsymbol{l} \right) \Bigg|_{x,y}\end{aligned} \tag{2.11}$$

$$\begin{aligned}\dot{u}_{i,z} = &\dot{u}_{i,z}^* + \frac{E_i I_i}{G_i J_i}\left(u_{i,x} u_{i,y}^* - u_{i,y} u_{i,x}^*\right) \\ &+ \frac{\left(\dot{G_i J_i}\right)}{G_i J_i}\left(u_{i,z}^* - u_{i,z}\right) - \frac{1}{G_i J_i} e_3^T R_i^T \boldsymbol{l}_i\end{aligned} \tag{2.12}$$

We have now obtained the fundamental first-order state equations for the torsionally compliant, multitube model. Their derivative equations are defined in (2.7), (2.8), (2.10), (2.11), and (2.12).

2.2 Boundary Conditions

We can now use the fundamental equations formulated to solve the CTR kinematics. Additionally however, the fact that we have multiple tubes with different curvatures, lengths, and stiffness, causes discontinuous boundaries, that present as a challenge in the implementation the model. These discontinuities can be accounted for by enforcing boundary conditions at these transition points, thus allowing us to solve each section as a bounded series of continuous system.

The boundary conditions that must be enforced at these points with discontinuous solution are as follows:

- The position and orientation of each tube to be continuous across a boundary.

- Static equilibrium of energy across a boundary, where the internal moment just before is equal to just after the boundary together with an external applied point moment.
- The static equilibrium at the distal end of the tube is that the total internal moment will just be an external applied point load.
- The curvatures at the proximal end at arclength $s = 0$, starting at the point of fixed entry for each tube, will be the initial precurvature values.

Refer to the section 'Model Implementation' by Rucker *et al.* [19] for a more detailed description with regard to these boundary conditions.

2.3 Model Solution

The fundamental derivative equations obtained from the model formulation can be solved by numerically integrating them simultaneously, as a boundary value problem. There is no agreed way that is considered as the best approach to solving the kinematics as of yet [7].

With the multiple boundary conditions enforced, as well as the various external mechanical influence to account for, analytical solutions for the kinematics proved to be very complex to derive. The analytical solutions that are available so far are using a torsionally rigid model as first derived in [23], or torsionally compliant but limited to two tubes with constant curvature [5], [21].

Therefore, most models to date, solve the equations numerically by implementing the shooting method. This is done by solving a series of initial value problems (IVP) by iteratively adjusting unknown state variables at one end of the robot and working through to the other end until all the boundary conditions are satisfied. Both cases have been done, either from the base to the tip of the robot [22], as well as from the distal end to the proximal end, with the help of stereo camera measurement system to measure the tip position and a circular graduated disk to measure the tip rotation [13].

Furthermore, other approaches are continuously produced to find improved solutions for solving the CTR kinematic model. Peyron *et al.* [17] proposed a new numerical approach of using dynamic relaxation to solve the CTR kinematics. This initial study suggested a discretisation method of the CTR dynamics, and then using the dynamic relaxation method to model the boundary conditions of the CTR system at any

given location. Another study simplifies the CTR model using piecewise linearisation to obtain a closed-form kinematic model for each linear model that is dicretised in terms of arc-length s segmentations [31]. This requires many segmentation to be more accurate. A torque/force sensor is also used at the base of the CTR tube to physically determine the proximal boundary condition. However, the sensor dependence may affect the flexibility of the robot design.

2.4 Model-less Approach

Model-based kinematics for CTR does not come without any disadvantages. To make a model more accurate by factoring more mechanical effects and system dynamics into it, would usually make a model more complex. This slows down computation and adversely affect the performance of a controller. Moreover, physical parameters of an actual CTR such as stiffness and shear modulus, have to be calibrated carefully in order to for a model to work properly.

A few prior studies have been done on model-less approach for continuum robots, although not specifically implemented on CTR type, proposed that a model-free approach could overcome some of the issues of the unknown, or noisy parameters that a model-based approach face, without the extra computational overhead. One study proposed a data-driven, reinforcement learning inspired approach for model-free control, by using a four steps process, namely discretisation, visualisation, learning, and optimisation [27]. The state-to-state control by learning each transition rewards, which was implemented on a three-limbed soft robot, however, will not be able to be applied to a more complex continuum robot configuration such as the CTR which have a massive amount of states.

Another model-free approach is to learn the Jacobian matrix in real-time by moving each actuator of a continuum robot independently, and then normalise the Jacobian estimation by extended data selection [33]. This particular study, which uses the position and force information of the tip of the robot, were able to navigate through unknown constrained environment. However, it has a few limitations including it's reliance on the sensor accuracy, placement of the sensor at the distal end of the robot, and difficulty in regulating sudden change in force at the tip. One other similar study which estimates the Jacobian matrix of the robot, normalises the Jacobian using Kalman filtering to remove motion artifacts [11], manages to reduce the the physical parameters uncertainty as expected from a model-free method and good robustness. However the

the controller still has slow convergence, even when implemented in a static environment. This nullifies the advantages of a model-free approach when comparing to a model-based control.

Generally, the model-free control approach is still currently very unrefined, and quite limiting to generalise to complex continuum robots and applications.

2.5 Controller Design

Generally, the position control system of CTR consists of two main steps which are the forward kinematic, and inverse kinematic to map the joint-space and work-space of the robot. CTR is a complex hybrid system where the the joint parameters at the proximal end of the CTR are in terms of rotations and translations of each tubes, whereas the end effector parameters are the coodinate location of the distal end tip, which have to be calculated in terms of tubes' curvature, bending, torsion, and length among others.

There have been several approaches done in designing the CTR control system, as neatly summarised by [7], in three general frameworks.

The first framework solves the forward kinematics by pre-computing the model solutions over the entire workspace of the robot and then approximate the forward kinematics by fitting the solution, for example with a multidimensional Fourier series [5]. While this approach is consistent enough for real-time implementation of the CTR, since the solutions were pre-calculated, the model constraints itself to only one solution of the forward kinematics. This model is also incapable of incorporating torsional effects of external loading.

The second framework, generally, utilises the capability for solving the kinematics using the Jacobian linaerisation method. The second of using approach has many advantages such as the potential to make computation time faster, capability to control the robots taking into account the multiple possible solutions, and the ability to control the CTR with known external loads. [20] for example, extended their model to include a faster Jacobian-based inverse kinematics algorithms. The extension provides differential kinematics maps for actuation and external loading, in terms of Jacobian matrix and compliance matrix respectively, enabling a resolve-rate type of control for it to be suitable for real-time implementation. There are disadvantages nonetheless, such as the increased in development complexity, and lack of convergence guarantees for the root-finding procedures as compared to the first method.

The third framework involves the usage of additional sensors, such as the torque

sensor at the proximal end of the CTR, in order to physically measure and eliminate unknown boundary conditions there [32], [31]. This approach simplifies the computations, retains the full model structure, and at the same time eliminates the concern of forward kinematics convergence. However, this framework will largely rely on the performance of the sensor, and complicates the mechanism for actuation of the tubes.

Since the subject of CTR is still generally in its infancy, there is little effort on developing these frameworks in a closed-loop model-based CTR controller. The prior works on position control of the robot have mostly been open-loop mechanics-based control [5], [32], or vision-based model-free control [30], [14]. However, there are some published materials related to the closed-loop model-based control strategies, which include the use of camera mounted on the tip of the CTR [9], and most promisingly, the use of magnetic resonance imaging (MRI) guided closed-loop CTR tested in free space, gelatin phantom, and *ex vivo* tissue [25]. The former one, while obtaining very good accuracy, may not be ideal if the CTR needs to be mounted with a different surgical device instead of a camera. For the latter, the three-tubed CTR design was very simplified to have only the middle tube to be pre-curved, eliminating the need to consider torsional effects.

Chapter 3

Methodology

This chapter delves further into the derivation and implementation of the specific algorithms, and the approach taken in this study. Frequent reference is made to the theory explained in the Background section.

Cascaded controller is used to solve the boundary value problem for the robot kinematics. In this project, to remain within specified scope, we will ignore external forces and only experiment an isolated robot environment. This on the upper hand, helps simplify our computation, and confine sources of errors to just the robot itself.

3.1 Kinematic Modelling

For this project we adopt one of the more recent model that incorporates torsional compliance, and account for external forces interaction to the CTR system, as well as allowing arbitrary lengths and curves [19]. However since the main objective for this thesis is solving the distal-end boundary value problem, we confine the scope such that the CTR system is only moving in free space without external loads. Another assumption is that the CTR consists of three tubes and each tube retains the same width and stiffness along its length. These assumptions allow us to simplify the model formulation that was derived in the Background section.

The new simplified model equations for the CTR, which will be used throughout the project are as shown below.

$$\dot{\boldsymbol{r}}_1(s) = R_1 \boldsymbol{e}_3 \tag{3.1}$$

$$\dot{R}_1 = R_1 \widehat{\boldsymbol{u}}_1 \tag{3.2}$$

$$\begin{aligned}\begin{bmatrix} \dot{u}_{i,x} \\ \dot{u}_{i,y} \end{bmatrix} = &- \left(\sum_{i=1}^{n} K^i \right)^{-1} \sum_{i=1}^{n} R_{\theta_i} \left(K_i \left(\dot{\boldsymbol{u}}_i - \dot{\boldsymbol{u}}_i^* \right) \right. \\ &\left. + \widehat{\boldsymbol{u}}_i K_i \left(\boldsymbol{u}_i - \boldsymbol{u}_i^* \right) \right)\Big|_{x,y} \\ &- K^{-1} \left(\widehat{\boldsymbol{e}}_3 R_1^T \int_0^s \boldsymbol{f}(\sigma,t) d\sigma \right) \Bigg|_{x,y}\end{aligned} \tag{3.3}$$

$$\dot{u}_{i,z} = \frac{E_i I_i}{G_i J_i} \left(u_{i,x} u_{i,y}^* - u_{i,y} u_{i,x}^* \right) \tag{3.4}$$

$$\dot{\theta}_i = u_{i,z} \tag{3.5}$$

3.2 Boundary Value Problem

Given the assumptions from the model formulation in Background section, as well as the assumptions made for our specific CTR system above, the boundary conditions that exist for the system are as follows

$$\boldsymbol{r}_1|_{s=0} = [0\ 0\ 0]^T \tag{3.6}$$

$$\mathbf{R}_1|_{s=0} = \mathbf{R}_{1,z}|_{s=\alpha_1 - \beta_1 u_{1,z}} \tag{3.7}$$

$$\theta_i|_{s=0} = \alpha_i - \beta_i u_{i,z} \tag{3.8}$$

$$u^i\big|_{s=\ell_i+\beta_i} = u_i^* \tag{3.9}$$

where the notation $\left(|_{s=\xi} \right)$ indicates the value of a variable at arclength of ξ, ℓ_i indicates the length of the i-th tube, β is the length of the tubes' translation, and α is the tubes' rotation by the CTR actuator.

These boundary conditions define that, respectively, the $\boldsymbol{r}$ vector starts at the proximal end of the tube, rotation at the proximal end is equal to only the z-axis rotation, the twist angle at the proximal end (arclength $s = 0$) is equal to the twist angle at the point where the tubes are grasped at (the tubes extent further back to the actuators), and the curvature at the distal end of the tubes is equal to its initial precurvature before deformation.

3.3 Controller Design

Figure 3.1 shows the full high-level architecture of the controller design. Additional distal end U_z controller is added that runs asynchronously to the main position controller. The U_z controller works by improving the Jacobian matrix calculation of the

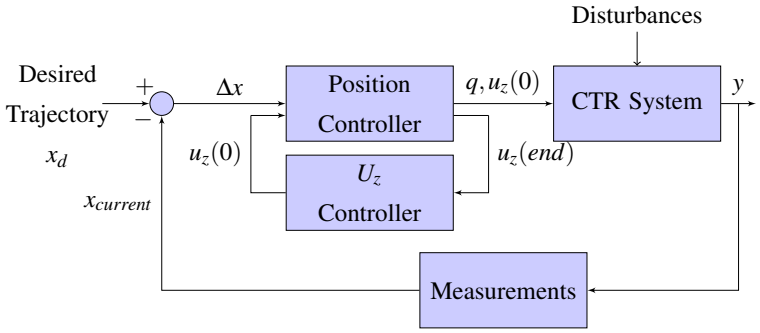

Figure 3.1: Controller block diagram attached to the outer loop.

main controller by making sure the distal-end boundary condition, as stated in (3.9), is complied to every time the CTR model is called by the main controller. The curvature at the CTR's proximal end, $u_z(0)$, is regulated, in order to ensure that the U_z at the distal end remains the same as its initial value before deformation.

Both the main controller for CTR tip positioning, and the U_z controller have the same design. The controller block in Figure 3.2 shows the block diagram of the controller design that will be used for both controllers. Figure 3.2, without the U_z controller attached, would be the standard closed-loop controller for a CTR system. A proportional controller controller is used together with Jacobian linearisation as the controller for the system.

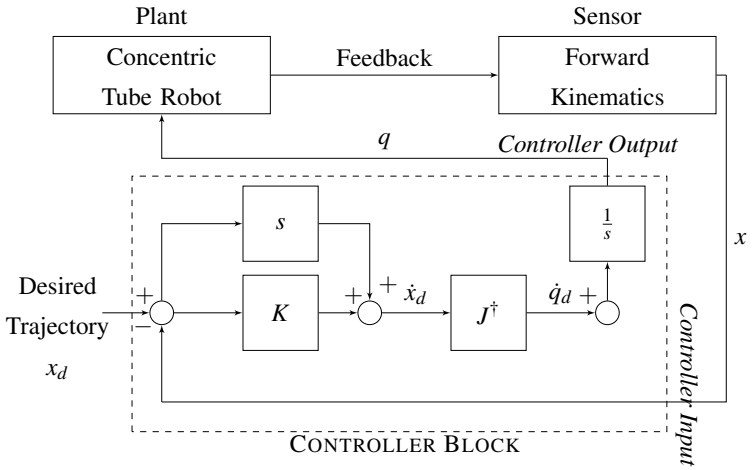

Figure 3.2: Controller block diagram of a position control closed-loop system without the distalend boundary condition U_z curvature controller.

For closing the loop in the controller, we use forward kinematics to obtain the robot end effector position as this project is purely simulation based. The forward kinematics will either use the same model as used in the Jacobian matrix calculation,

or a perturbed model to simulate a noisy system. Actual implementation will require external sensors as discussed in the Background section.

3.3.1 Controller Block

The controller block main components, as mentioned before, are the proportional controller for regulation, and the Jacobian linearisation for doing inverse kinematics. The $\boldsymbol{x}$ is the end effector position vector, here defined as the tip position of the inner-most tube, $\boldsymbol{x}(t) = \boldsymbol{r}_1|_{s=\ell_1+\beta_1}$. The output vector $\boldsymbol{q}$ will be the actuator input signal, consisting of the translations and rotations for each tubes, $\boldsymbol{q} = [\beta_1\ \beta_2\ \beta_3\ \alpha_1\ \alpha_2\ \alpha_3]^T$

The Jacobian matrix is used for mapping the robot's motor actuation joint-space for each of the tube to the robot's end effector task-space. This can be formally described as

$$\boldsymbol{x}' = \mathbf{J}\boldsymbol{q}' \tag{3.10}$$

where the $'$ denotes the velocity, or the derivative of a variable with respect to time. In order to obtain the inverse kinematics for the system, the above equation becomes

$$\boldsymbol{q}' = \mathbf{J}^{\dagger}x' \tag{3.11}$$

where $\mathbf{J}^{\dagger}$ is the pseudo-inverse of the Jacobian matrix [24] using Moore-Penrose inverse

$$\mathbf{J}^{\dagger} = \mathbf{J}^T\left(\mathbf{J}\mathbf{J}^T\right)^{-1} \tag{3.12}$$

The proportional controller regulates the error between the current position $\boldsymbol{x}$ and the desired position $\boldsymbol{x}_d$ in the trajectory. The controller is needed on top of the Jacobian linearisation, due to the non-linearities of the system. The proportional controller is integrated to the system as follows

$$\boldsymbol{q}_d' = \mathbf{J}^{\dagger}\left[\boldsymbol{x}_d' + \mathbf{K}_p\left(\boldsymbol{x}_d - \boldsymbol{x}\right)\right] \tag{3.13}$$

Equation (3.13) make up the 'Controller Block' as in the Figure 3.2, where the 's' block is the differentiator, and the '$1/s$' block is the integrator to convert the $\boldsymbol{q}_d'$ velocities to working motor actuator input for the CTR.

3.3.2 Jacobian Linearisation

The Jacobian is calculated numerically with central finite difference method as follows

$$\mathbf{J} = \frac{\Delta x}{\Delta q} = \left[\begin{array}{c} \frac{\boldsymbol{x}^T\left(\boldsymbol{q}+\frac{\Delta q_1}{2}\boldsymbol{e}_1\right)-\boldsymbol{x}^T\left(\boldsymbol{q}-\frac{\Delta q_1}{2}\boldsymbol{e}_1\right)}{\Delta q_1} \\ \frac{\boldsymbol{x}^T\left(\boldsymbol{q}+\frac{\Delta q_n}{2}\boldsymbol{e}_n\right)-\boldsymbol{x}^T\left(\boldsymbol{q}-\frac{\Delta q_n}{2}\boldsymbol{e}_n\right)}{\Delta q_n} \end{array} \right]^T \tag{3.14}$$

where the $\boldsymbol{e}_i$ is the i-th unit vector of the canonical basis of the n-dimensional joint space. This method of calculating the Jacobian has the advantage of enabling parallelisable calculation without sacrifices in the accuracy of the model's kinematics [10]. The column of the $\mathbf{J}$ matrix can be calculated in parallel.

The Jacobian matrix is calculated at every timestep. Small sampling time is used so that the linearisation approximation remains true for each action step taken by the robotic manipulator for achieving the desired trajectory.

3.4 Software Implementation

The software implementation for the project is mainly split into two parts which are the model implementation, and simulation and testing. This section will go through each part of the implementation with some details.

3.4.1 Model Implementation

The model implementation also consists of several components, namely the tube segmenting, solving the model forward kinematics with ordinary differential equation (ODE), and controller development.

The tube segmentation splits the CTR tubes into smaller sections, depending on the start and end of the straight part of each tubes, start and end of of the curved section of each tube, as well as taking into account any actuator translation. This part ensures that the boundary conditions can be enforced for each sections, caused by the discontinuity.

Next, the model or the forward kinematics solution is developed by solving the fundamental derivative equations as stated in the Kinematic Modelling section, together with the boundary conditions as stated in the Boundary Conditions section as a BVP. These equations are solved together, numerically by applying ODE. The Runge Kutta ODE method is used, which introduces negligible numerical error in comparison to the kinematic error, as expected in a numerical example [7].

Finally, the controller development is developed according to the design as detailed in the Controller Design section. Parallelisation during the Jacobian matrix calculation, as suggested in [10], was implemented to speed up the testing process later.

3.4.2 Simulation and Testing Environment

	TUBE 1	TUBE 2	TUBE 3
TUBE LENGTH (*mm*)	431	332	174
CURVED LENGTH (*mm*)	103	113	134
CURVATURE (m^{-1})	21.30	13.11	3.50
YOUNG'S MODULUS, E (*GPa*)	64.36	52.55	47.16
SHEAR MODULUS, G (*GPa*)	25.09	21.47	29.79

Table 3.1: Physical Parameters for Simulated Tubes in CTR

A simulation environment is developed and used for validation and testing. Reasonable CTR physical parameters and configuration are used for the robot simulation model in this project, based from a previous work done by Khadem *et al.* [8]. Figure A.1, Figure A.2, and Figure A.3 in the Appendix A shows each of the three tubes isolated. These parameters are usually measured and calibrated from an actual robot manually. The tubes' curvatures and lengths measurements, can be obtained in a straightforward manner. The stiffness parameters of the tubes, namely the Young's modulus and the shear modulus, are obtained by fitting the kinematics model with the shape of the robot at different configurations. The tubes are bent using a known force, where the resultant shape and position of the tubes are then captured with a stereo 3D camera that translate the tube's shape and into 3D point cloud. However, capturing accurate model parameters for the CTR is a challenge, and uncertainty in the model parameters may be present even after a careful calibration procedure.

The testing environment is then developed, which mainly consists of a trajectory following setup. The only requirements here are to provide the controller, the next desired end effector position vector in the time-step, as well as the desired end effector velocity vector. The final results are then included in the plots.

Chapter 4

Evaluation and Discussion

4.1 Evaluation and Tests

A few experiments were performed to study the differences and behaviour of the CTR controller in varying cases.

To test if the simulated environment and model is working as expected, a simplified test case is run. Three tubes with the same stiffness parameter as $Tube1$ from Table 3.1, at $300\,mm$ in length of which $200\,mm$ is curved with $10\,m^{-1}$ curvature, are set up rotated 120° from each other. The resultant shape of the three tubes combined is a straight tube, as expected, as shown in Figure 4.1.

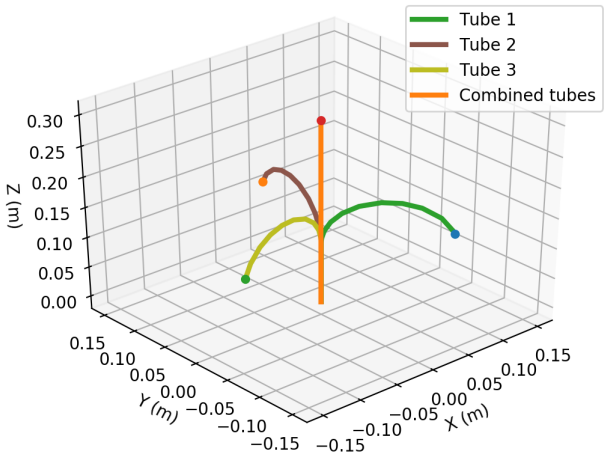

Figure 4.1: Figure shows three tubes rotated at 120° from each other and then merged to form the final combined tubes.

The workspace of the robot in terms of its distal-end tip position is also visualised to assist the experiment design. The actual robot configuration as in Table 3.1 is implemented, and then moved through a grid of different translation and rotation actuation combination. Figure 4.2 shows the workspace of the robot, which is mushroom-shaped and symmetrical. The robot's workspace is denser closer to bottom and middle. The plot is also expected given the curvature and length of the robot.

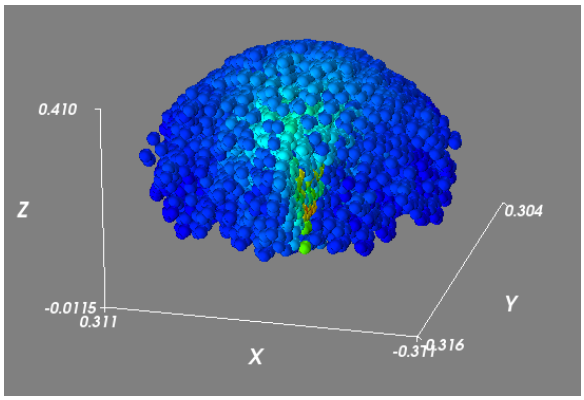

Figure 4.2: Density plot of the robot distal-end tip position workspace.

4.2 Experiments and Results

Further simulation studies are done to evaluate the effect of noncompliance on the distal-end boundary conditions. The first experiment is a straight line trajectory following for the CTR tip. Result is shown in Figure 4.3, where the actual trajectory is the CTR with the tip z-axis curvature controller integrated, while the trajectory without U_z controller is if the CTR has only the standard closed-loop controller (without distal-end boundary condition compliance). The proportional controller here is set at $K_p = 110$. The max error for the actual trajectory is $80\,\mu m$ as in Figure 4.4, whereas the standard controller is $15\,mm$ as in Figure 4.5, which is a massively significant difference.

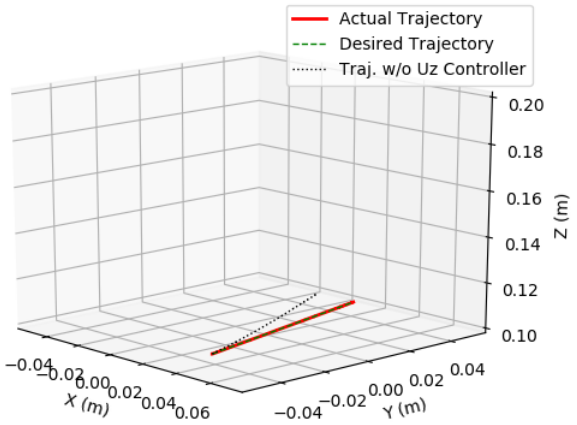

Figure 4.3: Straight trajectory following result comparison of the standard and proposed controller for CTR distal-end tip position.

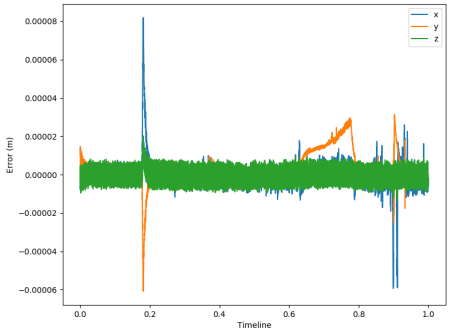

Figure 4.4: Error in x, y, and z-axis during straight trajectory following for controller with integrated U_z controller.

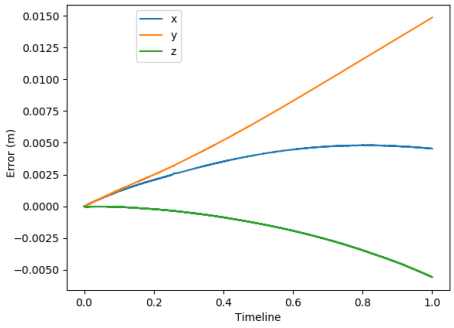

Figure 4.5: Error in x, y, and z-axis during straight trajectory following for standard controller.

The second experiment shows a similar experiment as done previously, but with a helical trajectory instead, with the result as shown in Figure 4.6. This result shows that the trajectory for the standard controller were inaccurate at the start (going from lower down to up) of the trajectory. This visualises that the distal-end boundary condition non-compliance happens only at certain positions. The distal-end boundary condition that the Z-axis curvature must be zero, is not met when the rotation of the tubes are not the same for all three of them. This is the instance where errors may occur to the control system if the Z-axis curvature fluctuation is not regulated. However, regardless

of the translation difference of the tubes, when the rotation for all the three tubes are equal, the Z-axis curvature remains at zero. This behaviour of the distal-end z-axis curvature varies depending on the rotation, translation, and curvature of the tubes.

The value of the curvature $U_z(end)$ for this helical trajectory experiment is shown in Figure 4.7. The proportional controller in this simulation is also aggressively set at $K_p = 110$ allowing the standard controller to correct its trajectory following error. However during the erroneous course of the planned path, the maximum error for the standard controller peaked up at $35\,mm$ near the start of the trajectory whereas the CTR with the curvature $U_z(end)$ controller integrated works normally. Both controller however retains some error of $15\,mm$ which occurred near the end of the trajectory as the robot moves further away from its workspace. The drift in accuracy here, shown more in Figure B.1 and Figure B.2 in Appendix B, which can be improved by making the controller more aggressive and increasing its sampling rate.

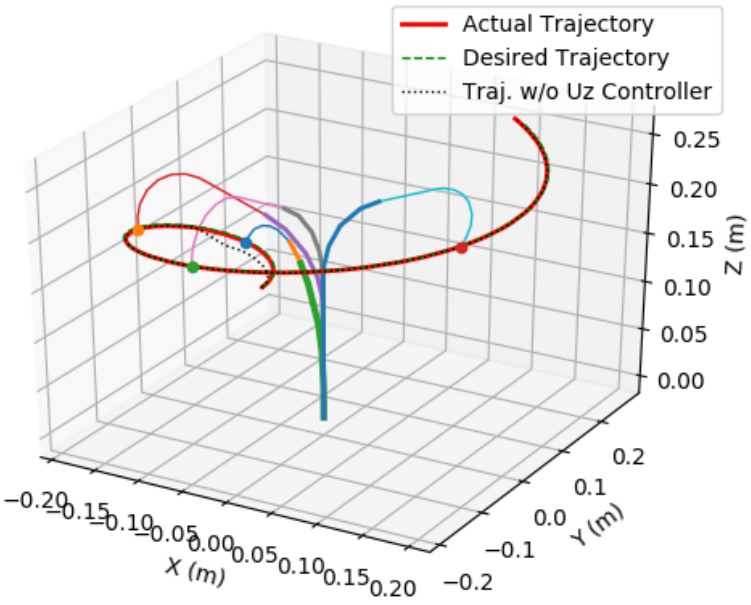

Figure 4.6: Helical trajectory following result comparison of the standard and proposed controller for CTR distal-end tip position.

The final experiment is done to illustrate other sources of error that may occur during implemetation, even if a complete model of a CTR system exist. The experiment is implemented only on the proposed integrated controller, on a straight line trajectory. The simulated testing environment is perturbed with 5% error for the robot's physical parameters, particularly the Young's modulus (E) and shear modulus (G). The experiment is firstly done at the sampling rate of $1\,kHz$, with the robot's trajectory in Eu-

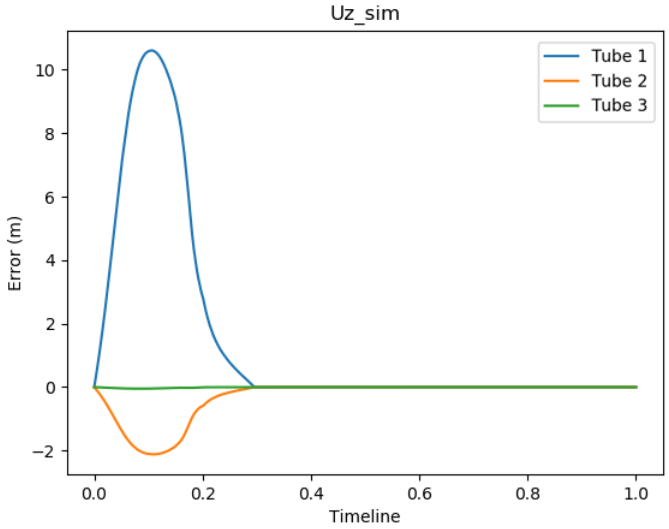

Figure 4.7: Curvature $U_z(end)$ during helical trajectory following experiment

clidean space is as shown in Figure 4.8, and then the experiment is repeated at $10\,kHz$ as shown in Figure 4.9. Both uses the same proportional controller, set at $K_p = 110$ as previous experiments.

From the results, we can see that the position control of the robot becomes very inaccurate, but improves when run at higher sampling rate. The inaccuracy however, still persist near the end of the trajectory. This could be due to singularities existing in the pertubed CTR system. On the other hand, this experiment shows the advantage of using a closed-loop control system, demonstrating that we could still optimise the performance of the controller even in noisy environment. However, this would require even more computing power to increase the sampling rate of a real-time controller. The errors for this experiment are visualised further in Appendix B, in Figure B.3 and Figure B.4.

As briefly discussed in the Methodology chapter, the CTR configuration and physical parameters are usually obtained through design and by measuring manually. This procedure is not error-proof and requires careful calibration. This type of errors may also occur with time as the flexibility of the tube material, and the mechanical system deteriorate. This consideration has to be taken into account when designing a CTR system for a surgical application. Solutions to overcome this error includes careful calibration of the CTR system, and ensuring that the robot are always operated within a safe constraint of operational limit.

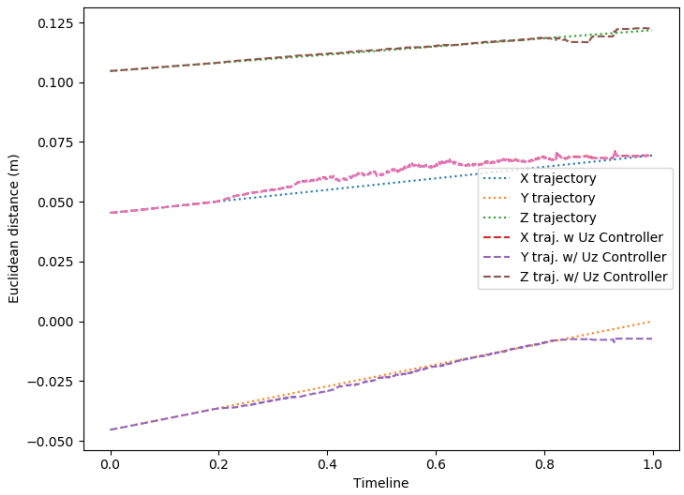

Figure 4.8: Straight trajectory following result comparison of the standard and proposed controller for CTR distal-end tip position, on pertubed CTR environment, at $1\,kHz$.

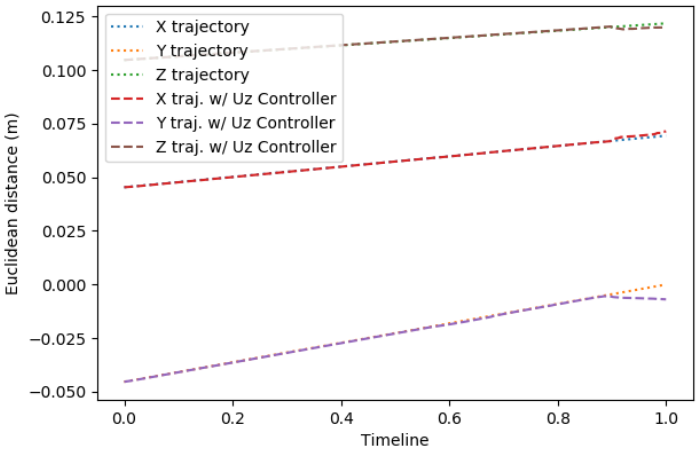

Figure 4.9: Straight trajectory following result comparison of the standard and proposed controller for CTR distal-end tip position, on pertubed CTR environment, at $10\,kHz$.

Chapter 5

Conclusions

5.1 Conclusion

In this thesis, we proposed an improved position controller for the distal-end point of a CTR. Most researches on the CTR have focused on its application for MIS. Such critical application necessitate optimum accuracy for CTR control system.

In order to achieve this, a model-based and closed-loop controller is proposed. Recent developed CTR models have incorporated substantial mechanical characteristics of the CTR, and external physical effects on the CTR into the model formulation. This however, requires the enforcement of multiple boundary conditions when solving the CTR model. Noncompliance adversely affect the controller accuracy when implemented onto an actual CTR system.

Therefore, this project starts with understanding the model formulation for better comprehension of the issue at hand. An asynchronous controller, running in series to the main position controller is developed and integrated, to regulate the distal-end z-axis curvature. This ensures that the results from the model kinematics calculation by the main controller always comply to all the BVP.

The proposed controller demonstrated a significant difference of position estimation when compared to the standard Jacobian-based controller. Given that the distal-end U_z boundary condition compliance is a requirement for the model, the proposed controller could serve as a viable approach for designing a closed-loop control system for CTR.

In summary, the objectives laid out for the thesis were successfully achieved;

- Closed-loop controller for CTR system implementation was successful.

- Boundary conditions of CTR model successfully implemented and complied to.
- A CTR simulation environment was developed, including perturbed environment to emulate reality.
- Jacobian-based controller and parallelism were also successfully implemented in the controller.

5.2 Improvement and Future Work

Analysis with an actual hardware will be required as a future effort in proving the actual measure of improvement for the solution studied here. We were unable to run experiments on an actual hardware due to unavoidable delay in the development of the mechanical system of the CTR throughout the duration of this project.

The controller performance analysis and comparison evaluated for this project were executed through software implementation built without computational efficiency as the primary concern. Results obtained were only to measure the degree of improvement that could be achieved with the suggested approach in this thesis as a potential solution. In order to achieve practicality, real-time closed-loop control system would require better computational efficiency.

As a future work, one might explore the feasibility of merging both the position control, as well as the end z-axis curvature BVP control, into one Jacobian matrix. The current proposed cascaded controller adds an extra curvature regulatory process for each inverse kinematics calculation done by the main controller, which slows down computation, even with parallelisation. This synchronous calculation could better improve the computation time of the proposed controller, as well as utilising the parallelising capability better. This approach however, may results in some error in the Jacobian matrix estimation, and would require even more tuning.

Bibliography

[1] Stuart S Antman. Nonlinear problems of elasticity. Springer, 1995.

[2] Richard L Bishop. There is more than one way to frame a curve. *The American Mathematical Monthly*, 82(3):246–251, 1975.

[3] Jessica Burgner-Kahrs, D Caleb Rucker, and Howie Choset. Continuum robots for medical applications: A survey. *IEEE Transactions on Robotics*, 31(6):1261–1280, 2015.

[4] Pierre E Dupont, Jesse Lock, and Evan Butler. Torsional kinematic model for concentric tube robots. In *2009 IEEE International Conference on Robotics and Automation*, pages 3851–3858. IEEE, 2009.

[5] Pierre E Dupont, Jesse Lock, Brandon Itkowitz, and Evan Butler. Design and control of concentric-tube robots. *IEEE transactions on robotics: a publication of the IEEE Robotics and Automation Society*, 26(2):209, 2010.

[6] Junji Furusho, Tomoya Ono, Ryousuke Murai, Tetsuo Fujimoto, Yoshihide Chiba, and Hiroyuki Horio. Development of a curved multi-tube (cmt) catheter for percutaneous umbilical blood sampling and control methods of cmt catheters for solid organs. In *IEEE International Conference Mechatronics and Automation, 2005*, volume 1, pages 410–415. IEEE, 2005.

[7] Hunter B Gilbert, D Caleb Rucker, and Robert J Webster III. Concentric tube robots: The state of the art and future directions. In *Robotics Research*, pages 253–269. Springer, 2016.

[8] Mohsen Khadem, Lyndon Da Cruz, and Christos Bergeles. Force/velocity manipulability analysis for 3d continuum robots. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4920–4926. IEEE, 2018.

[9] Andrey V Kudryavtsev, Mohamed Taha Chikhaoui, Aleksandr Liadov, Patrick Rougeot, Fabien Spindler, Kanty Rabenorosoa, Jessica Burgner-Kahrs, Brahim Tamadazte, and Nicolas Andreff. Eye-in-hand visual servoing of concentric tube robots. *IEEE Robotics and Automation Letters*, 3(3):2315–2321, 2018.

[10] Konrad Leibrandt, Christos Bergeles, and Guang-Zhong Yang. On-line collision-free inverse kinematics with frictional active constraints for effective control of unstable concentric tube robots. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3797–3804. IEEE, 2015.

[11] Minhan Li, Rongjie Kang, David T Branson, and Jian S Dai. Model-free control for continuum robots based on an adaptive kalman filter. *IEEE/ASME Transactions on Mechatronics*, 23(1):286–297, 2017.

[12] Edgar J Lobaton, Jinghua Fu, Luis G Torres, and Ron Alterovitz. Continuous shape estimation of continuum robots using x-ray images. In *2013 IEEE International Conference on Robotics and Automation*, pages 725–732. IEEE, 2013.

[13] Jesse Lock, Genevieve Laing, Mohsen Mahvash, and Pierre E Dupont. Quasistatic modeling of concentric tube robots with external loads. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2325–2332. IEEE, 2010.

[14] Yi Lu, Changchun Zhang, Shuang Song, and Max Q-H Meng. Precise motion control of concentric-tube robot based on visual servoing. In *2017 IEEE International Conference on Information and Automation (ICIA)*, pages 299–304. IEEE, 2017.

[15] Caroline Nadeau, Hongliang Ren, Alexandre Krupa, and Pierre Dupont. Intensity-based visual servoing for instrument and tissue tracking in 3d ultrasound volumes. *IEEE Transactions on Automation Science and Engineering*, 12(1):367–371, 2014.

[16] Dinesh K Pai. Strands: Interactive simulation of thin solids using cosserat models. In *Computer Graphics Forum*, volume 21, pages 347–352. Wiley Online Library, 2002.

[17] Quentin Peyron, Kanty Rabenorosoa, Nicolas Andreff, and Pierre Renaud. Evaluation of dynamic relaxation to solve kinematics of concentric tube robots. In

International Symposium on Advances in Robot Kinematics, pages 100–107. Springer, 2018.

[18] Mordecai B Rubin. *Cosserat theories: shells, rods and points*, volume 79. Springer Science & Business Media, 2000.

[19] D Caleb Rucker, Bryan A Jones, and Robert J Webster III. A geometrically exact model for externally loaded concentric-tube continuum robots. *IEEE transactions on robotics: a publication of the IEEE Robotics and Automation Society*, 26(5):769, 2010.

[20] D Caleb Rucker and Robert J Webster. Computing jacobians and compliance matrices for externally loaded continuum robots. In *2011 IEEE International Conference on Robotics and Automation*, pages 945–950. IEEE, 2011.

[21] D Caleb Rucker, Robert J Webster III, Gregory S Chirikjian, and Noah J Cowan. Equilibrium conformations of concentric-tube continuum robots. *The International journal of robotics research*, 29(10):1263–1280, 2010.

[22] Daniel Caleb Rucker and Robert J Webster III. Parsimonious evaluation of concentric-tube continuum robot equilibrium conformation. *IEEE Transactions on Biomedical Engineering*, 56(9):2308–2311, 2009.

[23] Patrick Sears and Pierre Dupont. A steerable needle technology using curved concentric tubes. In *2006 IEEE/RSJ international conference on intelligent robots and systems*, pages 2850–2856. IEEE, 2006.

[24] Bruno Siciliano. Kinematic control of redundant robot manipulators: A tutorial. *Journal of intelligent and robotic systems*, 3(3):201–212, 1990.

[25] Hao Su, Gang Li, D Caleb Rucker, Robert J Webster III, and Gregory S Fischer. A concentric tube continuum robot with piezoelectric actuation for mri-guided closed-loop targeting. *Annals of biomedical engineering*, 44(10):2863–2873, 2016.

[26] Motokazu Terayama, Junji Furusho, and Morito Monden. Curved multi-tube device for path-error correction in a needle-insertion system. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 3(2):125–134, 2007.

[27] Vishesh Vikas, Piyush Grover, and Barry Trimmer. Model-free control framework for multi-limb soft robots. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1111–1116. IEEE, 2015.

[28] Robert J Webster, Allison M Okamura, and Nah J Cowan. Toward active cannulas: Miniature snake-like surgical robots. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2857–2863. IEEE, 2006.

[29] Robet J Webster, John P Swensen, Joseph M Romano, and Noah J Cowan. Closed-form differential kinematics for concentric-tube continuum robots with application to visual servoing. In *Experimental Robotics*, pages 485–494. Springer, 2009.

[30] Keyu Wu, Liao Wu, and Hongliang Ren. An image based targeting method to guide a tentacle-like curvilinear concentric tube robot. In *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*, pages 386–391. IEEE, 2014.

[31] R Xu and RV Patel. A fast torsionally compliant kinematic model of concentric-tube robots. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 904–907. IEEE, 2012.

[32] Ran Xu, Ali Asadian, Anish S Naidu, and Rajni V Patel. Position control of concentric-tube continuum robots using a modified jacobian-based approach. In *2013 IEEE International Conference on Robotics and Automation*, pages 5813–5818. IEEE, 2013.

[33] Michael C Yip and David B Camarillo. Model-less hybrid position/force control: a minimalist approach for continuum manipulators in unknown, constrained environments. *IEEE Robotics and Automation Letters*, 1(2):844–851, 2016.

Appendix A

Simulation Environment

A.1 CTR Tubes

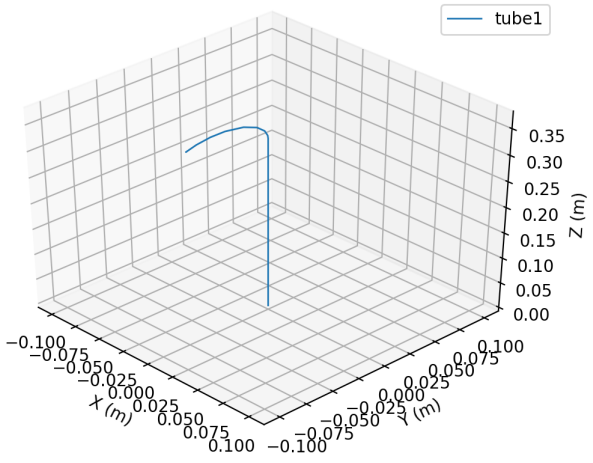

Figure A.1: Tube 1 (innermost tube) used in the Simulated Environment

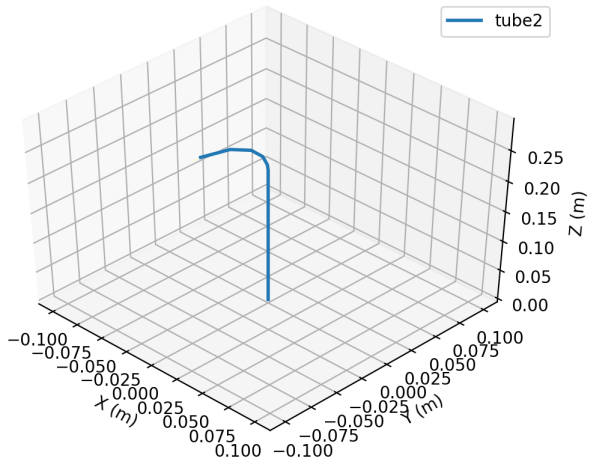

Figure A.2: Tube 2 used in the Simulated Environment

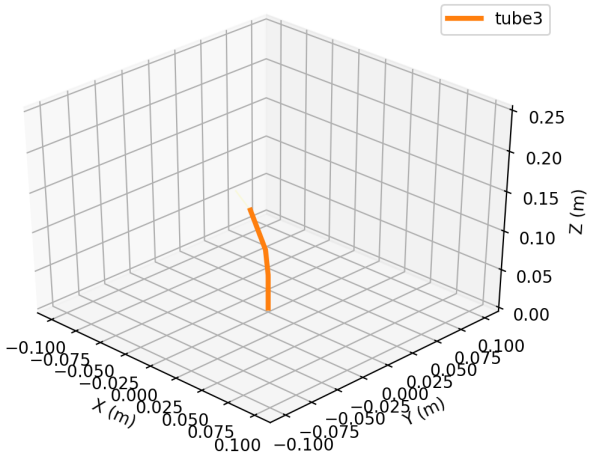

Figure A.3: Tube 3 used in the Simulated Environment

Appendix B

Extra Result Plots

B.1 Helical Trajectory Following Experiment

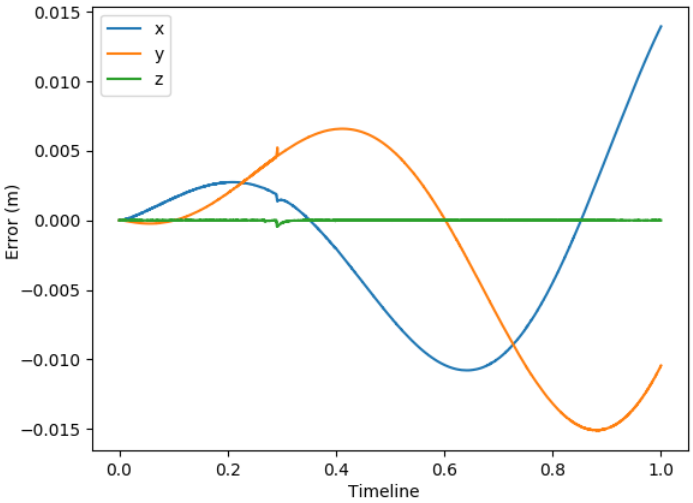

Figure B.1: Error in Euclidean distance of the CTR with the U_z boundary condition controller integrated.

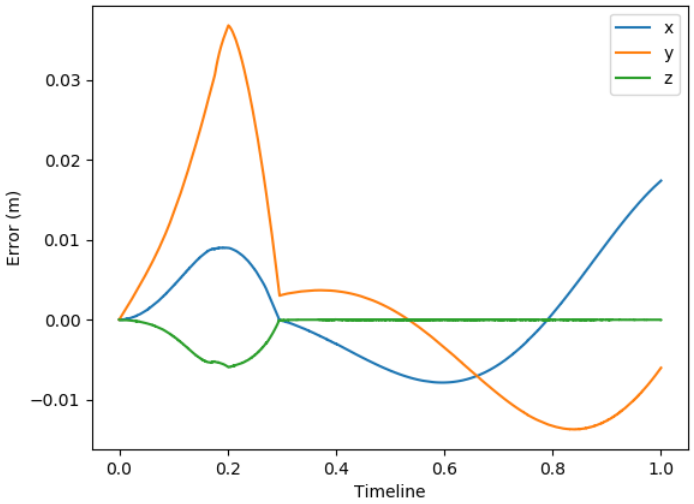

Figure B.2: Error in Euclidean distance of the CTR with just the standard controller.

B.2 Perturbed CTR System Experiment with Straight Trajectory Following

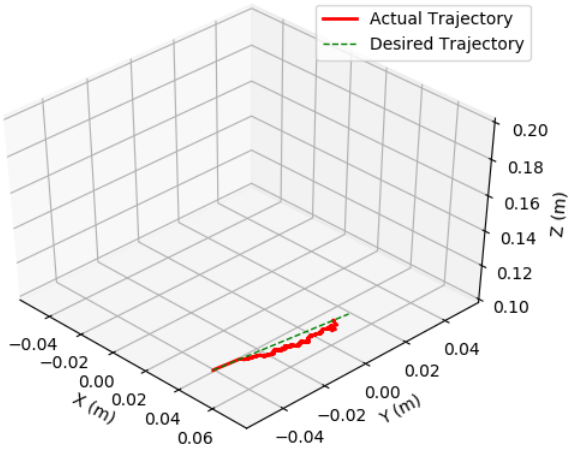

Figure B.3: Error in Euclidean distance of the pertubed CTR at $1\,kHz$.

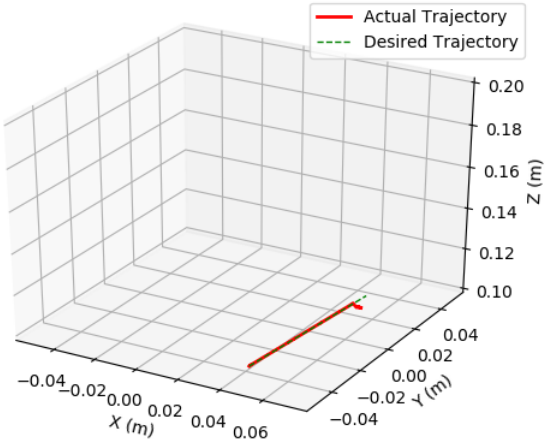

Figure B.4: Error in Euclidean distance of the pertubed CTR at $10\,kHz$.