

A Learning-Based Controller for a Tendon-Driven  
Parallel Continuum Robot

Thesis Interim Report

Spencer Teetaert

Advisor: Professor Jessica Burgner-Kahrs

January 30, 2023

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Background</b>	<b>2</b>
2.1	Continuum Robots . . . . .	3
2.1.1	Tendon-Driven Continuum Robots . . . . .	3
2.2	Parallel Continuum Robots . . . . .	4
2.2.1	Robot Configurations . . . . .	4
2.3	Modelling Methods . . . . .	4
2.3.1	Constant Curvature Assumption . . . . .	5
2.3.2	Variable Curvature Representation . . . . .	5
2.3.3	Tendon Constraints . . . . .	6
2.4	Open Continuum Robotics . . . . .	6
2.5	Learning-Based Control . . . . .	6
2.5.1	Data-driven approaches . . . . .	7
2.5.2	Deep learning approaches . . . . .	7
2.6	Summary . . . . .	8
<b>3</b>	<b>Progress to Date</b>	<b>8</b>
3.1	Physical Prototype . . . . .	9
3.1.1	Kinematic Structure of Prototype . . . . .	9
3.1.2	Hardware Implementation . . . . .	10
3.1.3	Maximum Curvature Test . . . . .	10
3.2	Baseline Kinematic Model . . . . .	11
3.3	Control Loop . . . . .	12
3.4	Experimental Setup . . . . .	13
3.5	Learning-Based Models . . . . .	15
3.6	Key Assumptions and Limitations . . . . .	16
<b>4</b>	<b>Future Work</b>	<b>17</b>
4.1	Data Collection . . . . .	17
4.2	Experiments . . . . .	17
4.2.1	Model Architectures . . . . .	18

4.2.2	Data Types . . . . .	18
4.2.3	Open Loop Hardware Validation . . . . .	18
4.3	Expected Results . . . . .	19
4.4	Project Timeline . . . . .	20
<b>5</b>	<b>Conclusion</b>	<b>20</b>

## List of Figures

1	Continuum robot examples (Source: [14]) . . . . .	3
2	Prototype setup . . . . .	9
3	Maximum Curvature Tests . . . . .	11
4	Cascade control loop . . . . .	13
5	Digital implementation of baseline model . . . . .	14
6	Trajectory A* search (left) and joint interpolation (right) results . .	14
7	Learning-based control loop options . . . . .	15
8	Open loop controller test configuration . . . . .	19

## List of Tables

1	Continuum robot model comparison . . . . .	8
2	Maximum curvature arm test results . . . . .	10

# 1 Introduction

Continuum Robots (CRs) are a current topic of research because of their inherent compliance, ability to be manufactured on sub-millimeter scales, and ability to have non-linear shape deformations [1], [2]. This makes them ideal robots to be used in surgical and inspection applications [3], [4]. Because of a CR’s compliance, it is limited in the force it can exert on the environment through a given end effector. Parallel Continuum Robots (PCRs) see multiple CRs attached at a common end effector, maintaining system compliance while enabling the robot to apply higher forces to the environment [2], [5]. This thesis will explore the design and modelling of a two-arm parallel tendon-driven planar continuum robot and propose a learning-based control method for dynamic control of the robot.

Forward and inverse kinematic modelling of a robot can be used in model-based control to move a robot with specified motions. Dynamic modelling allows for the consideration of forces on a system. Current models of CRs are unable to account for a number of complex factors in the system, such as internal robot friction, surface friction, and external loads, without significant computational overhead [6]. To achieve computation speeds that enable real-time control, often assumptions are made to simplify the robot’s static model to a simple kinematic one [7] or to represent the robot’s state with geometric approximations [6]–[8]. These simplifications greatly improve computation time while sacrificing model predictive accuracy. It is desirable to have the means to model these systems in real-time without making accuracy-sacrificing assumptions that seldom hold true in real-world applications.

The goal of this thesis is to develop a real-time learning-based controller for a planar PCR that improves upon the accuracy of existing real-time controllers for similar systems. Three objectives to achieve this goal are:

1. Re-design and validate a prototype for a parallel planar tendon-driven continuum robot
2. Establish a real-time baseline controller utilizing PID control and the constant curvature assumption
3. Develop and evaluate the performance of a learning-based controller

The project was started with the redesign of an existing planar parallel tendon-driven continuum robot initially constructed at the Continuum Robotics Lab [9]. The robot was prototyped and manufacturing began but the model was never completed or tested. As the robot was partially completed at the start of this thesis, justification for some of the design choices of this particular robot is outside of the scope of this report. Completion and validation of the physical prototype is the first major objective of this thesis.

Establishing a baseline controller serves to act as both a comparison point to the learning-based controller and to enable reasonable control of the robot for data collection. A baseline model has been developed using techniques consistent with other works in the CR field. Evaluation of the effectiveness of this baseline controller must be completed for use as a reference for future approaches.

Future work will explore the development and evaluation of a learning-based controller for the robot prototype to compensate for unmodelled effects in the kinematic model of a planar parallel continuum robot [8]. Learning makes sense in this scenario due to its relatively low computation time to approximate complex models. Data will be collected on the physical system by generating a sample of control inputs and recording the end effector position and motor feedback over time. Trajectories will be restricted to enforce  $C^4$  smoothness [10] to be in line with standard robotics practice. Collection of end effector position data will follow the procedure in [11] and collection of motor feedback as in [12].

**Document structure** In Section 2 a background of concepts is provided alongside a review of current state-of-the-art modelling methods. Section 3 highlights current work on the project as well as results from initial prototype validation. Lastly, Section 4 will discuss the project plan up until completion.

## 2 Background

In this section we consider a portion of available literature on both CRs and control to motivate the use of learning-based control for the PCR considered in this work. As learning-based control for a PCR configuration has not yet been studied in

the available literature, the scope of this section is only to provide a background of relevant topics and not a comprehensive review of all available control and modelling methods.

## 2.1 Continuum Robots

CRs can be defined as "an actuatable structure whose constitutive material forms curves with continuous tangent vectors." [3] CRs are robots with an infinite number of joints and degrees of freedom. By using flexible components, CRs are inherently compliant, making them useful in applications where interactions with stiff joints may cause damage to the environment. They are also able to be manufactured on sub-millimeter scales. As such, these robots have seen use in medical and inspection applications [1], [3], [4]. Continuum robots come in a variety of configurations, materials, and actuation methods [13]. Modelling methods for continuum robots are discussed in Section 2.3. The subset of continuum robots pertaining to this thesis is considered.

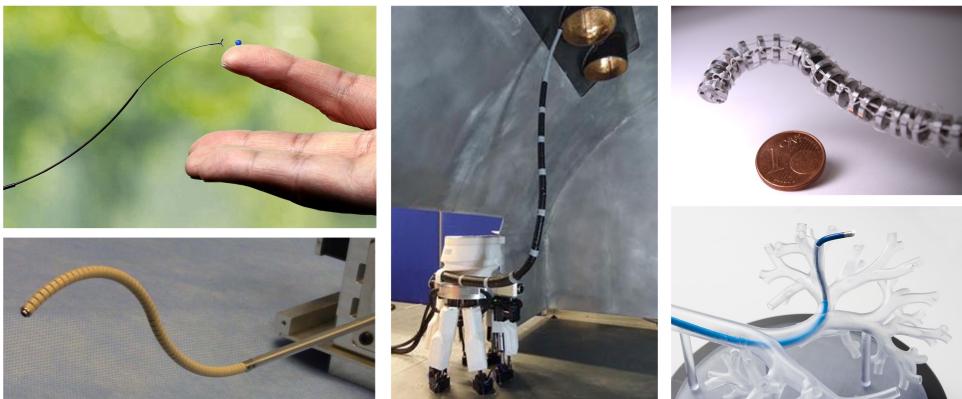


Figure 1: Continuum robot examples (Source: [14])

### 2.1.1 Tendon-Driven Continuum Robots

Tendon-Driven Continuum Robots (TDCR) are a class of CRs that are actuated by contracting a tendon that is attached to a point along the robot's backbone. The backbone is typically a highly elastic rod or beam that provides stiffness to the robot while remaining compliant. The tendon is constrained to be a fixed distance

from the backbone by a series of spacer disks. Contracting the tendon results in the backbone bending towards the side the tendon is offset on [6], [15]. An "arm" of a TDCR refers to a single beam that can be actuated independently of other arms in the robot.

## 2.2 Parallel Continuum Robots

PCRs use multiple arms joined together to increase the stiffness and maximum applied force of a CR while maintaining the inherent compliance that CRs have to offer [2].

### 2.2.1 Robot Configurations

PCRs can take on a wide variety of configurations. In [5], a six-beam parallel continuum robot is proposed and modelled based on Cosserat rod theory. Flexible rods are translationally pushed and pulled to achieve actuation. A tracking accuracy of 2.89% of the length of one arm of the robot was achieved. [8] introduces a three-beam tendon-driven parallel continuum robot with six degrees of freedom at the end effector. The authors model each beam using a geometric kinematic model based on the constant curvature assumption. A planar configuration reduces the task space to a 2D plane. These robots have at most three degrees of freedom at the end effector; two spatial components and a rotational one. In [7], a planar parallel continuum robot is proposed along with a geometric model controller based on the constant curvature assumption. It achieves an accuracy of 1% of the length of one arm of the robot.

## 2.3 Modelling Methods

It is desirable to construct a forward and inverse kinematic model when controlling a robot. A forward kinematic model provides an expression for the end effector position given a robot's joint variables. It is used to estimate the end effector position of a robot during operation when direct measurements are not available. An inverse kinematic model solves for a robot's joint variables given a desired end effector position. It is used in control tasks to move the robot into a desired configuration.

In both cases, the accuracy of the model is determined by a number of factors including the simplifying assumptions made in exchange for increased computational efficiency [6].

There are three major types of robot models that we consider: (1) kinematic models, (2) static models, and (3) dynamic models. Kinematic models model the shape of the robot without consideration of forces. Static models consider the robot at points of equilibrium, where the robot is not moving and has zero net force. Dynamic models model the robot as it's moving, accounting for transient effects (such as spring forces, friction, air resistance, etc.).

In rigid robotics, simple linear models exist that accurately model the robot's motion. CRs do not have as simple a time due to their non-linear deformation. As such, larger assumptions and/or more complex models must be used for even the simplest of CRs [6]. As this is an emerging research field, the accuracy of models for CRs is not yet comparable to conventional industrial robots.

### 2.3.1 Constant Curvature Assumption

The most common and convenient kinematic modelling approach for continuum robots uses the constant curvature assumption [16]. It assumes that the curvature throughout the entire length of an arm is constant. This assumption greatly reduces the complexity of the forward and inverse kinematic models of a continuum robot.

**Euler Beam Theory** Euler beam theory can be used to create a static model assuming that the effects of shear and twist are negligible [6]. The constant curvature assumption allows for computationally efficient implementations of the inverse kinematics problem. It does however result in a number of inaccuracies as beams under high bending do not have constant curvature. Modelling a continuum robot using the constant curvature assumption and Euler beam theory has been shown to result in poorer accuracy compared to using a variable curvature model [6].

### 2.3.2 Variable Curvature Representation

Without losing generality, one can model the curvature of a continuum robot using a variable curvature representation. This representation makes no assumptions about

the backbone shape as it can model any point along it with six degrees of freedom [6].

**Cosserat Rod Theory** The Cosserat theory of elastic rods extends the variable curvature kinematic representation into a static model. Modelling in this way accounts for shear deformations, something Euler beam theory did not. While the Cosserat rod theory approach has been shown to produce excellent accuracy in continuum robot applications [6], [17]–[19], it does so at great computation costs, resulting in the inability to be used for real-time control [6], [18]. A number of implementations for the cosserat theory model have been realized with accuracies ranging from 12%[19] to  $4 \times 10^{-10}$  m -  $8 \times 10^{-10}$  m Root Mean Squared Error (RMSE) for a robot of length 30 cm[18].

### 2.3.3 Tendon Constraints

When modelling tendon length with the constant curvature assumption, one can either assume the tendons are infinitely constrained or finitely constrained. Infinitely constrained tendons result in simpler expressions while finitely constrained tendons are more accurate to feasible implementations [6].

## 2.4 Open Continuum Robotics

With the rise in popularity of continuum robots, the Open Continuum Robotics project was created to reduce barriers to entry into the field [20]. The project hosts a variety of open-source materials for building and developing continuum robots, including an actuation unit [12]. The work of this thesis is open source as well and aims to contribute to the Open CR project.

## 2.5 Learning-Based Control

Machine learning has emerged as a useful tool for estimating complex functions using data collected from a system. In robotics, learning has been used to develop controllers for complex systems that are difficult to model. Deep learning approaches have shown promise in modelling complex dynamic behaviour for controller design in robotics [21]. Because data-driven approaches can estimate arbit-

trarily complex functions [22] they are thought to be useful in continuum robots. Learning-based applications in continuum robotics have primarily focused on learning the inverse statics and kinematics of a robot [23]. With each of the results reviewed here, one must be weary of directly comparing results directly as large variations in performance can arise from changes in robot configuration, actuation type, and construction. In each of the works reviewed, the proposed learning-based model outperformed baseline approaches on the same robot.

### 2.5.1 Data-driven approaches

In [24], the authors use a model-free approach using an adaptive Kalman Filter. This uses sensor data to update the model's uncertainty estimations during operation, resulting in a more controller that had an experimental tracking RMSE of 1.52 mm (with a 20 cm - 25 cm long pneumatic continuum robot). In [25], the authors implement three regression models to learn the inverse kinematics of a flexible surgical robot, achieving a top RMSE of 2.1275 mm with a KNN regression. The authors do not report the total robot length.

### 2.5.2 Deep learning approaches

For deep learning approaches to work, a large amount of data or an accurate simulation environment must be available. As the dynamic effects of a continuum robot are what is difficult to model, simulation environments are not readily available. In [11], a dataset is released for concentric tube continuum robots. The authors propose a baseline learning model that achieves tip tracking errors of 0.74 mm (or 0.4% of the robot's length). Using deep neural networks for approximating the inverse statics equation is the approach used in [26]. The authors are able to achieve errors of around 7 mm<sup>1</sup> for an underwater soft continuum robot (or about 2.5% of the robot's length). For both of these implementations, a large amount of data is required that maps the robot's end effector position to its joint configuration, marking a large drawback to deep learning-based models.

---

<sup>1</sup>Number, as reported, is an estimation by the thesis author. See reference paper for complete results breakdown.

Modelling Method	Accuracy	Complexity
Constant Curvature / Euler Beam Theory	Low	Low
Variable Curvature / Cosserat Rod Theory	High	High
Infinitely Constrained Tendon	Medium	Low
Finitely Constrained Tendon	High	Medium
Data-Driven Approaches	Medium	Medium
Deep Learning Approaches	High	Medium

Table 1: Continuum robot model comparison

## 2.6 Summary

PCR design is still an open area of research. Work has focused on the benefits of these robots in their relevant application spaces. Research modelling and control for CRs have largely focused on single-beam CRs. A research gap exists for modelling and control of PCRs that is unaddressed by current classical modelling methods. Large model assumptions allow for fast but inaccurate control, not making these assumptions results in infeasible computation speeds for real-time control, and data-driven approaches to date are underdeveloped for parallel continuum robots. This thesis aims to address this gap by developing a novel deep learning-based controller for a tendon-driven planar parallel continuum robot, expecting to achieve high accuracy at low computation speeds relative to a baseline implemented on the same robot.

## 3 Progress to Date

To date, progress made on this thesis<sup>2</sup> mainly focused on (1) design, assembly, and validation of the physical robot prototype, (2) development of a baseline control model, and (3) development of data gathering procedure and tools.

---

<sup>2</sup>All project code is available freely at: [https://github.com/spencerteetaert/pcr\\_control](https://github.com/spencerteetaert/pcr_control).

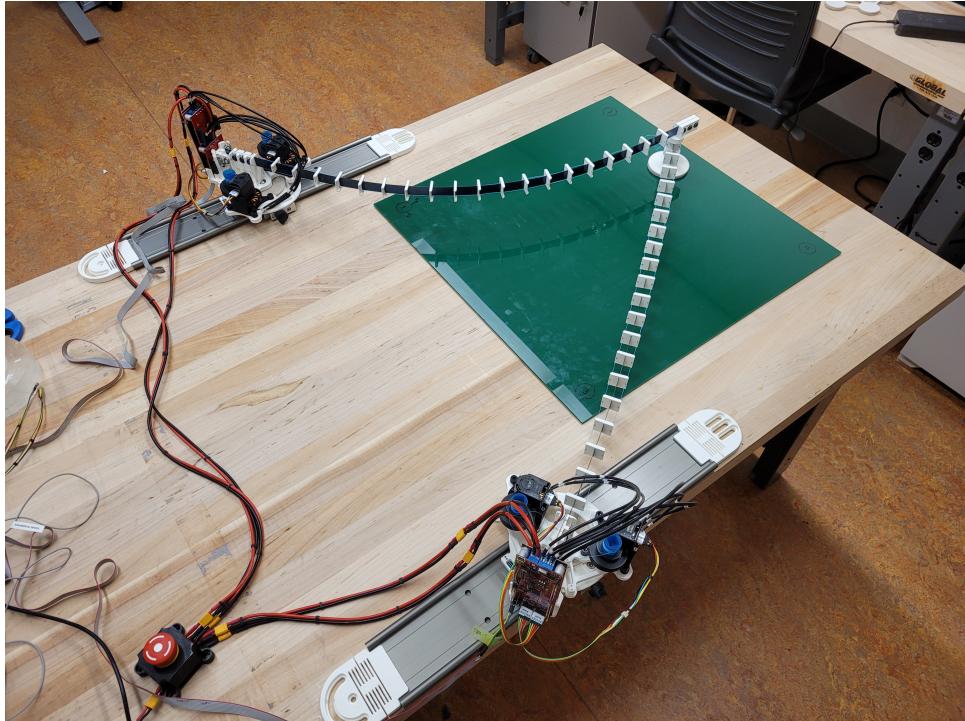


Figure 2: Prototype setup

### 3.1 Physical Prototype

#### 3.1.1 Kinematic Structure of Prototype

The robot developed for this thesis is a two-arm planar TDCR. The arms are each 80 cm long. Each arm is capable of being driven by two tendons that are 1 cm from the beam. Each arm is attached to its base and the end effector with a bearing, allowing the joints to rotate freely. The bases for each arm are positioned 60 cm apart. The end effector has two degrees of freedom along a table surface. The table workspace has an acrylic sheet used to reduce the friction of the table on the end effector. Beneath the table surface, an Aurora electromagnetic tracking system<sup>3</sup> is positioned. The tracker is positioned to ensure complete tracking coverage of the robot's task space.

---

<sup>3</sup>Electro magnetic tracking system (AURORA, Northern Digital Inc., ON, Canada)

### 3.1.2 Hardware Implementation

Each arm contains two motors (T-Motor Antigravity 4004) controlled by an off-the-shelf microprocessor (TI micro-controller evaluation board). The actuation setup comes from the Open Continuum Robot project [12], [20]. Each controller communicates with a real-time PC through the CAN bus protocol. Each tendon is actuated by a motor with an encoder that is used to provide system feedback. Due to limits on the IO of the computer that is assigned to this project, only a single tendon can be used for each arm. The robot is powered using a 24V, 10A power supply. An emergency stop is installed within reach of the operator for safety. The arms are thin beams made from an elastic material. The motor mounting brackets are 3D printed and house a 64:1 gearbox.

### 3.1.3 Maximum Curvature Test

Before the 64:1 gearbox was designed and installed, the motors had a 4:1 gear ratio. An experiment was conducted to determine the arm length that enabled the largest range in the end effector position. Knowing the maximum and minimum distance that can be reached from an arm's base is required to set workspace bounds for the robot controller. The maximum length is trivially given by the length of the arm. To find the minimum distance, a maximum curvature is applied by sending the highest torque command to a given motor. The curvature that equates the bending forces to the maximum motor torque is taken as the maximum curvature. In this position, the distance between the end effector and arm base was measured. The results are displayed in Table 2 and Figure 3. The arms were left at 80 cm length as no notable gain in performance was observed at either test length.

Extended Length	Fully Contracted Length	Contraction Length
80 cm	(59 ± 1) cm	(21 ± 1) cm
55 cm	(33 ± 1) cm	(22 ± 1) cm

Table 2: Maximum curvature arm test results

This experiment showed the torque output of the control motors resulted in a significantly smaller range of motion than what was initially expected. In response,

a new gearbox was designed and manufactured to increase the maximum torque output of the robot. With the new gearbox, the robot is able to contract its arm completely with 0cm between the end effector and base.

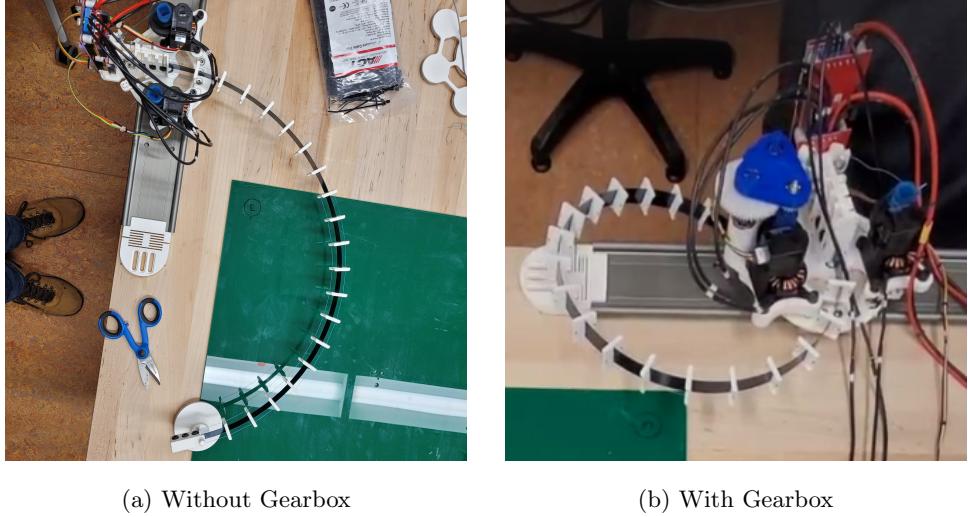


Figure 3: Maximum Curvature Tests

### 3.2 Baseline Kinematic Model

A model using the constant curvature assumption [6], [16] is used as a baseline model for this parallel continuum robot. This model is used to solve the inverse kinematic problem. This is used to determine control signals for the robot during the data generation, enabling the ability to control the position of the end effector (with some error). Equation (1) gives a general expression relating the curvature of a constant curvature beam and a beam's base and end position [8]. To solve for  $\kappa$  the end effector position must be known or assumed. For the parallel continuum case, one solves Equation (1) for each of the robot's arms.

$$0 = \frac{2}{L\kappa} \sin\left(\frac{L\kappa}{2}\right) - \frac{\|x_{ee} - x_b\|}{L} \quad (1)$$

where:

$$\begin{aligned}
L &= \text{Arm length [m]} \\
\kappa &= \text{Arm curvature [m}^{-1}\text{]} \\
x_{ee} &= \text{End effector position } [\in R^2] \\
x_b &= \text{Arm base position } [\in R^2]
\end{aligned}$$

For a model to be used in the control setting it must output a joint variable. Equation (3) solves for the tendon length required to achieve a given constant curvature. Two solutions are provided, one for each side of the beam. Equation 2 is used to find the tendon length given a beam's curvature, assuming an infinitely constrained tendon along the length of the beam.

$$\Delta l_t = L r_r \kappa \quad (2)$$

where:

$$\begin{aligned}
\Delta l_t &= \text{Change in tendon length [m]} \\
r_r &= \text{Tendon rib radius [m]}
\end{aligned}$$

We can extend Equation 2 to consider our motor's gear ratio. Dividing by the motor's spindle radius gives an expression for the change in motor rotation required to achieve the desired tendon length displacement. This value is the joint variable for our PCR.

$$q = \pm \kappa G L \frac{r_r}{r_m} \quad (3)$$

where:

$$\begin{aligned}
q &= \text{Motor rotation [rad]} \\
G &= \text{Motor gear ratio} \\
r_m &= \text{Motor spindle radius [m]}
\end{aligned}$$

### 3.3 Control Loop

The robot is controlled using the semi-open-loop cascade controller shown in Figure 4. The inner loop consists of a closed loop motor control loop. A PID controller uses motor encoder feedback with a reference signal from a planner to determine the current to be sent to each motor. An open loop controller uses the baseline model

to determine required joint displacements to achieve a desired end effector position. During regular operation, the end effector position is not available to the controller. In Figure 4, solid lines indicate information used in the control loop while dashed lines refer to data that is gathered during operation.

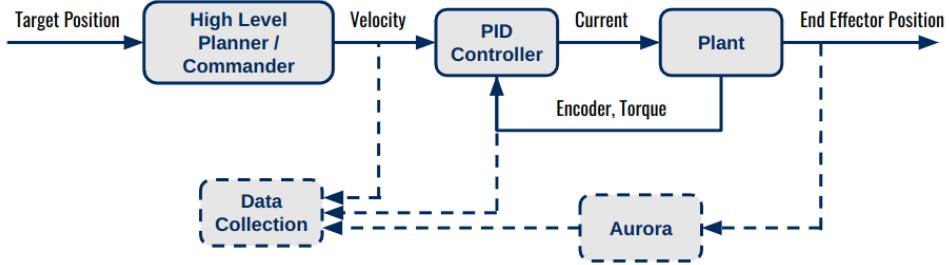


Figure 4: Cascade control loop

### 3.4 Experimental Setup

**Data Collection** An Aurora electromagnetic tracker is used to collect information on the robot’s end effector position. This data is never used in closed-loop control as it is assumed to not be available in the applications that this project is targeting. It is used as the ground truth position when learning control models and is collected at 5 Hz. Motor encoders provide feedback on the motor’s position and velocity at 100 Hz. Current draw data is collected with the microprocessor at 100 Hz. Each piece of data is saved using the computer’s timestamp at the time of the measurement.

**Trajectory Generation** A digital model (Figure 5) of the robot’s task space and state configuration is used to generate trajectories to run on the physical robot. A random trajectory is generated by selecting a random point in the robot’s task space and using A\* to solve for a path between the current end effector position and the target point. A noise map is generated using Perlin noise [27] and used as a cost map in the A\* algorithm to add task space variations in generated trajectories. The solution path from A\* in discretized task space is converted to joint space using the baseline inverse kinematic model. The solution is found numerically using

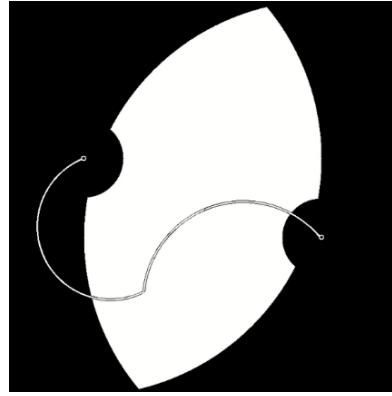


Figure 5: Digital implementation of baseline model

`scipy.optimize.fsolve`<sup>4</sup>. A continuous-time joint trajectory is found using a similar method as [10] to ensure  $C^4$  smoothness.  $C^4$  smoothness is required for exact reproducibility of desired trajectories, as well as helps prevent structural oscillations and requires less energy [10]. A sample of this process is shown in Figure 6. This trajectory is sampled at a given command frequency and is used to set the reference motor velocities in the PID controller. This same method can be used to generate test trajectories consisting of a square, a circle, a sinusoidal pattern, and a zig-zag.

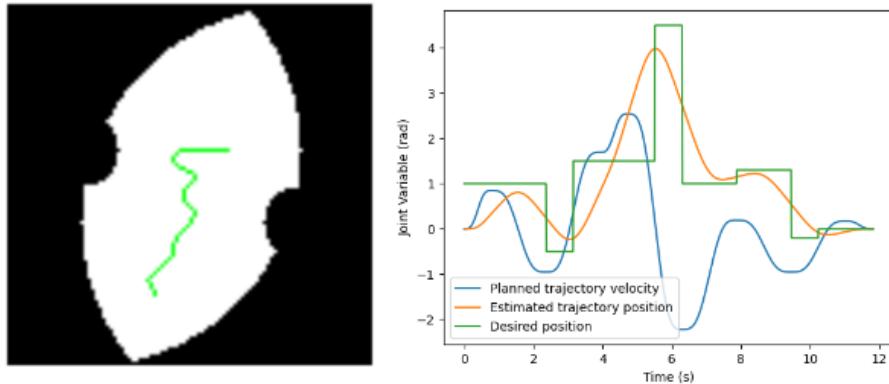


Figure 6: Trajectory A\* search (left) and joint interpolation (right) results

---

<sup>4</sup>[https://github.com/scipy/scipy/blob/v1.10.0/scipy/optimize/\\_minpack\\_py.py#L48-L181](https://github.com/scipy/scipy/blob/v1.10.0/scipy/optimize/_minpack_py.py#L48-L181)

### 3.5 Learning-Based Models

Two primary learning-based control methods have been proposed. The first trains a model to replace both the inverse kinematic model and PID controller proposed in the baseline implementation, while keeping the high-level planner. The second trains a model to replace just the inverse kinematic model and leaves the low-level control loop as is. There are expected pros and cons to both approaches.

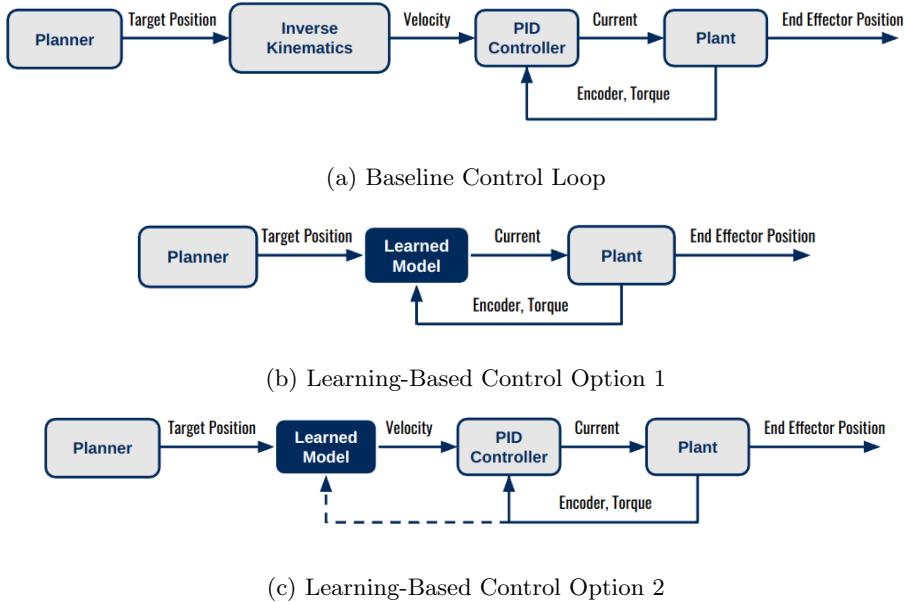


Figure 7: Learning-based control loop options

**Option 1** In the first option (7b), the learned model is responsible for more aspects of control. This gives the model more flexibility in its approach to control as it has access to lower-level system controls (current commands). However, elements of control that a PID controller accounts for such as limiting the change in command signals or reducing accumulated error over time are also required to be learned. This approach will require larger amounts of data and may result in less smooth control solutions.

**Option 2** In the second option (7c), the PID controller remains in the loop. This approach is likely to see the learning model replace the inverse kinematic model from

Equation (1). This approach is expected to require less data and provide smoother results, but may be limited in accuracy as the inclusion of a PID controller restricts the system’s output. This approach could additionally use motor encoder feedback as an additional data source.

### 3.6 Key Assumptions and Limitations

**Constant curvature assumption** A key assumption in the baseline implementation is that the constant curvature model sufficiently models the robot’s kinematics. Figure 3 can discredit the constant curvature assumption as the arm’s curvature visually increases near the end effector. This assumption ignores dynamic effects and the non-linear nature of a beam under high bending forces in exchange for fast compute times. A learning-based solution aims to account for these shortcomings.

**Baseline model for data collection** The baseline model is used in order to collect the data that will be used in the learning pipeline. The data collected will therefore only contain robot states accessible using the baseline model which has the potential to result in systemic missing state space data.

**Single motor control** By using only a single motor for each arm, we rely on the elastic forces of the beam to return the arm to a state of zero curvature. The dynamic effects of friction will result in this not being the case. The robot will be unable to reach a zero curvature state during regular operation. This only occurs on the boundary of the robot’s workspace. A maximum extension of 77cm was observed during the gearbox maximum curvature test, resulting in 3cm of lost workspace. This is a compromise made to allow the project to stay on pace.

**Constant curvature as a baseline** As no other research provides a baseline that we can compare to, this work proposes our own baseline. The selection of the constant curvature model presented follows from its ease of derivation and implementation. Other models, such as those based on a variable curvature backbone representation may provide more accurate results than our selected baseline at the expense of increased computation [6]. These models also have their own challenges representing the dynamic effects of continuum robots. Future works may look at

the results of these models on this system but it is deemed as beyond the scope of this thesis.

## 4 Future Work

### 4.1 Data Collection

Data must be collected for the robot within its operating range. The data schema must first be finalized. Data being collected includes:

1. Motor current commands
2. Baseline end effector position prediction
3. Motor encoder position and velocity feedback
4. End effector position
5. Timestamp for each piece of data collected

Data will be generated with three different collection policies:

1. Random trajectory generation (described in Section 3.4)
2. Task space raster scan pattern (sweeping task space back and forth)
3. Typical controls testing patterns (circle, square, sinusoidal, zig-zag, etc.)

The first two methods provide a reasonably sized cover of the robot's task space while the third can be held out as a test set. A holdout set will be generated by removing a random sample of 20% of the data points collected in the first two methods. This holdout set will be used for model validation and selection. The third method will also be held out and will only be used for final model testing. The data split will be approximately 60/20/20 for training/validation/testing.

### 4.2 Experiments

The following sections describe experiments that are to be completed given the gathered dataset. RMSE on a holdout dataset will be used as the comparison metric for each experiment (Equation (4)).

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^N \|x_{actual} - x_{desired}\|^2}{N}} \quad (4)$$

where:

$x_{actual}$  = End effector ground truth position [ $\in R^2$ ]

$x_{desired}$  = End effector desired position [ $\in R^2$ ]

$N$  = Number of test data points

A run-time constraint of 100Hz will be applied to each model to ensure that the models do not slow down the real-time execution rate of the robot. Any model that runs slower than this will not be considered a valid solution.

#### 4.2.1 Model Architectures

Which model architectures will create the best results? Architectures of interest include support vector machines (SVM), feed-forward neural networks (FNN), and recurrent neural networks (RNN). Each architecture will be experimented on in both the Option 1 and Option 2 learning control loop configurations. The architecture that has the lowest RMSE value on the holdout set that satisfies the run time constraint will be considered the best.

#### 4.2.2 Data Types

Which data is useful for this type of problem? The available collectable data may contain information that is not relevant to the tasks of learning a controller for this system. We will determine which data sources are most useful by training several equivalent model architectures using different available data as inputs. The input data types that have the lowest RMSE value on the holdout set that satisfies the run time constraint will be considered the best.

#### 4.2.3 Open Loop Hardware Validation

In order to compare controllers they must be implemented on the robot. A supervised learning-based controller is unable to provide test results on collected data as the controller is an agent in the system that must be able to affect future data

points. As such, testing each model involves running the test trajectories on the physical robot with the controller in the loop as shown in Figure 8.

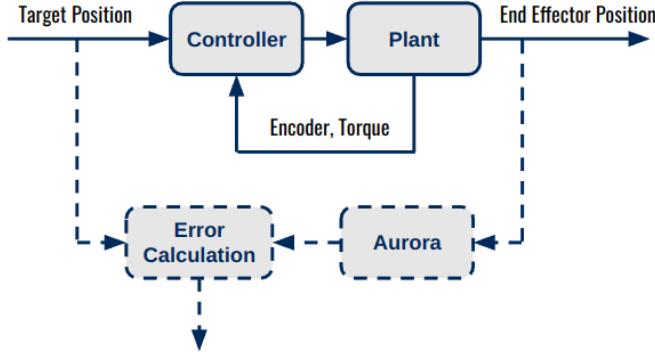


Figure 8: Open loop controller test configuration

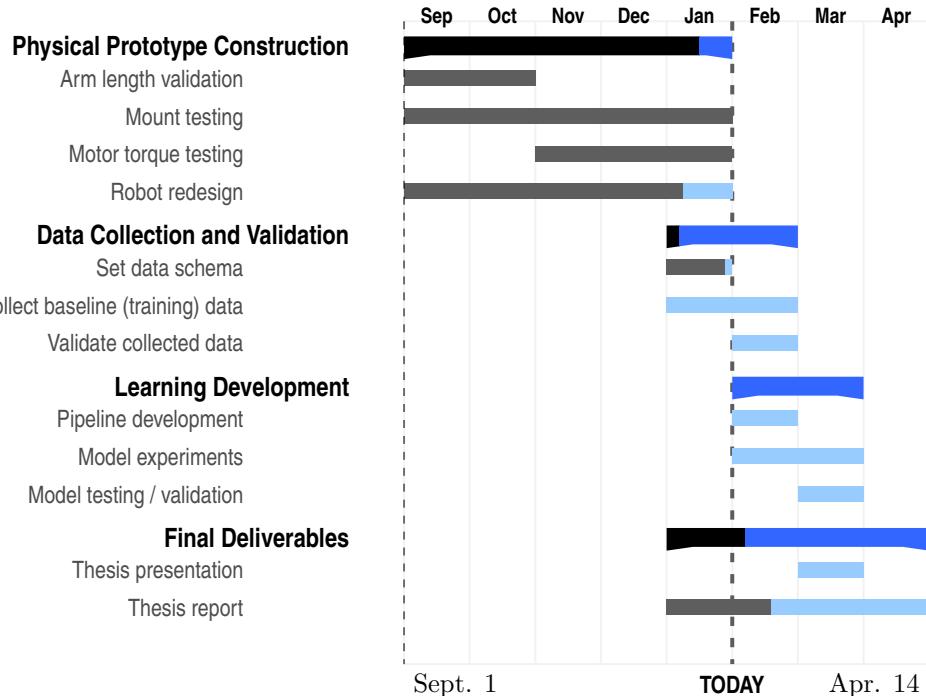
### 4.3 Expected Results

The constant curvature assumption is most accurate at small curvature values. It is expected that the baseline error will increase greatly under high curvature situations. Some of the factors that can cause this is higher tendon friction forces and friction applied to the end effector. Additionally, as the baseline model does not account for the dynamic effects of the robot in motion, the error is expected to be great during high velocity and acceleration. Increased friction is expected to play a large part in the baseline modelling error. The learning-based approaches proposed have the potential to model arbitrarily complex dynamic systems. We expect learning-based controllers to greatly improve upon the baseline accuracy when the model is moving fast or bending far. We anticipate these models will be able to more accurately estimate the system kinematics and dynamics given the computation constraints.

While there does not exist a direct comparison in the literature to our parallel planar system, several references suggest that accuracy results in the range of 0.5-2% of the robot's length could be expected [11], [23], [26]. As this is the first learning-based controller being proposed for a PCR, accuracy comparable with current state-of-the-art for other CRs will be a soft target accuracy.

## 4.4 Project Timeline

The project up until today has focused on developing the physical prototype and software tools for the project. Moving forward the emphasis will be on data collection and experimentation with learning-based controllers. A project timeline is proposed in the chart below.



## 5 Conclusion

In this report, we have considered the problem of modelling and control for a planar TDCR. Improving modelling accuracy and computation efficiency for this robot will contribute to CR research in all of their application spaces. We have motivated the use of learning-based methods for solving this problem as well as set out reasonable expectations for model performance. We have highlighted current progress in this project and have laid out future steps to ensure a successful conclusion within the semester. This work will lay foundations for future researchers looking to experiment with or develop real-time controllers for PCRs.

## References

- [1] R. J. Webster, A. M. Okamura, and N. J. Cowan, “Toward active cannulas: Miniature snake-like surgical robots,” in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 2857–2863. DOI: [10.1109/IROS.2006.282073](https://doi.org/10.1109/IROS.2006.282073).
- [2] S. Lilge, K. Nuelle, J. A. Childs, K. Wen, C. Rucker, and J. Burgner-Kahrs, “Parallel continuum robots: A survey,” In Revision.
- [3] J. Burgner-Kahrs, D. C. Rucker, and H. Choset, “Continuum robots for medical applications: A survey,” *IEEE Transactions on Robotics*, vol. 31, no. 6, pp. 1261–1280, 2015. DOI: [10.1109/TRO.2015.2489500](https://doi.org/10.1109/TRO.2015.2489500).
- [4] X. Dong, D. Axinte, D. Palmer, *et al.*, “Development of a slender continuum robotic system for on-wing inspection/repair of gas turbine engines,” *Robotics and Computer-Integrated Manufacturing*, vol. 44, pp. 218–229, 2017, ISSN: 0736-5845. DOI: <https://doi.org/10.1016/j.rcim.2016.09.004>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0736584516300801>.
- [5] C. E. Bryson and D. C. Rucker, “Toward parallel continuum manipulators,” in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 778–785. DOI: [10.1109/ICRA.2014.6906943](https://doi.org/10.1109/ICRA.2014.6906943).
- [6] P. Rao, Q. Peyron, S. Lilge, and J. Burgner-Kahrs, “How to model tendon-driven continuum robots and benchmark modelling performance,” *Frontiers in Robotics and AI*, vol. 7, 2021, ISSN: 2296-9144. DOI: [10.3389/frobt.2020.630245](https://doi.org/10.3389/frobt.2020.630245). [Online]. Available: <https://www.frontiersin.org/articles/10.3389/frobt.2020.630245>.
- [7] K. Nuelle, T. Sterneck, S. Lilge, D. Xiong, J. Burgner-Kahrs, and T. Ortmair, “Modeling, calibration, and evaluation of a tendon-actuated planar parallel continuum robot,” *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5811–5818, 2020. DOI: [10.1109/LRA.2020.3010213](https://doi.org/10.1109/LRA.2020.3010213).
- [8] S. Lilge, K. Nüll, G. Boettcher, S. Spindeldreier, and J. Burgner-Kahrs, “Tendon actuated continuous structures in planar parallel robots: A kine-

- matic analysis,” *Journal of Mechanisms and Robotics*, vol. 13, Nov. 2020. DOI: 10.1115/1.4049058.
- [9] *Continuum robotics lab*. [Online]. Available: <https://crl.utm.utoronto.ca/>.
- [10] R. M. Grassmann and J. Burgner-Kahrs, “Quaternion-based smooth trajectory generator for via poses in  $S E(3)$  considering kinematic limits in cartesian space,” *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4192–4199, 2019. DOI: 10.1109/LRA.2019.2931133.
- [11] R. M. Grassmann, R. Z. Chen, N. Liang, and J. Burgner-Kahrs, “A dataset and benchmark for learning the kinematics of concentric tube continuum robots,” in *Workshop on Learning from Diverse, Offline Data*, 2022. [Online]. Available: [https://openreview.net/forum?id=DW9uz\\_GZ0og](https://openreview.net/forum?id=DW9uz_GZ0og).
- [12] F. Grimminger, A. Meduri, M. Khadiv, *et al.*, “An open torque-controlled modular robot architecture for legged locomotion research,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3650–3657, Apr. 2020. DOI: 10.1109/lra.2020.2976639. [Online]. Available: <https://doi.org/10.1109/2Flra.2020.2976639>.
- [13] J. Zhang, Q. Fang, P. Xiang, *et al.*, “A survey on design, actuation, modeling, and control of continuum robot,” *Cyborg and Bionic Systems*, vol. 2022, 2022. DOI: 10.34133/2022/9754697. eprint: <https://spj.science.org/doi/pdf/10.34133/2022/9754697>. [Online]. Available: <https://spj.science.org/doi/abs/10.34133/2022/9754697>.
- [14] *Types of continuum robots*. [Online]. Available: <https://www.cs.toronto.edu/~jbk/opencontinuumrobotics/101/2022/11/04/cr-types.html>.
- [15] A. Hemami, “Studies on a light weight and flexible robot manipulator,” *Robotics*, vol. 1, no. 1, pp. 27–36, 1985, ISSN: 0167-8493. DOI: [https://doi.org/10.1016/S0167-8493\(85\)90306-7](https://doi.org/10.1016/S0167-8493(85)90306-7). [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167849385903067>.
- [16] I. Robert J. Webster and B. A. Jones, “Design and kinematic modeling of constant curvature continuum robots: A review,” *The International Journal of Robotics Research*, vol. 29, no. 13, pp. 1661–1683, 2010. DOI: 10.1177/

0278364910368147. eprint: <https://doi.org/10.1177/0278364910368147>. [Online]. Available: <https://doi.org/10.1177/0278364910368147>.
- [17] D. Cao and R. W. Tucker, “Nonlinear dynamics of elastic rods using the cosserat theory: Modelling and simulation,” *International Journal of Solids and Structures*, vol. 45, no. 2, pp. 460–477, 2008, ISSN: 0020-7683. DOI: <https://doi.org/10.1016/j.ijsolstr.2007.08.016>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0020768307003253>.
- [18] A. A. Alqumsan, S. Khoo, and M. Norton, “Robust control of continuum robots using cosserat rod theory,” *Mechanism and Machine Theory*, vol. 131, pp. 48–61, 2019, ISSN: 0094-114X. DOI: <https://doi.org/10.1016/j.mechmachtheory.2018.09.011>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0094114X18311777>.
- [19] S. M. H. Sadati, S. E. Naghibi, A. Shiva, I. D. Walker, K. Althoefer, and T. Nanayakkara, “Mechanics of continuum manipulators, a comparative study of five methods with experiments,” in *Towards Autonomous Robotic Systems*, Y. Gao, S. Fallah, Y. Jin, and C. Lekakou, Eds., Cham: Springer International Publishing, 2017, pp. 686–702, ISBN: 978-3-319-64107-2.
- [20] *Open continuum robotics project*. [Online]. Available: <https://www.cs.toronto.edu/~jbk/opencontinuumrobotics/>.
- [21] A. I. Károly, P. Galambos, J. Kuti, and I. J. Rudas, “Deep learning in robotics: Survey on model structures and training strategies,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 1, pp. 266–279, 2021. DOI: [10.1109/TSMC.2020.3018325](https://doi.org/10.1109/TSMC.2020.3018325).
- [22] K. Hornik, M. Stinchcombe, and H. White, “Multilayer feedforward networks are universal approximators,” *Neural Networks*, vol. 2, no. 5, pp. 359–366, 1989, ISSN: 0893-6080. DOI: [https://doi.org/10.1016/0893-6080\(89\)90020-8](https://doi.org/10.1016/0893-6080(89)90020-8). [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0893608089900208>.
- [23] X. Wang, Y. Li, and K.-W. Kwok, “A survey for machine learning-based control of continuum robots,” *Frontiers in Robotics and AI*, vol. 8, 2021, ISSN:

- 2296-9144. DOI: 10.3389/frobt.2021.730330. [Online]. Available: <https://www.frontiersin.org/articles/10.3389/frobt.2021.730330>.
- [24] M. Li, R. Kang, D. T. Branson, and J. S. Dai, “Model-free control for continuum robots based on an adaptive kalman filter,” *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 1, pp. 286–297, 2018. DOI: 10.1109/TMECH.2017.2775663.
  - [25] W. Xu, J. Chen, H. Y. Lau, and H. Ren, “Data-driven methods towards learning the highly nonlinear inverse kinematics of tendon-driven surgical manipulators,” *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 13, no. 3, e1774, 2017, e1774 RCS-16-0056.R1. DOI: <https://doi.org/10.1002/rcs.1774>. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/rcs.1774>. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/rcs.1774>.
  - [26] M. Giorelli, F. Renda, M. Calisti, A. Arienti, G. Ferri, and C. Laschi, “Neural network and jacobian method for solving the inverse statics of a cable-driven soft arm with nonconstant curvature,” *IEEE Transactions on Robotics*, vol. 31, no. 4, pp. 823–834, 2015. DOI: 10.1109/TRO.2015.2428511.
  - [27] K. Perlin, “Making noise,” GDC Talk, 1999.