

Design Optimization Algorithms for Concentric Tube Robots

Cenk Baykal

Senior Honors Thesis
Department of Computer Science
University of North Carolina at Chapel Hill

April, 2015

Approved by:

Ron Alterovitz, Advisor

Ming Lin, Reader

© 2015
Cenk Baykal
ALL RIGHTS RESERVED

ABSTRACT

CENK BAYKAL: Design Optimization Algorithms for Concentric Tube Robots. (Under the direction of Ron Alterovitz.)

Concentric tube robots are tentacle-like surgical robots that can bend around anatomical obstacles to access hard-to-reach surgical targets. These robots have potential to enable minimally invasive surgical procedures by allowing physicians to access clinical regions that were previously unreachable using traditional instruments. Concentric tube robots are composed of nested, customizable tubes which undergo complicated mechanical interactions that generate tentacle-like motion. As a consequence of this intricate kinematic mechanism, the physical specifications of each of the robots tubes, i.e. the robot's design, significantly affect the shapes that the robot can undertake and the regions it can reach. Customizing the design of these robots can potentially facilitate successful surgical procedures on a variety of patients. In this thesis, we present design optimization algorithms to generate appropriate design parameters on an application- and patient-specific basis.

We consider three design optimization problems. First, we present a design optimization algorithm that generates a concentric tube robot design under which the robot can maximize the reachable volume of a given goal region in the human body. We provide analysis establishing that our design optimization algorithm for generating a single design is asymptotically optimal. Second, we present an algorithm that computes sets of concentric tube robot designs that can collectively maximize the reachable volume of a given goal region in the human body. Third, we introduce an algorithm that generates the set of designs of minimal size such that the designs in the set can collectively reach a physician-specified percentage of the goal region.

Each of our algorithms combines a search in the design space of a concentric tube robot using Adaptive Simulated Annealing with a sampling-based motion planner in the robot's configuration space in order to find a single or sets of designs that enable paths to the goal regions while avoiding contact with anatomical obstacles. We demonstrate the effectiveness of each of our algorithms in a simulated scenario based on lung anatomy and compare our algorithms' performance with that of current state-of-the-art design optimization algorithms.

Dedicated to my family

ACKNOWLEDGMENTS

I would like to express my gratitude and appreciation for all the great people with whom I had the pleasure to work with and learn from during my undergraduate education at UNC. I am truly grateful and honored to have had the opportunity engage in research with some of the brightest people, without whom this thesis would not have been possible.

First and foremost, I would like to thank my thesis advisor, Professor Ron Alterovitz, for his advising and mentoring for over more than 2 years of my undergraduate education. I would like to thank him for giving me the opportunity to join the Computational Robotics Group at UNC as a sophomore. Professor Alterovitz patiently introduced me to the exciting world of research and I am incredibly thankful for his generous time and invaluable advice over the years. In addition to being my research advisor, Professor Alterovitz has been a highly influential figure; namely, I would like to thank Professor Alterovitz for motivating my decision to apply to graduate school.

I would also like to thank Professor Ming Lin not only for her part as a reader on this thesis, but also for her generous role as my research advisor for projects in the GAMMA group at UNC for 2 years. I am thankful for the opportunities that Professor Lin has given me to participate in research and for being an incredible advisor. Professor Lin's insights and advice introduced me to and helped me refine my research methods, which have helped me improve as a researcher.

I would also like to thank Professor Gary Bishop for his advising. Professor Bishop was the first professor with whom I had worked with as a research assistant at UNC and I am grateful for the opportunities and motivation that he has provided to continue my research endeavors.

I would like to thank Professor Alterovitz, Lin, and Bishop for not only being incredible research advisors, but also for being great mentors in the subjects relating to academics, graduate school, and my career beyond UNC. In multiple instances, they allocated time out of their busy schedules to guide and advise with matters outside of research. I am forever grateful to have worked with advisors who care for my education during and beyond Carolina and for their invaluable advice and time.

I would also like to thank the students of the Computational Robotics Group, who have been incredibly helpful mentors over the years. First and foremost, I would like to thank and express my gratitude for Ph.D. student Luis Torres, without whom this thesis would not have been possible. Luis's help on this thesis has been so significant that I would list him as a second author of this thesis if I could. Luis has helped me in matters spanning technical writing, debugging bugs in my code, and making time to hear and give feedback to my crazy research ideas over the years. I am truly grateful to have had an experienced research mentor such as Luis, who was generous with his time to help me amidst his Ph.D. work. I would also like to thank Ph.D. student Chris Bowen, who was always receptive to hearing my research, giving me feedback, and helping me become a better researcher. I would also like to express my gratitude to the other members of the Computational Robotics Group: Alan, Jeff, and Wen (now at CMU), who have helped me become a better computer scientist and from whom I have learned a wealth of information.

I would also like to thank my family and friends, who have always been supportive of my ambitions and goals.

This research was funded by the National Institutes of Health (NIH) under awards R21EB017952 and R01EB017467, a Summer Undergraduate Research Fellowship from the Office for Undergraduate Research at the University of North Carolina at Chapel Hill, and the Dunlevie Honors Undergraduate Research Fund, administered by Honors Carolina.

TABLE OF CONTENTS

LIST OF FIGURES	viii
1 Introduction	1
2 Related Work	5
3 Problem Definition	7
3.1 Optimization Objectives	8
4 Methods	10
4.1 Computing Reachable Goal Percentage for a Single Design	10
4.2 Sampling a New Single Design	11
4.3 Finding an Optimal Single Design	12
4.4 Computing Reachable Goal Percentage for a Design Set	12
4.5 Sampling New Sets of Designs	13
4.6 Finding a Design Set of Minimal Size	14
5 Analysis	17
6 Results	21
6.1 Single Design Experiments	21
6.2 Set of Designs Experiments	24
6.2.1 Maximizing Reachability of a Design Set of Fixed Size	25
6.2.2 Minimal Design Set with Sufficient Goal Reachability	25
6.2.3 Benchmarking Variations on Algorithm	26
7 Conclusions	29
BIBLIOGRAPHY	30

LIST OF FIGURES

1.1	A simulated concentric tube robot operating under 4 different designs reaching clinical goal regions within the lung. In the figures above, the three colors aqua, orange, and pink correspond to the 3 different tubes that constitute the robot. The green sphere marks the position in the workspace that the robot can reach with its end effector without colliding with the anatomical obstacles, which include arteries (red), veins (blue), and bronchial tubes (light pink).	2
1.2	We show the points reachable in a human lung model by a concentric tube robot (deployed near the base of the primary bronchus) under two different designs in red and green. The figure shows that the two designs complement one another; the two designs can collectively reach a larger volume of the lung than can either of the designs alone. By generating sets of designs that can collectively reach the entire lung, physicians can perform a wider variety of surgical procedures with just one robot platform.	3
1.3	A flow diagram illustrating the envisioned clinical workflow and the stage at which our design optimization algorithms would be used. We note that the inputs to our algorithm are a medical image and a physician-specified goal region. By feeding this input to our design optimization algorithm, we enable our method to generate designs specific to the patient and application.	4
6.1	A screenshot of the experimental setup. In this image, we can see the concentric tube robot being deployed at the base of the primary bronchus with the objective of reaching physician-specified goal regions within the lung without colliding with vital anatomical structures. We note that the anatomical obstacles in the figure include arteries (red), veins (blue), and bronchioles and bronchi (light pink).	22
6.2	A visualization of the random goal region that was generated for the single design optimization experiment. The green voxels in the figure constitute the goal region, i.e. volume of points that the physician wants to reach with the robot's tip. In the figure, we see the concentric tube robot reaching a point within random goal region with its end effector (denoted by the green sphere)..	23

6.3 We compare the performance of our proposed algorithm (ASA+MP) with other design optimization algorithms for generating an optimal single design for the lung scenario, some of which are inspired by the design algorithm from prior work (Burgner et al., 2013). Moreover, we compare the performance of our algorithm with the method that we extend in this thesis: the algorithm of Torres et al. which uses the method of RRT of RRTs for design optimization (Torres et al., 2012). 24

6.4 We show the performance of Algorithm 3 over time in finding design sets of fixed sizes 1, 2, 4, and 6 that collectively maximize the reachable percentage of a human lung. In general, larger design sets can reach a greater percentages of the goal region. 26

6.5 We show the performance of Algorithm 4 over time in finding a design set of minimal size that can reach 95% of the human lung. This plot is averaged over 20 trials and we began the plot when all 20 trials had found their first design set capable of reaching 95% of the lung. 27

6.6 We compare the performance of our proposed algorithm (ASA+MP) with variants inspired by a design algorithm from prior work (Burgner et al., 2013). This prior method used the Nelder-Mead (NM) optimization algorithm, whereas we use a globally optimal algorithm called Adaptive Simulated Annealing (ASA). We also extended the prior work to consider obstacle avoidance in the entire deployment of the concentric tube robot by using motion planning. These extensions result in finding better sets of robot designs in less time. 28

Chapter 1

Introduction

Concentric tube robots are tentacle-like surgical robots that can potentially enable safer minimally invasive interventions to many sites in the human body, including the lungs, the skull base, and the heart (Gilbert et al., 2013). These robots are composed of nested Nitinol tubes that each are precurved, typically with a straight segment followed by a curved segment. To perform a task, the robot axially rotates and translates each tube relative to one another, causing the entire device's shape to change. Concentric tube robots act like shape-changing robotic needles that can curve around anatomical obstacles (e.g., bones, blood vessels, critical nerves, etc.) to reach surgical targets not easily accessed using traditional straight surgical instruments.

The curvilinear shapes achievable by concentric tube robots are highly dependent on the component tubes' physical specifications. The *design* of the concentric tubes, including the tubes' lengths and precurvatures, affects the robot's workspace and the space of the robot's attainable shapes. Consequently, the design of the concentric tubes determines the set of clinical targets that the robot can safely reach.

The fact that the robot's design has a profound impact on the clinical targets the robot can reach implies that a successful medical procedure is contingent upon using an appropriately-designed robot. In this thesis, we present a novel algorithm for optimizing the design of a single concentric tube robot with respect to the patient's anatomy and the physician-specified goal region. We thoroughly analyze the optimality property of our single design optimization algorithm and show that our algorithm is asymptotically optimal.

Even with the shape-changing capabilities of a concentric tube robot, due to kinematic constraints a single design is may not capable of reaching all targets in a physician-specified goal region. Fortunately, concentric tube robots can be built to facilitate simple swapping of tubes of varying physical specifications; selecting concentric tubes for a particular task

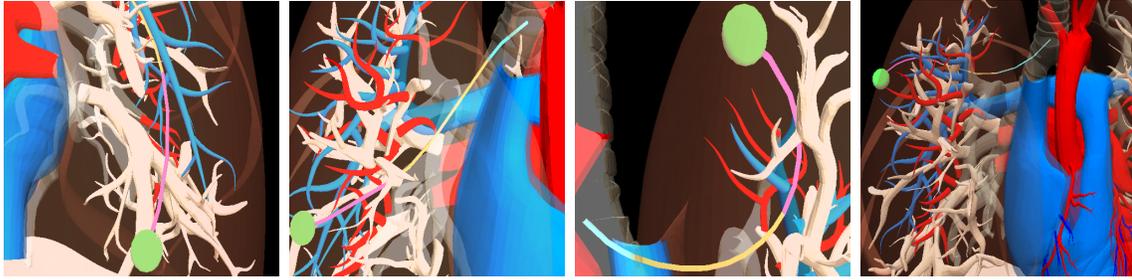


Figure 1.1: A simulated concentric tube robot operating under 4 different designs reaching clinical goal regions within the lung. In the figures above, the three colors aqua, orange, and pink correspond to the 3 different tubes that constitute the robot. The green sphere marks the position in the workspace that the robot can reach with its end effector without colliding with the anatomical obstacles, which include arteries (red), veins (blue), and bronchial tubes (light pink).

could maximize the robot’s efficacy during the procedure. For the sake of efficiency, it is desirable to generate and use the minimum number of designs required in order to conduct a successful medical procedure. To this end, we introduce a new algorithm to efficiently compute a minimal *set* of designs for a concentric tube robot, such that this set of designs can be sequentially swapped into the robot to access as much of the goal region as feasibly possible while avoiding anatomical obstacles.

The methods we propose could be used to create designs for classes of procedures or on a patient-specific basis. Prior to a procedure, physicians typically obtain a CT scan or MRI of the relevant anatomy, and we can use these volumetric images to segment (either manually or via automatic segmentation software) the goal region as well as anatomical obstacles that must be avoided. Fig. 1.3 illustrates the envisioned clinical workflow at highlights the juncture when our design optimization algorithm would be used. Unfortunately, the complex kinematics of concentric tube robots makes it difficult to assess if a given design of concentric tubes can safely access a given target while avoiding anatomical obstacles. As the device’s tip moves, the shape of the shaft of the device may change substantially, and this shape change must be considered to ensure obstacle avoidance. Torres et al. previously addressed the challenge of computing a single design to reach a finite set of points by using a sampling-based motion planning method that explicitly considered the shape of the entire device en route to a target point (Torres et al., 2012).

In this thesis, we build on the prior approach by Torres et al. by interleaving a search in the concentric tube robot’s design space (i.e., the lengths and precurvatures of the robot’s component tubes) with a motion planner in the robot’s configuration space (i.e., the rotations

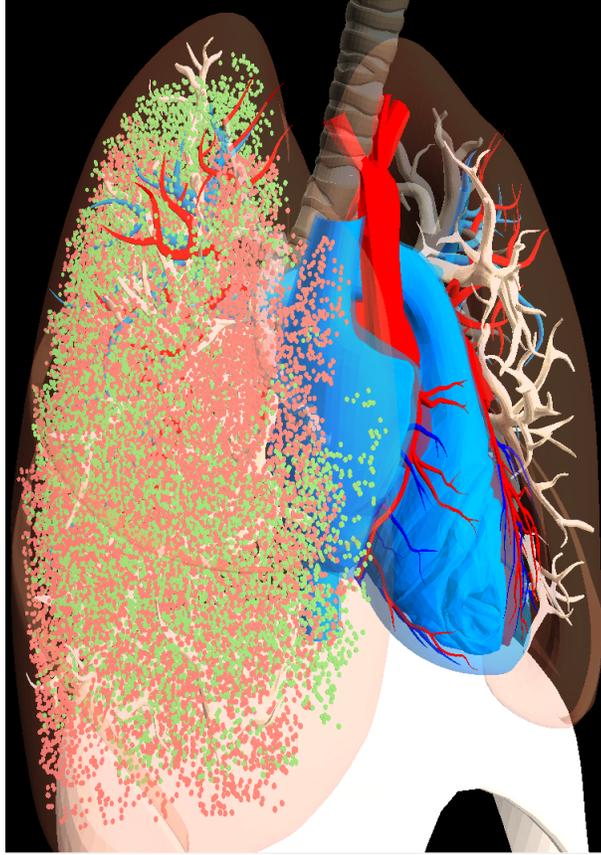


Figure 1.2: We show the points reachable in a human lung model by a concentric tube robot (deployed near the base of the primary bronchus) under two different designs in red and green. The figure shows that the two designs complement one another; the two designs can collectively reach a larger volume of the lung than can either of the designs alone. By generating sets of designs that can collectively reach the entire lung, physicians can perform a wider variety of surgical procedures with just one robot platform.

and translations of the robot's tubes). As an extension to the algorithm of Torres et al., our algorithms employ Adaptive Simulated Annealing (ASA) (Ingber, 1989) global optimization algorithm to accelerate the design space search. As was done by the algorithm introduced by Torres et al., our algorithms employ the Rapidly-exploring Random Trees (RRT) (LaValle, 2006) motion planning algorithm to quickly evaluate the goal region reachability of candidate designs. We demonstrate the effectiveness of our algorithms in generating appropriate designs and minimal sets of designs for concentric tubes on an application- and patient-specific basis using anatomy-inspired scenarios.

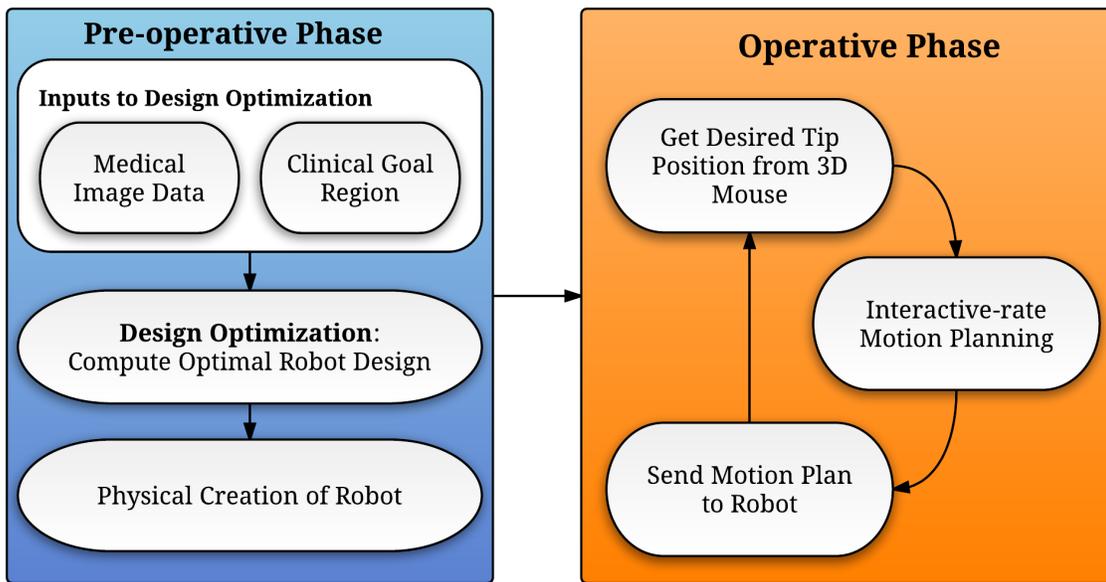


Figure 1.3: A flow diagram illustrating the envisioned clinical workflow and the stage at which our design optimization algorithms would be used. We note that the inputs to our algorithm are a medical image and a physician-specified goal region. By feeding this input to our design optimization algorithm, we enable our method to generate designs specific to the patient and application.

Chapter 2

Related Work

A number of approaches have previously been developed for the single design problem. Moreover, the problem of computing an optimal set of designs can be seen as a generalization of the single design problem.

Bergeles et al. proposed a powerful optimization framework for generating robot tube designs that can reach sets of points subject to anatomical constraints, and then applied this method to brain and heart surgery scenarios (Bergeles et al., 2015). They achieve computational tractability by (1) reducing the motion planning problem to finding individual configurations that can reach each specified task point, and (2) using a simpler and faster kinematic model for the general optimization, and then refining the solution using a more accurate (but slower) kinematic model. Although this method works well for a variety of cases, the assumptions that enable computational tractability can sometimes yield suboptimal solutions (Torres et al., 2012). This can happen because the method does not explicitly consider the entire robot deployment to the surgical task site (Torres et al., 2012).

Ha et al. presented a method for designing concentric tube robots while maximizing device stability (Ha et al., 2014). This method complements this paper's approach to computing minimal sets of concentric tube robot designs.

Burgner et al. addressed the problem of finding a concentric tube robot design that maximized the reachable region of points in the sella of the human skull, subject to physical constraints imposed by the bones in the skull (Burgner et al., 2013). They achieved this by performing a nonlinear optimization over the design space of the robot; they quantify how much of the sella is reachable under a given design by computing the forward kinematics over a grid on the robot's configuration space. Their approach is well-suited for the neurosurgical scenario in question, but can be subject to the same suboptimal solutions as the work by Bergeles et al. due to not considering the full robot deployment to the surgical task site.

We take an alternative approach that explicitly considers the entire robot deployment to the surgical task site. Torres et al. first explored this approach by using sampling-based motion planning (Torres et al., 2012). In this paper we extend the work of Torres et al. work to (i) find an optimal single design with respect to volume-based objectives and (ii) generate a minimal set of robotic designs that can enable a wide variety of surgical tasks. We also use an improved search strategy based on Adaptive Simulated Annealing (ASA) (Ingber, 1989) that enables faster convergence to higher quality solutions. Moreover, we present analysis outlining a proof asserting that the single design optimization algorithm is asymptotically optimal.

Designing a concentric tube robot in advance to perform a particular task requires accurate kinematic modeling. Kinematic modeling of concentric tube robots has rapidly progressed in accuracy and sophistication, from bending models (Sears and Dupont, 2006), to torsionally compliant models (Dupont et al., 2009; Rucker and Webster III, 2009), externally loaded models (Rucker et al., 2010; Lock et al., 2010), and friction models (Lock and Dupont, 2011). In this work we used a mechanics-based model developed by Rucker et al. (Rucker, 2011).

Our approach depends on the ability to determine the positions in the anatomy that are safely reachable by a concentric tube robot. Burgner-Kahrs et al. developed a novel method to characterize the workspace of concentric tube robots (Burgner-Kahrs et al., 2014); we instead take a motion planning approach in order to find points that are reachable by robot motions that avoid contact with anatomical obstacles. Prior work in motion planning for concentric tube robots include fast planners using simplified kinematic models (Lyons et al., 2009; Trovato and Popovic, 2009), non-interactive planning for mechanics-based kinematic models (Torres and Alterovitz, 2011), and fast planning for mechanics-based models using precomputation (Torres et al., 2014). In this paper we prioritize an accurate approximation of the robot’s reachability, so we use an accurate kinematic model combined with the classical Rapidly-exploring Random Tree motion planning algorithm (LaValle, 2006).

The problem of optimizing robotic design has been addressed in previous work for serial manipulators. Prior work has used genetic algorithms to optimize the structure of manipulators under various metrics (Katragadda, 1997; Leger, 1999; Chocron, 2008; Salle et al., 2004). Other approaches to optimal manipulator design have used interval analysis (Merlet, 2005), geometric methods (Vijaykumar et al., 1986), and grid-based methods (Park et al., 2003). We explore an alternative approach that can handle the complex kinematics of concentric tube robots.

Chapter 3

Problem Definition

A concentric tube robot design \mathbf{d} is the set of physical parameters of the robot's component tubes that are selected and fixed before performing a surgical task and cannot be changed during the performance of the procedure. Specifically, we describe each tube's design with the following 3 parameters.

- L_i^s : length of tube i 's straight section
- L_i^c : length of tube i 's pre-curved section
- κ_i : curvature of tube i 's pre-curved section

Therefore, a concentric tube robot composed of n tubes has a *design space* D with $3n$ parameters, i.e., $D \subset \mathbb{R}^{3n}$.

During operation of a concentric tube robot, each tube can be independently axially rotated and translated, meaning that an n -tube robot's configuration is a $2n$ -dimensional vector \mathbf{q} . The configuration space of a concentric tube robot is $Q \subset (S^1)^n \times \mathbb{R}^n$.

We represent the shape of a concentric tube robot during operation as a 3D space curve that depends on (i) the robot's design and \mathbf{d} (ii) the robot's configuration \mathbf{q} . We therefore denote the robot's shape using a function $\text{Shape}(\mathbf{d}, \mathbf{q}) : D \times Q \rightarrow ([0, 1] \rightarrow \mathbb{R}^3)$ that assigns a 3D space curve representing the robot's shape under design \mathbf{d} at configuration \mathbf{q} . The curve itself is parametrized by $s \in [0, 1]$, i.e. $\text{Shape}(\mathbf{d}, \mathbf{q})(s)$ generates points along the shaft of the robot. The positions of the robot's insertion point and end-effector correspond to $s = 0$ and $s = 1$, respectively. We compute Shape using an accurate mechanics-based model of concentric tube interactions (Rucker, 2011).

Safe operation of a concentric tube robot requires that we avoid collisions between the robot's shaft and anatomical obstacles such as bones, blood vessels, and sensitive tissue. We

define the anatomical obstacles $O \subset \mathbb{R}^3$ as all 3D points in space that should never intersect with the robot’s shape Shape . The collision-free subset of robot configurations is Q_{free} . We note that, because the robot shape Shape varies with the robot’s design, the collision-free subset of configurations also depends on the robot’s current design, so we denote it as $Q_{\text{free}}^{\mathbf{d}}$. We can determine O via manual or automatic segmentation on the patient’s preoperative medical imaging (Johnson et al., 2013).

3.1 Optimization Objectives

We wish to find a single design or sets of concentric tube robot designs that can access surgical targets by following collision-free paths. We consider a path $\Pi = (\mathbf{q}_1, \dots, \mathbf{q}_l)$ to be collision-free if the continuous paths to each subsequent configuration \mathbf{q}_i are all collision-free (i.e., free of intersection with O).

The goal of this thesis is to find concentric tube robot designs that allow for collision-free paths to as many points as possible in a *goal region* $G \subset \mathbb{R}^3$, which is identified in medical images by physicians in a manner similar to obstacles. We emphasize that this goal region is different from typical motion planning problems since we want to reach as many points as possible in the goal region rather than finding a single collision-free path to any point in the goal region.

We denote the set of points that a concentric tube robot can reach by collision-free paths as $W(\mathbf{d}) \subset \mathbb{R}^3$ (i.e., the *feasible workspace*). We can therefore quantify the quality of a given design as the percentage of G that lies in the robot’s feasible workspace W . For computational feasibility, we discretize the goal region G into a countable and finite set of voxels V (i.e., cells in a 3D grid). We compute the reachable goal percentage r of a given design \mathbf{d} as follows

$$r(\mathbf{d}) = \frac{|\text{VoxelsReached}(\mathbf{d})|}{|V|}. \quad (3.1)$$

Similarly, the reachable goal percentage r of a given design set S is evaluated by

$$r(S) = \frac{|\bigcup_{\mathbf{d} \in S} \text{VoxelsReached}(\mathbf{d})|}{|V|}. \quad (3.2)$$

When computing VoxelsReached , we emphasize that we must consider the entire sequence of motions executed to reach a goal voxel. With these definitions in place, we consider the following two problems:

Problem 1: Optimal Single Design Generate the single design \mathbf{d}^* that maximizes the reachable goal percentage, i.e.,

$$\mathbf{d}^* = \operatorname{argmax}_{\mathbf{d} \in D} r(\mathbf{d}). \quad (3.3)$$

Problem 2: Minimal Design Set Find a set S^* of robot designs that is *minimal* (in cardinality) but with a reachable goal percentage greater than a physician-specified threshold $r_{\text{threshold}}$:

$$S^* = \operatorname{argmin}_{S \in 2^D} |S|, \quad \text{s.t.} \quad r(S^*) > r_{\text{threshold}}, \quad (3.4)$$

where 2^D is the set of all possible sets of designs.

Chapter 4

Methods

Our algorithms for generating \mathbf{d}^* and S^* in Eq. 3.3 and 3.4 respectively interleave a guided sampling-based search in the robot’s design space with a sampling-based motion planner in the robot’s configuration space. For computation of \mathbf{d}^* , we use a global optimization algorithm called Adaptive Simulated Annealing (ASA) (Ingber, 1989). For motion planning in the configuration space, we use the Rapidly-exploring Random Tree (RRT) (LaValle, 2006) algorithm. We use ASA to sample a single design, and then we use RRT to evaluate this design’s reachable goal percentage. We iterate on this process, incrementally finding designs with higher reachable goal percentages.

The algorithm for generating S^* can be seen as an extension of our algorithm for finding \mathbf{d}^* . In this case, we use ASA to sample *sets* of m designs and then we employ RRT to evaluate this set’s reachable goal percentage. We iterate on this process until we find a set of designs that can collectively reach a sufficient percentage of the goal region ($r_{\text{threshold}}$). If a design set of size m is found, then we try to find a set of smaller size by repeating this process for a group of designs of size $m - 1$ and so on until a minimal set of designs reaching a satisfactory percentage of the goal region is found.

4.1 Computing Reachable Goal Percentage for a Single Design

According to Eq. 3.3 in order to evaluate the reachable goal percentage r of a single design \mathbf{d} , we need to compute $\text{VoxelsReached}(\mathbf{d})$. Computing the output of $\text{VoxelsReached}(\mathbf{d})$, i.e. checking whether a given voxel can be reached by a collision-free path is equivalent to solving the motion planning problem, which is known to be PSPACE-hard (Reif, 1979).

This implies that, in order to generate solutions in a feasible amount of time, we must accept approximate solutions. We therefore use a probabilistic, sampling-based motion planning algorithm called RRT that can quickly compute an approximation of the robot’s feasible workspace (LaValle, 2006).

RRT incrementally builds a tree of robot configurations that can be reachable by collision-free paths from a given start configuration under a given design \mathbf{d} . After a given number t of iterations of RRT, we iterate over each configuration \mathbf{q} in the tree to check which goal voxels can be reached from these configurations.

In this way we compute an approximation of $\text{VoxelsReached}(\mathbf{d})$ and then use Eq. 3.1 to compute the design’s approximate reachable goal percentage $\hat{r}_t(\mathbf{d})$. We use the t in $\hat{r}_t(\mathbf{d})$ to denote that this approximation was generated using t iterations of the RRT algorithm. We note that the nature of RRT’s feasible workspace approximation is such that we never overestimate the design set’s true reachable goal percentage, i.e., $\hat{r}_t(\mathbf{d}) \leq r(\mathbf{d})$. RRT also provides *probabilistic completeness*, a useful property in which the longer we execute the RRT algorithm, the more likely it is to find a collision-free path to a given target (if a feasible path exists). This implies that, as we increase the iterations t of RRT, the probability of our approximation $\hat{r}_t(\mathbf{d})$ being equal to the true $r(\mathbf{d})$ approaches 100%.

4.2 Sampling a New Single Design

In the previous section we described how we compute the reachable goal percentage of a single design. In this section, we discuss the strategy we use to generate new single designs that are evaluated by the process outlined in Sec. 4.1.

We note that the space of single designs is D , i.e. the design space. In accordance with traditional optimization terminology, we will refer to members of the set D as *states*. For an n -tube concentric tube robot, this problem is a $3n$ -dimensional search for an optimal design (i.e., a state), \mathbf{d} , with highest reachable goal percentage. Due to the high dimensionality of the search space, we opt for a stochastic approach based on the Adaptive Simulated Annealing (ASA) algorithm. We use ASA because because it provides global optimality, which guarantees that we will not get stuck indefinitely in local optima during our search for better designs. Moreover, as shown in Sec. 5, ASA exhibits favorable algorithmic properties that allow us to prove asymptotic optimality of our single design optimization algorithm.

ASA is always centered on a “current” state $\mathbf{d}_{\text{current}}$ in the search space. At the beginning, ASA tends to sample states far away from $\mathbf{d}_{\text{current}}$ in order to adequately explore the space. As ASA progresses, it tends to sample states nearer and nearer to $\mathbf{d}_{\text{current}}$ in order to make

local refinements. ASA controls this *sampling variance* using a *temperature* parameter T that decreases with each iteration of ASA. Whenever ASA samples a state $\mathbf{d}_{\text{sample}}$ with a lower cost than that of $\mathbf{d}_{\text{current}}$, ASA updates $\mathbf{d}_{\text{current}}$ to be equal to this new sample $\mathbf{d}_{\text{sample}}$. Additionally, if the cost of the new sample is higher than that of the current state, ASA might still update to the new sample with an *acceptance probability* that decreases over time (also controlled by the temperature T). This potential to take steps of increasing costs allows ASA to escape local minima in state space.

In our method, a state $\mathbf{d} \in D$ has a low cost if it has a high reachable goal percentage, which we approximate with $\hat{r}_t(\mathbf{d})$ as described in Sec. 4.1. Evaluation of $\hat{r}_t(\mathbf{d})$ requires that we specify t , i.e., the number of iterations of RRT to use for the approximation. We cannot know in advance how many iterations of RRT it will take to compute an adequate approximation of a design’s goal reachability, so we set this number of iterations t to an initial value t_{start} and increase it by t_{increase} after every design we consider. This enables us to more quickly (but more coarsely) and evaluate many designs in the initial phases of our algorithm, and then evaluate designs at a slower rate with higher accuracy as the algorithm progresses. This behavior is analogous to ASA’s decreases in sampling variance and acceptance probability over time. This key property of our algorithm is fundamental in establishing asymptotic optimality for the single design optimization algorithm.

4.3 Finding an Optimal Single Design

In the previous sections, we described how a single design is evaluated (Sec. 4.1) and sampled (Sec. 4.2). In this section we present Alg. 1, our asymptotically optimal single design optimization that interleaves the search and evaluation subroutines. At each iteration of our algorithm, our algorithm samples a design using ASA’s sampling procedure and evaluates it using an RRT. This process of sampling and evaluation is repeated until the algorithm is terminated, at which point our algorithm outputs the best design found thus far by our algorithm.

4.4 Computing Reachable Goal Percentage for a Design Set

Similar to the case of evaluating the reachable goal percentage of a single design outlined in Sec. 4.1, according to Eq. 3.4 in order to evaluate the reachable goal percentage r of a set of designs S , we need to compute $\text{VoxelsReached}(\mathbf{d})$ for each design \mathbf{d} in the set S . This com-

Algorithm 1 Find a single design with maximum reachable goal percentage

Output: \mathbf{d}^* : a concentric tube robot design with maximum reachable goal percentage

```
1:  $t \leftarrow t_{\text{start}}$ ;
2:  $T \leftarrow T_{\text{initial}}$ ;
3:  $r_{\text{current}} \leftarrow 0$ ;
4:  $\mathbf{d}_{\text{current}} \leftarrow$  random initial design;
5: while allotted time remains do
6:    $\mathbf{d}' \leftarrow \text{ASA\_SampleSingleDesign}(\mathbf{d}_{\text{current}}, T)$ 
7:    $V_{\mathbf{d}'} \leftarrow \text{executeRRT}(\mathbf{d}', t)$ 
8:    $r' \leftarrow |V_{\mathbf{d}'}|/|V|$ 
9:   if  $r' > r_{\text{current}}$  then
10:      $S_{\text{current}} \leftarrow S'$ ;
11:      $r_{\text{current}} \leftarrow r'$ ;
12:   else
13:     if  $\text{ASA\_maybeAccept}(r', r_{\text{current}}, T)$  then
14:        $S_{\text{current}} \leftarrow S'$ ;
15:        $r_{\text{current}} \leftarrow r'$ ;
16:    $t \leftarrow t + t_{\text{increase}}$ ;
17:    $T \leftarrow \text{ASA\_updateTemperature}(T)$ ;
18: return design  $\mathbf{d}^*$  found with highest goal reachability;
```

putation can be done by executing the process mentioned in Sec. 4.1 for each design $\mathbf{d} \in S$. We compute an approximation of $\text{VoxelsReached}(\mathbf{d})$ for each $\mathbf{d} \in S$ in parallel for a considerable computational speedup and then use Eq. 3.2 to compute the design set’s approximate reachable goal percentage $\hat{r}_t(S)$.

4.5 Sampling New Sets of Designs

In the previous section we described how we compute the goal reachability of a design set; in this section, we discuss the strategy we use to generate new sets of designs. We will discuss how to find a set of designs of minimal size in Sec. 4.6; for this section, we will assume that the required size of the design set is fixed, i.e., $|S^*| = m$.

The procedure for sampling sets of designs is very similar to the process of sampling a single design which was outlined in Sec. 4.2. Nevertheless, in the case of a set of designs, the space of possible sets of designs is D^m and states in this context refer to design sets, rather than single designs. In other words, we treat members of the set D^m as *states* in the context

of the optimization objective of Problem 2. For an n -tube concentric tube robot, this problem is a $3nm$ -dimensional search for a design set (i.e., a state) with sufficient goal reachability. With the exception of the expanded state space and the consideration of design sets as states, the ASA algorithm’s behavior for sampling new states is the same as that for sampling single designs, described in Sec. 4.2.

As mentioned in Sec. 4.4, for efficiency we compute $\hat{r}_t(S)$ by parallelizing the computations of $\text{VoxelsReached}(\mathbf{d})$ for all $\mathbf{d} \in S$ across multiple processor cores. However, we often have more processor cores than designs in a design set, i.e., $c > m$ for c processor cores and m designs per design set. This leaves $c - m$ cores that are free for additional computation. In order to make use of all our cores, at each iteration of ASA we actually sample a design set S' of size c and evaluate each design’s reachable goal voxels. We then iterate over all $\binom{c}{m}$ subsets of S' of size m to find the set of designs that collectively yield the highest reachable goal percentage $\hat{r}_t(S)$. This subset iteration step is completely dominated in computation time by the evaluation of each design’s reachable goal percentage, so this method effectively enables us to sample design sets of higher quality with no extra computation time due to parallelization.

4.6 Finding a Design Set of Minimal Size

In the previous section, we described how we find a set of designs of fixed size that collectively maximize the reachable percentage of the goal. We use this method as a subroutine to solve our full problem of finding a set of designs of minimal size with a reachable goal percentage greater than a specified threshold $r_{\text{threshold}}$ (shown in Alg. 4).

We begin by invoking the fixed size algorithm (Alg. 3) with a user-specified maximum set size m_{max} . Once a design set of size m_{max} has been found that can reach a percentage of the goal greater than $r_{\text{threshold}}$, we invoke Alg. 3 using a design set size of $m_{\text{max}} - 1$. We iterate this process until we run out of the time allotted for this task.

Algorithm 2 Sample and evaluate a design set

Input:

S_{current} : Current design set of size m
 m : required design set size
 c : number of available processing cores
 t : number of RRT iterations to execute
 T : ASA's current annealing temperature

Output:

S_{new} : new set of robot designs of size m
 r_{new} : reachable goal percentage of S_{new}

```
1:  $S \leftarrow \text{ASA\_SampleDesignSet}(S_{\text{current}}, c, t, T)$ ;  
2:  $\text{designToVoxelsMap} = \emptyset$ ;  
3: for  $\mathbf{d}_i \in S$  (in parallel) do  
4:    $\text{designToVoxelsMap}[\mathbf{d}_i] \leftarrow \text{executeRRT}(\mathbf{d}_i, t)$ ;  
5:  $\text{candidateSets} \leftarrow$  subsets of size  $m$  of  $S$ ;  
6:  $r_{\text{new}} \leftarrow 0$ ;  
7:  $S_{\text{new}} \leftarrow \emptyset$ ;  
8: for each  $S' \in \text{candidateSets}$  do  
9:    $r \leftarrow |\bigcup_{\mathbf{d} \in S'} \text{designToVoxelsMap}[\mathbf{d}]|$ ;  
10:  if  $r > r_{\text{new}}$  then  
11:     $r_{\text{new}} \leftarrow r$ ;  
12:     $S_{\text{new}} \leftarrow S'$ ;  
13: return  $S_{\text{new}}, r_{\text{new}}$ ;
```

Algorithm 3 Find a design set of fixed size with maximum reachable goal percentage

Input:

m : number of designs in the design set S
 c : number of available processing cores
 $r_{\text{threshold}}$: stopping point for our design search

Output:

S^* : a set of m concentric tube robot designs that together maximize the reachable goal percentage

```
1:  $t \leftarrow t_{\text{start}}$ ;  
2:  $T \leftarrow T_{\text{initial}}$ ;  
3:  $r_{\text{current}} \leftarrow 0$ ;  
4:  $S_{\text{current}} \leftarrow$  arbitrary random set of designs;  
5: while  $r_{\text{current}} < r_{\text{threshold}}$  do  
6:    $S', r' \leftarrow$  Algorithm2( $S_{\text{current}}, m, c, t, T$ )  
7:   if  $r' > r_{\text{current}}$  then  
8:      $S_{\text{current}} \leftarrow S'$ ;  
9:      $r_{\text{current}} \leftarrow r'$ ;  
10:  else  
11:    if ASA_maybeAccept( $r', r_{\text{current}}, T$ ) then  
12:       $S_{\text{current}} \leftarrow S'$ ;  
13:       $r_{\text{current}} \leftarrow r'$ ;  
14:   $t \leftarrow t + t_{\text{increase}}$ ;  
15:   $T \leftarrow$  ASA_updateTemperature( $T$ );  
16: return design set  $S^*$  found with highest goal reachability;
```

Algorithm 4 Find a minimal design set that reaches a sufficient percentage of the goal

Input:

m_{max} : maximum number of designs in the design set
 $r_{\text{threshold}}$: desired reachable goal percentage

Output:

S^* : a minimal set of designs with goal reachability greater than $r_{\text{threshold}}$

```
1:  $m \leftarrow m_{\text{max}}$ ;  
2: while allotted time remains and  $m > 0$  do  
3:    $S^* \leftarrow$  Algorithm3( $m, c, r_{\text{threshold}}$ );  
4:    $m \leftarrow m - 1$ ;  
5: return  $S^*$ ;
```

Chapter 5

Analysis

In this section we consider the quality of the designs generated by our single design optimization algorithm (Alg. 1) as the number of optimization iterations tends to infinity. We outline a proof that the solution generated by our algorithm converges in probability to the globally optimal solution. We remark that detailed proofs of the Lemmas and Theorems are being formalized and will be published in future work. For the outline of this proof, we will utilize the following preliminaries and assumptions.

Assumption 1 (Voxels as Open Sets). *Each voxel $v \in V$ is as an open set, i.e.*

$$\forall v \in V, \forall x \in v, \exists \epsilon \in \mathbb{R}_+ B_\epsilon(x) \subseteq v$$

Recall that $\text{Shape}(\mathbf{d}, \mathbf{q}) : D \times Q \rightarrow ([0, 1] \rightarrow \mathbb{R}^3)$ is a function that assigns a 3D space curve representing the robot's shape under design \mathbf{d} at configuration \mathbf{q} . In this representation, $\text{Shape}(\mathbf{d}, \mathbf{q})(1)$ denotes the position of the robot's end effector. Let $\text{Tip} : D \times Q \rightarrow \mathbb{R}^3$ be the function that outputs the tip position of the robot under design \mathbf{d} at configuration \mathbf{q} . The following assumption on the *Shape* function ensures that robots under similar designs have similar shapes at the same configuration.

Assumption 2 (Lipschitz Continuity of the Shape Function). *The Shape function is Lipschitz continuous: for all $\mathbf{d}_1, \mathbf{d}_2 \in D$ and $\mathbf{q} \in Q^{\mathbf{d}_1} \cap Q^{\mathbf{d}_2}$*

$$\|\text{Shape}(\mathbf{d}_1, \mathbf{q}) - \text{Shape}(\mathbf{d}_2, \mathbf{q})\|_\infty \leq L \|\mathbf{d}_1 - \mathbf{d}_2\|_\infty$$

for some constant $L \in \mathbb{R}_+$.

Note that we utilize the infinity (Chebyshev) norm in Assumption 2.

Assumption 3 (Obstacle Spacing). *Let $X_{free} \subset \mathbb{R}^3$ denote the obstacle-free space. There exists a constant $\delta \in \mathbb{R}_+$ such that for any point $x \in X_{free}$, there exists $x' \in X_{free}$ such that $B_\delta(x') \subset X_{free}$ and $x \in B_\delta(x')$.*

Assumption 3 is the same assumption that was made by Karaman and Frazzoli (Karaman and Frazzoli, 2010).

The outline of the proof is as follows. First, we show that the optimal set of designs D^* has non-zero measure, i.e. $\mu(D^*) > 0$ given Assumptions 1, 2, and 3 hold. Subsequently, we show that the ASA algorithm will sample designs infinitely many times from D^* , which will lead to exact evaluations of $R(\mathbf{d})$ for designs sampled from D^* , i.e., $\mathbf{d} \in D^*$ as the number of iterations $n \rightarrow \infty$.

Note that in order for the Lipschitz condition of Assumption 2 to be utilized, there must exist a set of configurations that the robot can achieve under both designs $\mathbf{d}_1, \mathbf{d}_2 \in D$, i.e., $Q^{\mathbf{d}_1} \cap Q^{\mathbf{d}_2}$, since due to mechanical constraints, some configurations may be unachievable under certain designs. In the following lemma, we establish, for all $\mathbf{d} \in D$, the existence of an open set of designs, D_{near} such that $\mu(D_{near}) \in \mathbb{R}_+$ and $\forall \mathbf{d}' \in D_{near}, Q^{\mathbf{d}} \subseteq Q^{\mathbf{d}'}$.

Lemma 4 (Configuration Space of Nearby Designs). *For any design $\mathbf{d} \in D$ there exists a set of designs $D_{near} \subseteq B_\varepsilon(\mathbf{d})$, with $\varepsilon \in \mathbb{R}_+$, such that each design $\mathbf{d}' \in D_{near}$ can achieve the configurations that \mathbf{d} can, i.e.*

$$D_{near} = \{\mathbf{d}' \in B_\varepsilon(\mathbf{d}) \mid Q^{\mathbf{d}} \subseteq Q^{\mathbf{d}'}\}$$

such that $\mu(D_{near}) \in \mathbb{R}_+$.

Lemma 5 (Proximity of Tip Positions). *For all $\mathbf{d}_1, \mathbf{d}_2 \in D$ and $\mathbf{q} \in Q^{\mathbf{d}_1} \cap Q^{\mathbf{d}_2}$, if $\|Shape(\mathbf{d}_1, \mathbf{q}) - Shape(\mathbf{d}_2, \mathbf{q})\|_\infty \leq \varepsilon$, then $\|Tip(\mathbf{d}_1, \mathbf{q}) - Tip(\mathbf{d}_2, \mathbf{q})\| \leq \varepsilon$, for $\varepsilon \in \mathbb{R}_+$.*

For a design $\mathbf{d} \in D$, let $\Sigma^{\mathbf{d}}(\mathbf{p})$ denote the set of feasible paths that the robot under design \mathbf{d} can undertake to reach the goal $\mathbf{p} \in \mathbb{R}^3$. The following lemma establishes the existence of an open set of designs around $\mathbf{d} \in D$, D_{near} , with positive measure such that the robot under a design $\mathbf{d} \in D_{near}$ can reach arbitrarily "close" points to those reachable under design \mathbf{d} .

Lemma 6 (Existence of Designs Capable of Reaching Close Goal Points). *Consider any $\mathbf{d} \in D$ that can reach a goal point $\mathbf{p} \in \mathbb{R}^3$ with its end effector by following a sequence of collision-free configurations ψ . Then, $\forall \xi \in \mathbb{R}_+$ satisfying $\xi \leq \delta$ there exists a non-empty, open subset of D , D_{near} with $\mu(D_{near}) \in \mathbb{R}_+$, such that the robot under any design in D_{near}*

can reach a goal point $\mathbf{p}' \in \mathbb{R}^3$ satisfying $\|\mathbf{p} - \mathbf{p}'\|_2 \leq \xi$, with its end effector by following a collision free path, i.e.,

$$D_{near} = \{\mathbf{d}' \in D \mid \Sigma^{\mathbf{d}'}(\mathbf{p}') \neq \emptyset \wedge \|\mathbf{p} - \mathbf{p}'\|_2 \leq \xi\}$$

where δ is the obstacle-spacing constant as defined in Assumption 3.

Recall that the function $\text{VoxelsReached}(\mathbf{d})$ computes the set of voxels reached by the robot's end effector under design \mathbf{d} . Note that there exists a voxel $v \in \text{VoxelsReached}(\mathbf{d})$ if and only if there exists a point $\mathbf{p} \in v$ that the robot's end effector can reach under design \mathbf{d} by following a collision-free path. For all $v \in \text{VoxelsReached}(\mathbf{d})$, let $\mathbf{p}_v^{\mathbf{d}} \in v$, denote the reached point in voxel v under design \mathbf{d} . Moreover, let the function $\text{SupRadius}(v, \mathbf{p}) : V \times \mathbb{R}^3 \rightarrow \mathbb{R}_+$ compute the least upper bound of the radius of an open ball centered at \mathbf{p} that is entirely within voxel v .

Lemma 7 (Existence of an Open Set of Solutions). *If a design $\mathbf{d} \in D$ can reach a set of voxels $V' \subseteq V$, then there exists a non-empty, open subset of designs $D_{near} \subseteq D$ such that $\forall \mathbf{d}' \in D_{near}, \text{VoxelsReached}(\mathbf{d}') = V'$ satisfying $\mu(D_{near}) \in \mathbb{R}_+$.*

Theorem 8 (Positive Measure of the Set of Optimal Solutions). *Let Assumptions 1, 2, and 3 hold. The set of optimal solutions, D^* , to Problem 1 as defined in Section 3.1 has positive measure, i.e. $\mu(D^*) > 0$.*

We now show that our algorithm's approximation of $\hat{r}(\mathbf{d})$ almost-surely approaches the true value, $r(\mathbf{d})$, as the number of optimization iterations approaches infinity. We consider design optimization as a mathematical optimization problem by treating the function R as our objective function that we seek to *maximize*. Since $r(\mathbf{d})$ cannot be exactly computed within a practical amount of time, we generate an iterative approximation in the form of the sequence $\{\hat{r}_n^{\mathbf{d}}\}_{n \in \mathbb{N}}$.

We note that randomness may arise in our approximation due to the probabilistic nature of RRT which we use to generate our approximations. To this end, we let Ω be the sample space of outcomes and redefine our iterative approximation as a sequence of functions mapping outcomes to real values. More precisely, we let $\hat{r}_n^{\mathbf{d}}(\omega)$ denote the approximation generated by a specific outcome $\omega \in \Omega$ at iteration $n \in \mathbb{N}$. Moreover, we let $\hat{V}_n^{\mathbf{d}}(\omega)$ denote our algorithm's approximation of the voxels reached by design \mathbf{d} by a specific outcome $\omega \in \Omega$ at iteration $n \in \mathbb{N}$.

Lemma 9. *The approximation of $r(\mathbf{d})$ almost-surely approaches the true value as $n \rightarrow \infty$, $\forall \mathbf{d} \in D$, i.e.,*

$$\forall \omega \in \Omega, \forall \mathbf{d} \in D, \Pr \left(\lim_{n \rightarrow \infty} \hat{r}_n^{\mathbf{d}}(\omega) = r(\mathbf{d}) \right) = 1$$

Lemma 10 (Approximation Lower Bound). *Our approximation of $r(\mathbf{d})$ is always a lower bound*

$$\forall n \in \mathbb{N}, \forall \mathbf{d} \in D, \forall \omega \in \Omega, \hat{r}_n^{\mathbf{d}}(\omega) \leq r(\mathbf{d})$$

.

To prove asymptotic optimality of our design optimization algorithm, we will need the following result about Adaptive Simulated Annealing algorithm (ASA).

Lemma 11 (Frequency of Sampling from a Design Set). *For any set of designs $D' \subseteq D$ with non-zero measure, i.e. $\mu(D') > 0$, elements from D' will be sampled infinitely many times by ASA as the number of optimization iterations $n \rightarrow \infty$.*

For the purposes of the asymptotic optimality proof, let $\mathbf{d}^* \in D^*$ denote a globally optimal design, i.e. a design that solves Problem 3.1 optimally and r^* as the optimal reachable goal percentage, i.e., $r^* = r(\mathbf{d}^*)$.

Theorem 12 (Asymptotic Optimality of Design Optimization). *Given that Assumptions 1, 2, and 3 hold, the design found by Alg. 1 converges in probability to a globally optimal design $\mathbf{d}^* \in D^*$, where $r^* = r(\mathbf{d}^*)$ is the optimal reachability value, as the number of iterations $n \rightarrow \infty$.*

Chapter 6

Results

We evaluate our design optimization algorithms in a simulated scenario based on lung anatomy. In this scenario, the concentric tube robot is deployed near the base of the primary bronchus of the right human lung using a rigid bronchoscope with the objective of reaching regions within the lung while avoiding anatomical obstacles, e.g., blood vessels and smaller bronchial tubes. All experiments were conducted on a PC with two 2.40 GHz Intel Xeon E5620 processors (8 cores total) and 12 GB of RAM.

6.1 Single Design Experiments

In this experiment, we consider finding a solution to Problem 1 as described in Section 3.1. Namely, we seek to find a single robot design, \mathbf{d}^* , that maximizes the reachable goal percentage for a specified goal region G . To simulate a clinical target region that a physician may want to reach with the robot, e.g. a lung nodule for biopsy used for early-stage lung cancer diagnosis, we generated a random contiguous goal region within the right human lung consisting of 8 voxels, each voxel with a side length of $v_s = 5$ mm. Goal voxels that were in collision with anatomical obstacles were pruned before the execution of trials. Fig. 6.1 shows a screenshot of the random goal region used for the experiments.

We compared our single design optimization algorithm, Alg. 1, with four other design optimization algorithms, three of which borrow some elements of the design optimization algorithm presented by Burgner et al. (Burgner et al., 2013), which we note was developed for different anatomical scenarios.

- *NM + G*: We use the Nelder-Mead optimization algorithm instead of ASA for generating new designs to consider. To evaluate the reachable goal percentage of a design, we do not use motion planning; we instead consider a goal voxel reachable if there exists

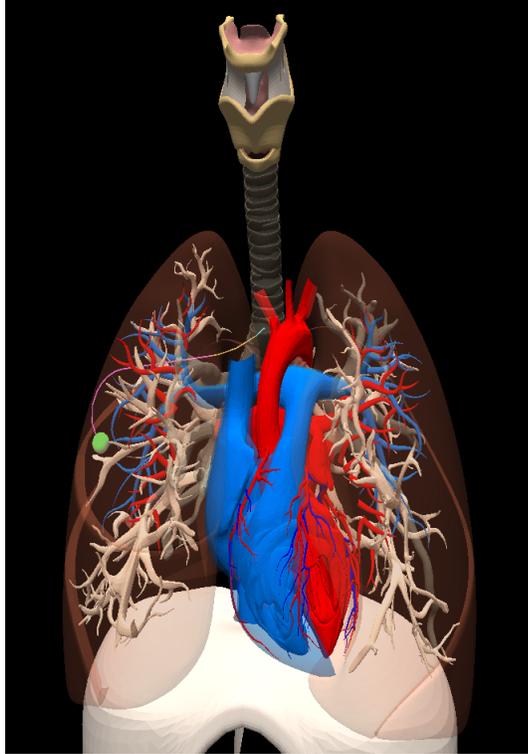


Figure 6.1: A screenshot of the experimental setup. In this image, we can see the concentric tube robot being deployed at the base of the primary bronchus with the objective of reaching physician-specified goal regions within the lung without colliding with vital anatomical structures. We note that the anatomical obstacles in the figure include arteries (red), veins (blue), and bronchioles and bronchi (light pink).

a single collision-free robot configuration where the tip lies inside the goal voxel. We compute the reachable goal voxels of a design by discretizing the robot's configuration space into a grid and iterating over each point on the grid.

- *NM + MP*: We use the Nelder-Mead optimization algorithm to sample new designs instead of ASA. We use motion planning to compute the reachable goal percentage of designs.
- *ASA + G*: We use the ASA optimization algorithm to sample new designs. To evaluate the reachable goal percentage of a design, we do not use motion planning; we instead use the grid-based evaluation approach as is used in *NM + G*.
- *RRT of RRTs*: We use a search strategy based on generating an RRT in the design space to sample new designs instead of ASA. We use motion planning to compute the reachable goal percentage of designs (Torres et al., 2012).

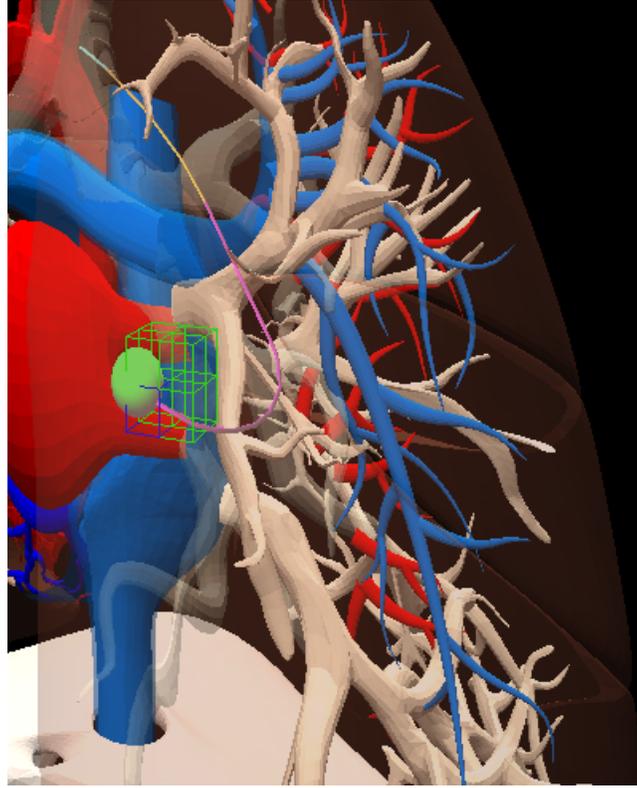


Figure 6.2: A visualization of the random goal region that was generated for the single design optimization experiment. The green voxels in the figure constitute the goal region, i.e. volume of points that the physician wants to reach with the robot's tip. In the figure, we see the concentric tube robot reaching a point within random goal region with its end effector (denoted by the green sphere)..

We compared the above approaches against our full method, which we will denote as “ASA + MP”. We executed each algorithm on the human lung scenario and allotted 6 hours (360 minutes) of computation time per trial. Since the grid-based algorithms, “NP + G” and “ASA + G”, do not ensure that goal voxels are reachable by entirely collision-free paths, we verified the reachable goal voxels of designs generated by these variant by executing 300,000 RRT iterations on each design returned (and we did not count this verification step in the timing results).

We executed 16 trials of each approach and averaged their reachable goal percentage over time to generate the results in Fig. 6.3. The results demonstrate that the use of ASA and motion planning yields favorable results that are, in a sense, greater than the sum of each algorithm's contribution alone, as noted by the performance by other algorithms which utilize only ASA or only motion planning. Moreover, we can see the emergence of convergence of the reachable goal percentage to 100% for the ASA + MP method which highlights the

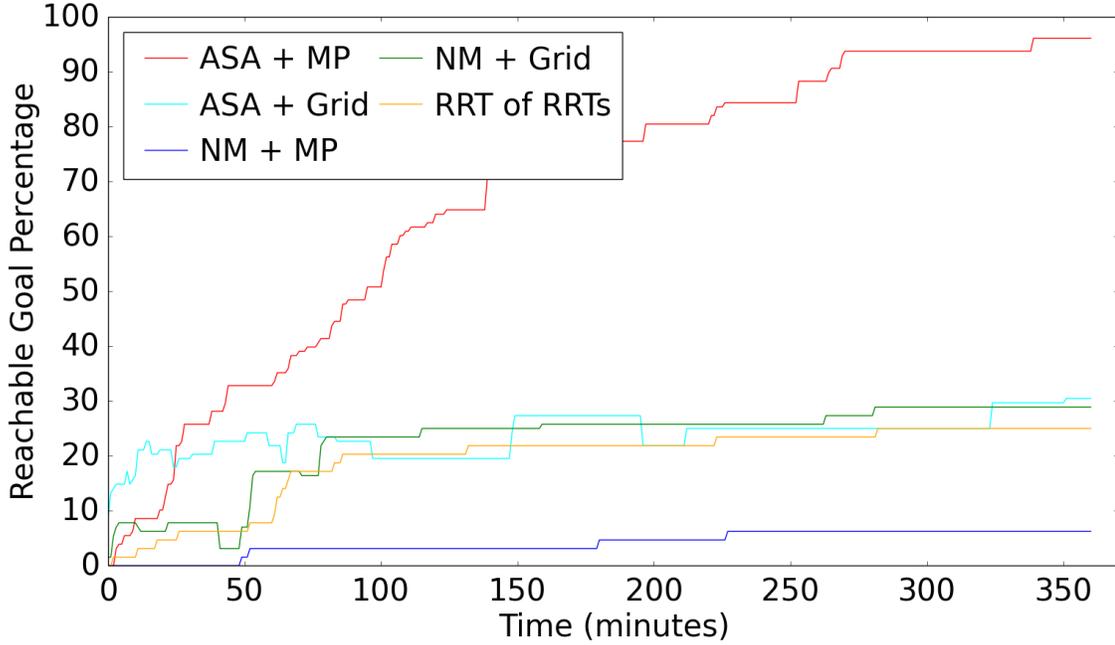


Figure 6.3: We compare the performance of our proposed algorithm (ASA+MP) with other design optimization algorithms for generating an optimal single design for the lung scenario, some of which are inspired by the design algorithm from prior work (Burgner et al., 2013). Moreover, we compare the performance of our algorithm with the method that we extend in this thesis: the algorithm of Torres et al. which uses the method of RRT of RRTs for design optimization (Torres et al., 2012).

established asymptotic optimality of our algorithm.

It should be noted that, interestingly enough, the combination of “NM + MP” did not perform better than “NM + G”, contrary to the expected. This may be due to the fact that the number of trials (16) was not large and hence the Nelder-Mead algorithm, a local optimization method highly sensitive to initial designs, could have had been initialized with pathological random initial states during the ‘NM + MP’ algorithm. For future work, we will conduct experiments with a larger number of trials and longer computation time per trial.

6.2 Set of Designs Experiments

In this section, we consider finding a solution to Problem 2 as described in Section 3.1. For the experiments in this section, we set the goal region G to be set the goal region to the entire lung. We subdivide the interior volume of the lung into 4156 equally-sized cubic voxels for purposes of evaluating voxels reached. In our optimization, we also consider the

start pose of the concentric tube robot as a design parameter, which for this scenario are the additional variables α and β , which correspond to angular offsets in two directions from the rigid bronchoscope’s tangent axis.

6.2.1 Maximizing Reachability of a Design Set of Fixed Size

We first show how the reachable goal percentage of a set of designs is affected by the size of the design set. We considered set sizes $M = \{1, 2, 4, 6\}$. For each $m_i \in M$, we executed our subroutine Alg. 3 to find a design set of size m_i that maximizes the reachability of the right lung. For each trial we recorded how the solutions’ reachable goal percentage progressed over an allotted time of 3 hours, and we averaged the results of 20 trials for each m_i . These progressions of goal reachability over time are shown in Fig. 6.4.

In the time allotted, design sets of larger size were found to reach a larger percentage of the goal region by our design algorithm, with design sets of sizes 1, 2, 4, and 6 being found to reach approximately 70%, 84%, 94%, and 97% of the right lung, respectively. This demonstrates the need to consider collections of designs in order to enable a wider variety of possible surgical procedures. Also, the marginal difference in reachable goal percentage between using 4 and 6 designs highlights the diminishing returns of adding more and more designs to the set. This implies that, depending on the physician’s requirements, we may be able to reduce the design set size while maintaining sufficient coverage of the goal region.

6.2.2 Minimal Design Set with Sufficient Goal Reachability

We next evaluated the ability of our full design algorithm (Alg. 4) to generate a robot design set of minimal size that can reach at least $r_{\text{threshold}} = 95\%$ of the right lung. We used a maximum design set size of $m_{\text{max}} = 12$. We executed 20 trials of our algorithm, with 3 hours of computation time per trial.

We show the average minimum set size found by our algorithm over time in Fig. 6.5. We note that we did not begin averaging results until all trials had found their first design set with a sufficient reachable goal percentage $r_{\text{threshold}}$, which occurred at 116 minutes. The figure shows that, over time up to 3 hours, our algorithm progressively finds smaller and smaller sets of robot designs that can still reach a sufficient percentage of the goal region.

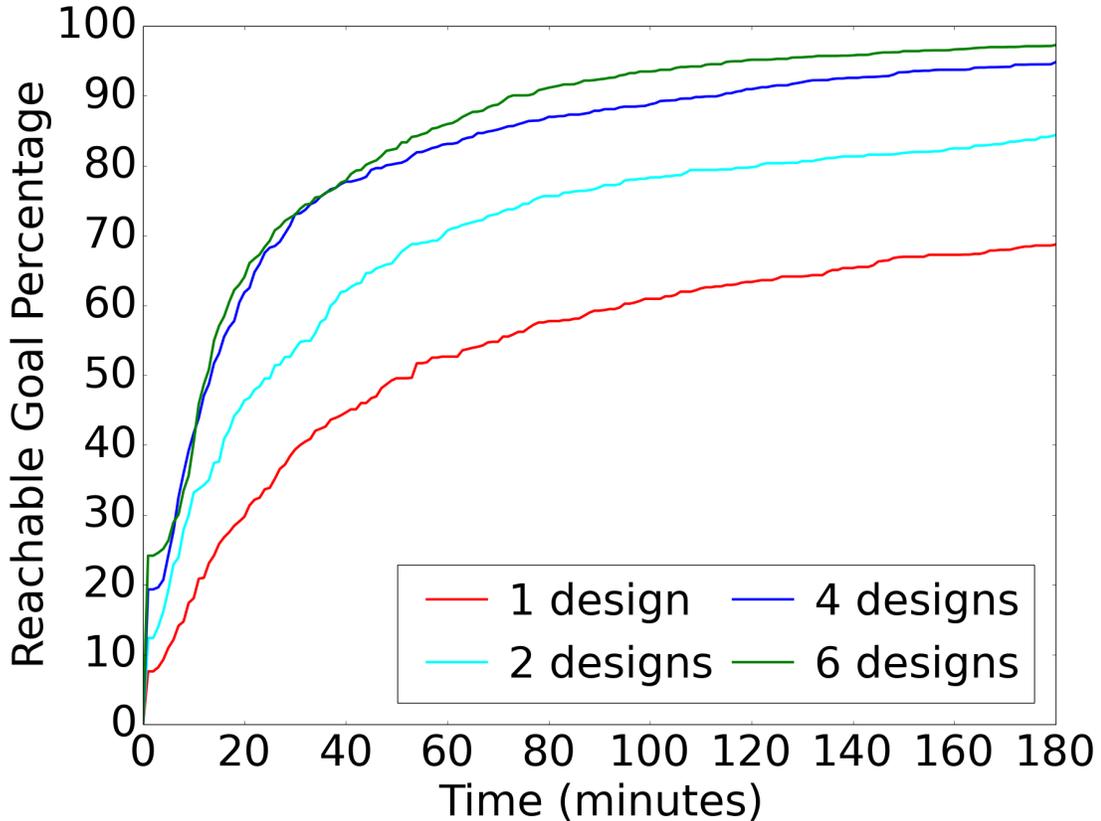


Figure 6.4: We show the performance of Algorithm 3 over time in finding design sets of fixed sizes 1, 2, 4, and 6 that collectively maximize the reachable percentage of a human lung. In general, larger design sets can reach a greater percentages of the goal region.

6.2.3 Benchmarking Variations on Algorithm

We next compare our method with different approaches to design optimization. We compare our algorithm for fixed-size design sets against two other design optimization algorithms that were described in the single design experiments section (Sec. 6.1): “NM + G” and “NM + MP”.

We compare the above approaches against our full method, which we denote as “ASA + MP”. We executed each algorithm on the human lung scenario with a fixed design set size of 2. Since the “NM + G” variant does not ensure that goal voxels are reachable by entirely collision-free paths, we verified the reachable goal voxels of designs generated by this variant by executing 200,000 RRT iterations on each design returned (and we did not count this verification step in the timing results).

We executed 20 trials of each approach and averaged their reachable goal percentage

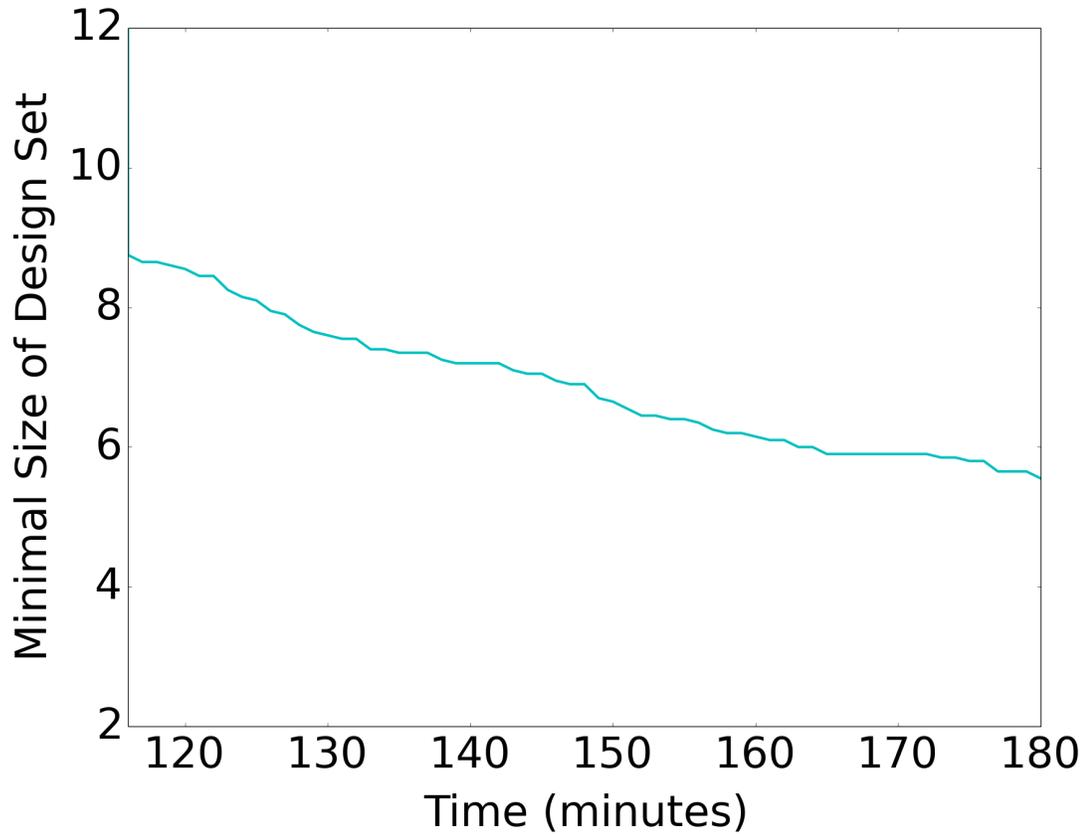


Figure 6.5: We show the performance of Algorithm 4 over time in finding a design set of minimal size that can reach 95% of the human lung. This plot is averaged over 20 trials and we began the plot when all 20 trials had found their first design set capable of reaching 95% of the lung.

over time to generate the results in Fig. 6.6. The results demonstrate that (1) using motion planning to determine a set of designs' reachable goal percentage and (2) using a globally optimal optimization algorithm like ASA enable us to most quickly compute sets of concentric tube robot designs that can reach large portions of the goal region without colliding with anatomical obstacles.

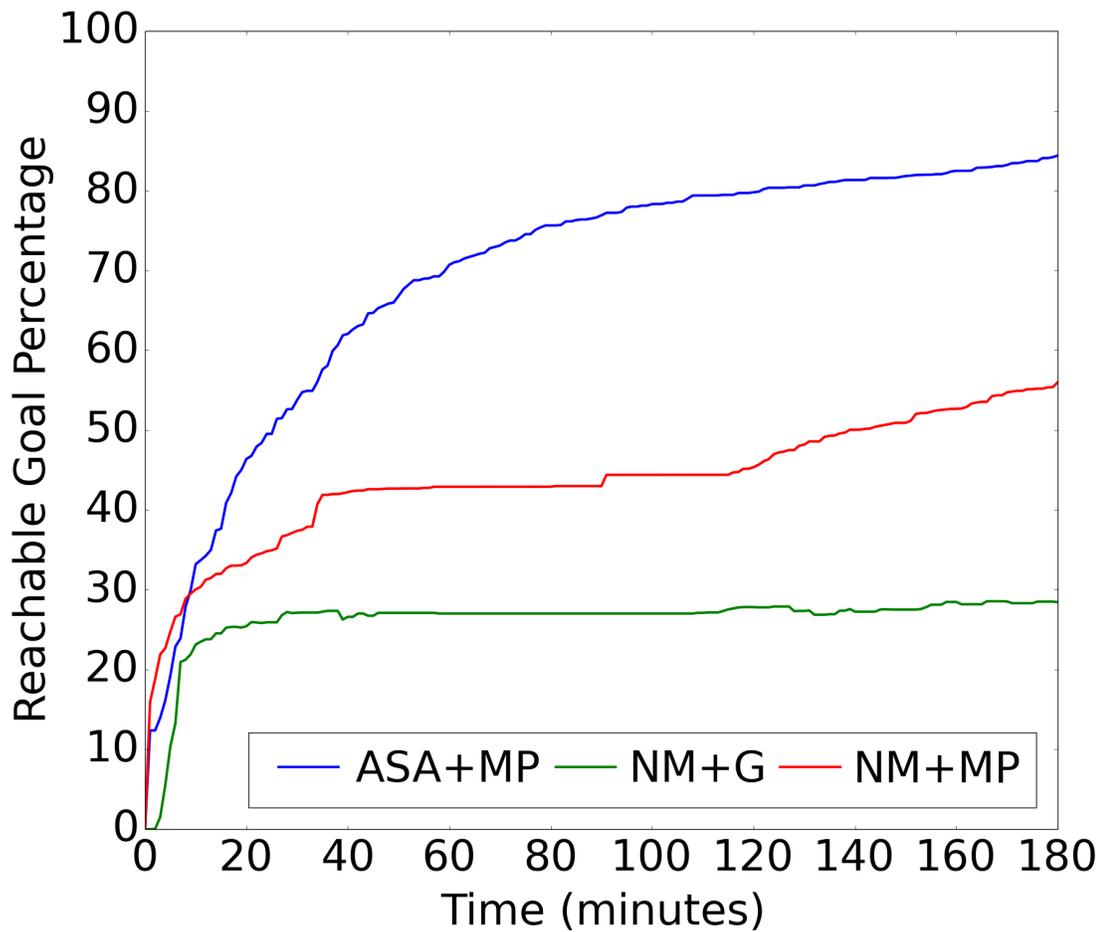


Figure 6.6: We compare the performance of our proposed algorithm (ASA+MP) with variants inspired by a design algorithm from prior work (Burgner et al., 2013). This prior method used the Nelder-Mead (NM) optimization algorithm, whereas we use a globally optimal algorithm called Adaptive Simulated Annealing (ASA). We also extended the prior work to consider obstacle avoidance in the entire deployment of the concentric tube robot by using motion planning. These extensions result in finding better sets of robot designs in less time.

Chapter 7

Conclusions

In this thesis, we presented design optimization algorithms capable of generating an optimal single concentric tube robot design and a minimal set of designs on an application- and patient-specific basis, that can reach a sufficient percentage of a physician-specified goal region while avoiding contact with anatomical obstacles. Our algorithms interleave a globally optimal stochastic search over the space of robot designs with a sampling-based motion planner in the robot's configuration space.

We outlined a proof of the theoretical guarantees of asymptotic optimality for our single design optimization algorithm and empirically showed that our algorithm was able to compute higher quality designs faster when compared to other design optimization algorithms. We also showed experimentally that our algorithm for computing minimal sets of designs and showed that our method was able to generate a low number of design sets capable of reaching almost all of the right lung.

In this work, we used one set of anatomical obstacles to generate our minimal design set. In clinical settings, we would want to find a set of robot designs that would enable a wide variety of procedures on many different patients of varying internal anatomy (and consequently varying anatomical obstacles). To accommodate this use case, our algorithm can be extended to consider many different sets of anatomical obstacles in assessing goal region reachability.

For future work, we will flesh out the proofs of the theorems and lemmas enumerated in Moreover, another avenue of interest for the minimal designs algorithm is the rigorous analysis of its optimality guarantees, as was done for the single design case in this thesis.

There is also potential for future investigation in design quality measures and objectives to be used in addition to goal reachability, such as maximizing probability of surgical success and minimizing tissue damage.

BIBLIOGRAPHY

- Bergeles, C., Gosline, A. H., Vasilyev, N. V., Codd, P., del Nido, P. J., and Dupont, P. E. (2015). Concentric tube robot design and optimization based on task and anatomical constraints. *IEEE Trans. Robotics*.
- Burgner, J., Gilbert, H. B., and Webster III, R. J. (2013). On the Computational Design of Concentric Tube Robots: Incorporating Volume-Based Objectives. In *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, pages 1185–1190.
- Burgner-Kahrs, J., Gilbert, H. B., Granna, J., Swaney, P. J., and Iii, R. J. W. (2014). Workspace Characterization for Concentric Tube Continuum Robots. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, pages 1269–1275.
- Chocron, O. (2008). Evolutionary design of modular robotic arms. *Robotica*, 26(3):323–330.
- Dupont, P. E., Lock, J., and Butler, E. (2009). Torsional kinematic model for concentric tube robots. In *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, pages 3851–3858.
- Gilbert, H. B., Rucker, D. C., and Webster III, R. J. (2013). Concentric tube robots: The state of the art and future directions. In *Int. Symp. Robotics Research (ISRR)*.
- Ha, J., Park, F. C., and Dupont, P. E. (2014). Achieving Elastic Stability of Concentric Tube Robots Through Optimization of Tube Precurvature. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*.
- Ingber, L. (1989). Very Fast Simulated Re-Annealing. *Mathematical and computer modelling*, 12(8):967–973.
- Johnson, H. J., McCormick, M., Ibáñez, L., and Insight Software Consortium (2013). The ITK Software Guide. Available: <http://www.itk.org/ItkSoftwareGuide.pdf>.
- Karaman, S. and Frazzoli, E. (2010). Incremental Sampling-based Algorithms for Optimal Motion Planning. In *Robotics: Science and Systems (RSS)*.
- Katragadda, L. K. (1997). *A Language and Framework for Robotic Design*. PhD thesis, Carnegie Mellon University.
- LaValle, S. M. (2006). *Planning Algorithms*. Cambridge University Press, Cambridge, U.K.
- Leger, C. (1999). *Automated Synthesis and Optimization of Robot Configurations: An Evolutionary Approach*. PhD thesis, Carnegie Mellon University.
- Lock, J. and Dupont, P. E. (2011). Friction modeling in concentric tube robots. In *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, pages 1139–1146.

- Lock, J., Laing, G., Mahvash, M., and Dupont, P. E. (2010). Quasistatic modeling of concentric tube robots with external loads. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, pages 2325–2332.
- Lyons, L. A., Webster III, R. J., and Alterovitz, R. (2009). Motion planning for active canulas. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, pages 801–806.
- Merlet, J.-P. (2005). Optimal design of robots. In *Robotics: Science and Systems (RSS)*.
- Park, J., Chang, P., and Yang, J. (2003). Task-oriented Design of Robot Kinematics using the Grid Method. *J. Advanced Robotics*, 17(9):879–907.
- Reif, J. H. (1979). Complexity of the Mover’s Problem and Generalizations. In *20th Annual IEEE Symp. on Foundations of Computer Science*, pages 421–427.
- Rucker, D. C. (2011). *The mechanics of continuum robots: model-based sensing and control*. PhD thesis, Vanderbilt University.
- Rucker, D. C., Jones, B. A., and Webster III, R. J. (2010). A geometrically exact model for externally loaded concentric-tube continuum robots. *IEEE Trans. Robotics*, 26(5):769–780.
- Rucker, D. C. and Webster III, R. J. (2009). Parsimonious evaluation of concentric-tube continuum robot equilibrium conformation. *IEEE Trans. Biomedical Engineering*, 56(9):2308–2311.
- Salle, D., Bidaud, P., and Morel, G. (2004). Optimal design of high dexterity modular MIS instrument for coronary artery bypass grafting. In *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, pages 1276–1281.
- Sears, P. and Dupont, P. E. (2006). A steerable needle technology using curved concentric tubes. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, pages 2850–2856.
- Torres, L. G. and Alterovitz, R. (2011). Motion planning for concentric tube robots using mechanics-based models. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, pages 5153–5159.
- Torres, L. G., Baykal, C., and Alterovitz, R. (2014). Interactive-rate motion planning for concentric tube robots. In *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, pages 1915–1921.
- Torres, L. G., Webster III, R. J., and Alterovitz, R. (2012). Task-oriented design of concentric tube robots using mechanics-based models. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, pages 4449–4455.

- Trovato, K. and Popovic, A. (2009). Collision-free 6D non-holonomic planning for nested cannulas. In *Proc. SPIE Medical Imaging*, volume 7261.
- Vijaykumar, R., Waldron, K. J., and Tsai, M. J. (1986). Geometric optimization of serial chain manipulator structures for working volume and dexterity. *Int. J. Robotics Research*, 5(2):91–103.