Asymptotically Optimal Design of Piecewise Cylindrical Robots using Motion Planning

Cenk Baykal*
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology, USA
baykal@mit.edu

Ron Alterovitz
Department of Computer Science
University of North Carolina at Chapel Hill, USA
ron@cs.unc.edu

***Abstract*—In highly constrained settings, e.g., a tentacle-like medical robot maneuvering through narrow cavities in the body for minimally invasive surgery, it may be difficult or impossible for a robot with a generic kinematic design to reach all desirable targets while avoiding obstacles. We introduce a design optimization method to compute kinematic design parameters that enable a single robot to reach as many desirable goal regions as possible while avoiding obstacles in an environment. We focus on the kinematic design of piecewise cylindrical robots, robotic manipulators whose shape can be modeled via cylindrical components. Our method appropriately integrates sampling-based motion planning in configuration space into stochastic optimization in design space so that, over time, our evaluation of a design's ability to reach goals increases in accuracy and our selected designs approach global optimality. We prove the asymptotic optimality of our method and demonstrate performance in simulation for (i) a serial manipulator and (ii) a concentric tube robot, a tentacle-like medical robot that can bend around anatomical obstacles to safely reach clinically-relevant goal regions.**

I. Introduction

In a cluttered environment, the ability of a robotic manipulator to reach desired targets while avoiding obstacles depends significantly on the robot's kinematic design. A robot's kinematic design can be seen as a set of kinematic parameters that define a robot's shape and are fixed throughout the robot's use, e.g., the length of each link of a serial manipulator or the lengths and curvatures of tubes in a concentric tube robot [8, 11]. In highly constrained settings, e.g., a tentacle-like robot maneuvering through narrow cavities in the body for minimally invasive surgery, it may be difficult or impossible for a robot with a generic kinematic design to reach all desirable targets while avoiding obstacles (see Figure 1).

Fortunately, advances in methods that enable the rapid fabrication of customized robot designs is introducing the potential to create robots that are kinematically optimized on a task-specific basis. Advances in 3D printing enable the rapid creation of robots with links of customizable lengths. Customized medical robots, like concentric tube robots, can be created in minutes by shape-setting or 3D printing [10, 22]. Our objective is to computationally optimize the kinematic design parameters of a robotic manipulator on a task-specific basis: given the shapes of obstacles in the environment as well as goal regions the robot should be capable of reaching, we aim to compute a single robot design that can reach as many of the goal regions as possible while avoiding obstacles.

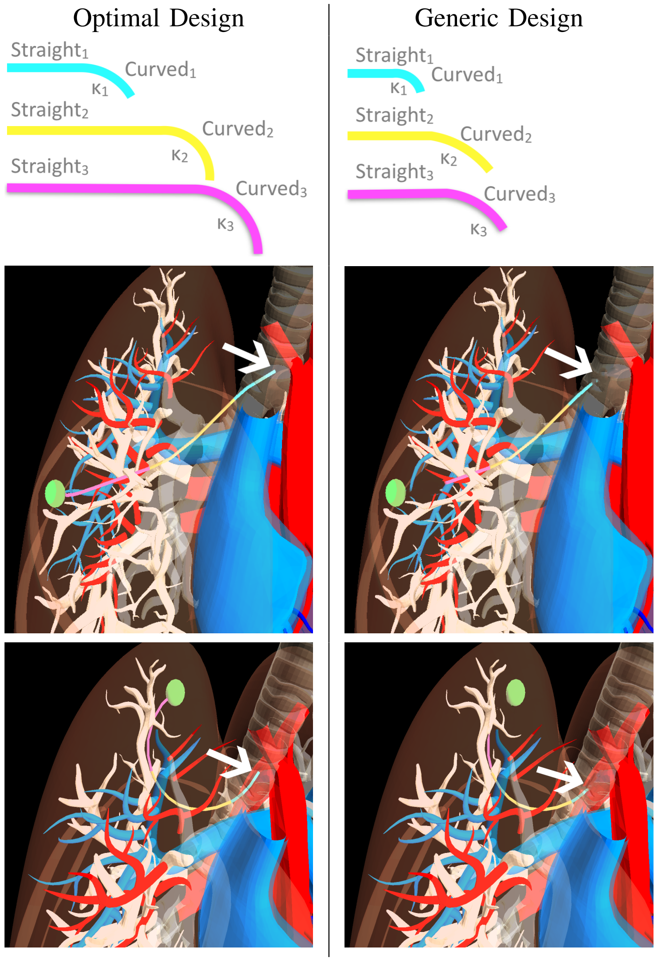

Fig. 1. Optimizing the kinematic design of a robotic manipulator can enable it to reach more goal regions in a cluttered environment. In this example, the objective is to optimize the design of a concentric tube robot, a surgical manipulator composed of nested, precurved tubes (cyan, yellow, magenta) whose lengths and curvatures can be customized (top). The objective is to move from a bronchial tube (white arrow) to reach goal regions (green spheres) in the lung while avoiding anatomical obstacles, e.g., blood vessels (red, blue) and bronchial tubes (off-white). A generic design may fail to reach some goal regions in a cluttered environment (right column), while an optimized design (left column) has the potential to reach more goal regions.

In this paper, we specifically focus on optimizing the

*Work done as a student at University of North Carolina at Chapel Hill.

kinematic design of *piecewise cylindrical robots*, a subclass of robotic manipulators whose shape can be modeled via a connected sequence of cylindrical components. This model can be applied to standard multi-link robot arms by modeling each link as a cylinder, where the length of each link is a kinematic design parameter. Another type of robot that can be modeled as piecewise cylindrical is the concentric tube robot, a tentacle-like robot for minimally invasive surgery that can curve around anatomical obstacles to reach surgical sites in constrained spaces [8, 11]. Concentric tube robots are composed of nested, pre-curved tubes, where each tube is typically shaped with a straight section followed by a pre-curved constant-curvature section. Each of the robot's component tubes can be independently rotated or extended, enabling the entire device to change shape as the nested tubes elastically interact. This robot's kinematic design parameters include the pre-curvatures and lengths of each constituent tube. These parameters have a significant impact on the surgical targets reachable by the device in constrained spaces, so proper selection of kinematic design parameters for a patient's anatomy is critical to the success of a medical procedure.

Optimizing a robot's kinematic design on a task-specific basis is challenging. We desire to compute high quality designs reasonably quickly (i.e., minutes, not days), particularly for medical applications in which the physician customizes the robot design based on a patient-specific anatomy identified in medical images. However, the kinematic design space of a robot may be large, and for any candidate design we must evaluate whether that design can move from an initial configuration to the goal regions while avoiding obstacles. This implies we need to compute motion plans to multiple goal regions for successively selected designs, but current state-of-the-art motion planners based on sampling-based methods cannot determine with certainty in finite time whether a goal region can be reached by a particular design.

Our novel contributions are as follows. First, we introduce a new method for optimizing the kinematic design parameters of a single piecewise cylindrical robot on a task-specific basis by appropriately integrating sampling-based motion planning into iterations of a stochastic optimization method for design selection. We implement the integration so that, over time, our reachability evaluations increase in accuracy and our design selections improve toward global optimality. Second, we analyze our algorithm and prove asymptotic optimality, i.e., almost sure convergence to a globally optimal design, which guarantees that our method avoids getting trapped in local optima. Third, we demonstrate the broad applicability and effectiveness of our design optimization method via evaluations for two distinct piecewise cylindrical robots: (i) a 4-link serial manipulator, and (ii) a concentric tube medical robot.

II. Related Work

Our approach to optimizing the kinematic design of piecewise cylindrical robots integrates prior work in robot design optimization, stochastic search, and robot motion planning. Prior work has investigated combinatorial approaches to design optimization over a finite and discrete set of robot features. Examples include optimizing over discrete components of modular robots [14, 29], monopedal jumping robots [26], snake-like and multi-modal robots [24, 33], and kinematic chains such as proteins [7]. In this paper, we focus on piecewise cylindrical robots with continuous design parameters.

There has been extensive work on optimizing the kinematic design of serial manipulators. Approaches have optimized various metrics and have used genetic algorithms [6, 16, 19, 30], interval analysis [21], geometric methods [32], and grid-based methods [25]. These methods typically lack theoretical performance guarantees or achieve computational tractability by imposing significant assumptions on the robot's workspace and by using simplified kinematic models, which limit the effectiveness and applicability of the optimization procedure.

Kinematic design optimization for concentric tube robots is particularly challenging due to their complex kinematics, which is computationally expensive to evaluate due to the complex elastic and torsional interactions of their constituent tubes. Morimoto et al. present a complementary approach to automatic design optimization by providing a human with an intuitive interface to manually design the tubes [23]. Bergeles et al. computationally optimize the robot's design to reach a set of points without colliding with anatomical obstacles [2]. For computational efficiency, they reduce the motion planning problem to finding goal configurations that can reach the targets and do not offer a global optimality guarantee. Burgner et al. combine a grid-based evaluation of the robot's kinematics in configuration space with a nonlinear optimization method over the design space to maximize the reachable region of points subject to anatomical constraints [3]. Burgner et al. extended this work to characterize the workspace of concentric tube robots [4, 5]. Ha et al. present a method for generating designs to maximize device stability [12]. By focusing only on computing designs and goal configurations, the works above cannot guarantee that a collision-free path from start to goal exists for the computed design. Torres et al. integrated a motion planner into concentric tube robot design to ensure the computed design is able to avoid obstacles en route to specific points [31], but offers slow performance. Baykal et al. investigated computing minimal sets of concentric tube robot designs to reach multiple targets [1], although no analysis was provided regarding a guarantee on optimality. In this paper, we focus on the broader class of piecewise cylindrical robots.

III. Problem Definition

A robot's design $\mathbf{d}$ is an n-dimensional vector of kinematic parameters that correspond to physical properties of the robot's shape that are fixed for the duration of a given task. This vector includes kinematic parameters such as the length of each link of a serial manipulator or the lengths and curvatures of tubes in a concentric tube robot. The *design space* $\mathcal{D} \subset \mathbb{R}^n$ of a robot is the n-dimensional compact set corresponding to the space of all possible kinematic designs of the robot.

We assume that the robot operates in a workspace $\mathcal{W} \subseteq \mathbb{R}^k$ containing a compact set of obstacles $\mathcal{O} \subset \mathcal{W}$, where $k \in \{2, 3\}$. The robot should be designed so it can reach (while avoiding obstacles) a set of m user-specified goal regions, where each goal region $\mathcal{G}_i \subset \mathcal{W}$, $i \in \{1, \ldots, m\}$ is defined as a volume of workspace positions. We define $\mathcal{G} = \{\mathcal{G}_i : i = 1, \ldots, m\}$ as the union of the goal regions.

Let the compact set $\mathcal{Q} \subset \mathbb{R}^d$ denote the d-dimensional configuration space of the robot. Since the shape of a robot at any configuration is a function of its design $\mathbf{d}$, the set of configurations for which the robot's shape does not intersect an obstacle is dependent on $\mathbf{d}$. Thus, we denote the set of collision-free configurations for a robot of design $\mathbf{d}$ as $\mathcal{Q}_{\mathbf{d}}^{\text{free}} \subset \mathcal{Q}$. We model the shape of a piecewise cylindrical robot with design $\mathbf{d} \in \mathcal{D}$ at configuration $\mathbf{q} \in \mathcal{Q}$ by the mapping $\text{Shape} : \mathcal{D} \times \mathcal{Q} \to ([0, 1] \to \mathcal{W})$, which defines the curve of the robot's backbone, and a radius r of its circular cross-section. Note that $\text{Shape}(\mathbf{d}, \mathbf{q})(0)$ and $\text{Shape}(\mathbf{d}, \mathbf{q})(1)$ correspond to the robot's base and end-effector points respectively. We assume Shape is computed using an accurate kinematic model. We define $\mathbf{q}_0 \in \mathcal{Q}$ as the robot's start configuration. The robot's motion is a *path* in the configuration space $\mathcal{Q}$ defined by the continuous function $\sigma : [0, 1] \to \mathcal{Q}$, where $\sigma(0) = \mathbf{q}_0$. A path σ executed under robot design $\mathbf{d}$ is *collision-free* if it lies entirely in the collision-free configuration space, that is, if $\sigma(\tau) \in \mathcal{Q}_{\text{free}}^{\mathbf{d}}$ for all $\tau \in [0, 1]$.

Our objective is to compute an optimal robot design $\mathbf{d}^* \in \mathcal{D}$ that enables the robot to reach as many goal regions in $\mathcal{G}$ as possible in a *safe* manner, i.e., via collision-free paths that avoid the workspace obstacles $\mathcal{O}$. The quality of a design $\mathbf{d} \in \mathcal{D}$ is defined with respect to the extent of the design's *reachability* to the goal regions in $\mathcal{G}$. That is, the objective function value of $\mathbf{d}$ is expressed as the percentage of goal regions in G that are reachable with design $\mathbf{d}$ relative to the total number of goal regions in $\mathcal{G}$. Formally, the reachability of design $\mathbf{d}$ is given by the mapping $R(\mathbf{d}) : \mathcal{D} \to [0, 1]$:

$$R(\mathbf{d}) := \frac{|\text{GoalRegionsReachable}(\mathbf{d})|}{|\mathcal{G}|}, \tag{1}$$

where $\text{GoalRegionsReachable}(\mathbf{d}) : \mathcal{D} \to 2^{|\mathcal{G}|}$ denotes the set of goal regions that the robot of design $\mathbf{d}$ can reach with its end effector by following a collision-free path. $R(\mathbf{d})$ expresses the *reachable goal percentage* of the robot under design $\mathbf{d}$, which we seek to maximize. We formalize the objective of kinematic design optimization as follows: Given an environment $\mathcal{W} \subseteq \mathbb{R}^k$ (where $k = 2$ or 3), a set of obstacles $\mathcal{O} \subset \mathcal{W}$, and a set of user-specified goal regions $\mathcal{G}$, generate a design $\mathbf{d}^*$ that maximizes the reachable goal percentage, i.e.,

$$\mathbf{d}^* \in \underset{\mathbf{d} \in \mathcal{D}}{\operatorname{argmax}}\, R(\mathbf{d}). \tag{2}$$

IV. Methods

In this section, we present our algorithm for optimizing the kinematic design parameters of a single piecewise cylindrical robot to maximize the robot's reachable goal percentage while avoiding obstacles in a task-specific environment.

A. Method Overview and the Key Challenge

Our design optimization approach combines a stochastic search in the robot's design space $\mathcal{D}$ with a sampling-based motion planner in the robot's configuration space $\mathcal{Q}$ to efficiently generate designs with high reachability. To select candidate designs for evaluation, we use Adaptive Simulated Annealing (ASA) [13], a global optimization algorithm. For each selected design, we use the Rapidly-exploring Random Tree (RRT) [17] algorithm to estimate the design's reachable workspace and evaluate its reachable goal percentage. We provide an overview of our approach in Algs. 1 and 2 and formally prove the method's asymptotic optimality in Sec. V.

To ensure that we converge toward a globally optimal design, a key challenge we must address is that state-of-the-art, practical motion planners cannot guarantee *completeness* [17], i.e., they cannot always in finite time answer the question of whether a collision-free motion plan exists from the start configuration to a goal region. This limitation of current state-of-the-art motion planners introduces a significant challenge for design optimization; to use a standard optimization algorithm to optimize the design $\mathbf{d}$ in equation 2, a motion planner must evaluate the reachable goal percentage accurately and in finite time in each iteration of the optimization algorithm. Commonly used sampling-based motion planners only offer *probabilistic completeness* (and in some cases also asymptotic optimality in terms of path quality), meaning the probability that they will produce a collision-free path (if one exists) to a goal region approaches 1 as more time is spent [17]. Terminating a sampling-based motion planner after finite time may result in an incorrect computation of the reachable goal percentage. *The lack of full completeness makes it impossible to simply plug a standard sampling-based motion planner into a standard optimization algorithm and expect convergence toward a globally optimal design.* We address this challenge by appropriately integrating sampling-based motion planning into stochastic optimization in design space so that, over time, our reachable goal percentage evaluations increase in accuracy and our selected designs approach global optimality.

B. Evaluating a Design's Reachable Goal Percentage

Evaluating the objective function value $R(\mathbf{d})$ in equation (1) for an arbitrary design $\mathbf{d} \in \mathcal{D}$ requires computing $\text{GoalRegionsReachable}(\mathbf{d})$, the set of goal regions that design $\mathbf{d}$ can reach by executing collision-free paths. Thus, evaluating the reachability of a design is fundamentally a motion planning problem, which is known to be PSPACE-hard [27]. This renders the use of exact evaluation methods computationally intractable and motivates the use of a sampling-based motion planning algorithm, such as the Rapidly-exploring Random Tree (RRT) [17], to generate approximations of a design's reachability (albeit an approximation that can improve over time, as will be discussed in Sec. IV-D).

For a given design $\mathbf{d} \in \mathcal{D}$ and a start configuration $\mathbf{q}_0$, the RRT algorithm incrementally constructs a tree of configurations that can be reached by collision-free paths from the root of the tree, $\mathbf{q}_0$. For a given design, we run the RRT algorithm

Algorithm 1 Select and evaluate a kinematic design

```
Input:
        $\mathcal{G}$: set of goal regions
        $\mathcal{O}$: set of environmental obstacles
        $\mathbf{d}_{\text{current}}$: previously considered robot design
        $T$: ASA's current annealing temperature
        $i$: number of RRT iterations to execute
Output:
        $\mathbf{d}_{\text{new}}$: new robot design
        $\hat{R}_{\text{new}}$: approximate reachable goal percentage of $\mathbf{d}_{\text{new}}$
1: $\mathbf{d}_{\text{new}} \leftarrow \texttt{SampleDesign}(\mathbf{d}_{\text{current}}, T)$;
2: $\texttt{goalRegionsReached} \leftarrow \texttt{RRT}(\mathbf{d}_{\text{new}}, i, \mathcal{O})$;
3: $\hat{R}_{\text{new}} \leftarrow |\texttt{goalRegionsReached}|/|\mathcal{G}|$;
4: return $\mathbf{d}_{\text{new}}, \hat{R}_{\text{new}}$;
```

for $i \in \mathbb{N}_+$ iterations and iterate over the configurations in the constructed tree to compute the set of goal regions that can be feasibly reached by the robot with design $\mathbf{d}$ (Line 2, Alg. 1). From this we can approximate the design's reachable goal percentage in a computationally tractable manner (Line 3, Alg. 1). Because RRT provides *probabilistic completeness*, as we increase the iterations i of RRT, the probability of our approximation $\hat{R}_i(\mathbf{d})$ being equal to the true $R(\mathbf{d})$ approaches 1. For any finite i, our approximations of the reachable goal percentage at each iteration is ensured to be a lower bound of the ground-truth reachability, i.e., $\hat{R}_i(\mathbf{d}) \leq R(\mathbf{d})$.

A key challenge is appropriately setting the number of iterations i the RRT will run for. In Sec. IV-D, we introduce an approach to setting i in Alg. 1 that ensures asymptotic optimality of the design optimization.

C. Selecting Designs

We use the ASA algorithm [13, 20] for optimizing the kinematic design to maximize the reachable goal percentage. We use ASA because it is a global optimization method that escapes local optima, it is efficient in practice for problems in high dimensional spaces, and it has favorable algorithmic properties which we exploit. Specifically, we are able to use sampling-based motion planning in each iteration of ASA in a manner that ensures asymptotic optimality of the design, as discussed in Sec. IV-D.

Our ASA-based algorithm is shown as Alg. 2 and operates similar to a hill climbing algorithm in that it is centered on a design $\mathbf{d}_{\text{current}}$ that it incrementally attempts to improve. However, unlike a hill climbing algorithm, the algorithm may in some iterations select an inferior design, which enables escaping local minima. Next designs are determined by sampling a new design (`SampleDesign`; Line 1, Alg. 1) and deciding whether to accept that new design (`Accept`; Line 6, Alg. 2), with both procedures being highly dependent on a temperature parameter $T \in \mathbb{R}_{\geq 0}$. `Accept` returns true if the sampled design $\mathbf{d}'$ is higher quality than $\mathbf{d}_{\text{current}}$ (i.e., $\hat{R}' > \hat{R}_{\text{current}}$) or with some probability (dependent on T) if $\mathbf{d}'$ is inferior. The temperature variable is initially set to a high value and is decreased after each iteration based on a cooling schedule (Alg. 2, Line 13). When T is high, ASA is more likely to sample states that are far away from $\mathbf{d}_{\text{current}}$ and also more likely to probabilistically accept inferior designs, which leads to exploratory behavior initially. As T is cooled down over time, ASA samples states in smaller neighborhoods around $\mathbf{d}_{\text{current}}$ and is increasingly less likely to accept inferior designs, which leads to eventual convergence to a high quality design. We cache the best found design (Alg. 2, Line 10) so the best found design is returned when the algorithm terminates.

Algorithm 2 Iterative design optimization

```
Input:
        $\mathcal{G}$: set of goal regions
        $\mathcal{O}$: set of environmental obstacles
        $i_{\text{init}}$: initial number of RRT iterations
        $i_{\triangle}$: additional RRT iterations after each sample
        $\mathbf{d}_{\text{init}}$(optional): initial design for the search
Output:
        $\mathbf{d}^*$: a robot design that maximizes (1)
1: $i \leftarrow i_{\text{initial}}$; $T \leftarrow T_{\text{initial}}$; $\hat{R}_{\text{current}} \leftarrow 0$; $\hat{R}^* \leftarrow 0$;
2: $\mathbf{d}_{\text{current}} \leftarrow$ random initial design or $\mathbf{d}_{\text{init}}$ if provided;
3: $\mathbf{d}^* \leftarrow \mathbf{d}_{\text{current}}$;
4: while allotted time remains do
5:     $\mathbf{d}', \hat{R}' \leftarrow \texttt{Algorithm1}(\mathcal{G}, \mathcal{O}, \mathbf{d}_{\text{current}}, T, i)$
6:     if $\texttt{Accept}(\hat{R}', \hat{R}_{\text{current}}, T)$ then
7:         $\mathbf{d}_{\text{current}} \leftarrow \mathbf{d}'$;
8:         $\hat{R}_{\text{current}} \leftarrow \hat{R}'$;
9:     if $\hat{R}' > \hat{R}^*$ then
10:        $\mathbf{d}^* \leftarrow \mathbf{d}'$;
11:        $\hat{R}^* \leftarrow \hat{R}'$;
12:    $i \leftarrow i + i_{\triangle}$;
13:    $T \leftarrow \texttt{UpdateTemperature}(T)$;
14: return $\mathbf{d}^*$
```

D. Integrating Motion Planning into Stochastic Optimization

A key requirement to converging toward a globally optimal design is an accurate evaluation of any candidate design's reachable goal percentage. This implies we need to compute motion plans to multiple goal regions for each design considered in the optimization, but current state-of-the-art motion planners based on sampling-based methods cannot specify with certainty in finite time whether a goal can be reached by a particular design.

To address this challenge, we use a simple-to-implement idea: we incrementally increase the number of RRT iterations by $i_{\triangle}$ after each sampled design (Line 13, Alg. 2). This approach ensures that our approximations become increasingly accurate over time. This approach is sufficient for establishing the asymptotic optimality of our algorithm (see Sec. V). This approach also has a secondary benefit: it enables us to more quickly (but more coarsely) evaluate many designs in the initial iterations and subsequently evaluate candidate designs with higher accuracy (albeit at a slower rate) in later iterations.

V. Analysis

We prove under mild assumptions that the design computed by our algorithm almost surely converges [9] to a globally optimal design. The outline of our proof is as follows. First, we establish that the set of optimal designs with respect to the problem in equation 2 has strictly positive measure. Then, we show that by property of the ASA algorithm, optimal designs will be sampled and evaluated infinitely often. We conclude by proving that, eventually, an optimal design will be sampled and evaluated accurately by the RRT algorithm.

A. Preliminaries

Let $\gamma : \mathcal{Q} \times \mathcal{Q} \to \mathbb{R}_{\geq 0}$ be the distance function in $\mathcal{Q}$ for any arbitrary $\mathbf{d} \in \mathcal{D}$ defined by $\gamma(\mathbf{q}, \mathbf{q}') = ||\text{Shape}(\mathbf{d}, \mathbf{q}) - \text{Shape}(\mathbf{d}, \mathbf{q}')||_\infty$ [17]. For any configuration $\mathbf{q} \in \mathcal{Q}$, the open ball of radius ϵ centered at $\mathbf{q}$, $\{\mathbf{q}' \in \mathcal{Q} \mid \gamma(\mathbf{q}, \mathbf{q}') < \epsilon\}$, is denoted by $B_\epsilon(\mathbf{q})$. A collision-free path under design $\mathbf{d}$, $\sigma : [0, 1] \to \mathcal{Q}^{\mathbf{d}}_{\text{free}}$, is said to have ξ-clearance if

$$\forall \mathbf{q} \in \sigma([0, 1]) \quad \inf_{\mathbf{o} \in \mathcal{O}} ||\text{Shape}(\mathbf{d}, \mathbf{q}) - \mathbf{o}||_\infty \geq \xi,$$

where the infimum is taken over all the obstacle points in the workspace, with a slight abuse of notation in taking the norm of a function and the obstacle point $\mathbf{o}$. This analysis focuses on the piecewise cylindrical robot's backbone but can extend to a cross-sectional radius $r > 0$ by appropriately accounting for the thickness whenever measuring a distance from the robot's shape to an obstacle in the workspace.

Assumption 1 (Goal Regions as Open Sets)**.** *Each goal region $G_i \in \mathcal{W}$, $i \in [m]$, is defined as an open set.*

Assumption 2 (Continuity of the Shape Function)**.** *Shape : $\mathcal{D} \times \mathcal{Q} \to ([0, 1] \to \mathcal{W})$ is continuous over the domain $\mathcal{D} \times \mathcal{Q}$.*

Assumption 1 rules out pathological problem instances where only a single optimal design lying on the boundary of the design space exists. Assumption 2 guarantees that robots of similar designs have similar shapes at similar configurations.

B. Sampling Optimal Designs Infinitely Often

Lemma 1 (Paths with Non-zero Clearance)**.** *Under any design $\mathbf{d} \in \mathcal{D}$, any collision-free path $\sigma : [0, 1] \to \mathcal{Q}^{\mathbf{d}}_{free}$ has ξ-clearance for some $\xi > 0$.*

Proof: For any $\mathbf{d} \in \mathcal{D}$ and $\sigma : [0, 1] \to \mathcal{Q}^{\mathbf{d}}_{\text{free}}$, define $\mathcal{W}_\sigma = \{\text{Shape}(\mathbf{d}, \mathbf{q}, s) : \mathbf{q} \in \sigma([0, 1]), s \in [0, 1]\}$, i.e., the set of workspace points the robot occupies along the path σ. Note that $\mathcal{W}_\sigma$ is compact since Shape is continuous on a compact domain. Since $\mathcal{O}$ is also compact and $\mathcal{W}_\sigma$ and $\mathcal{O}$ are disjoint by definition of collision-free paths, it follows by continuity of norms that $0 < \inf \{||\mathbf{w} - \mathbf{o}||_\infty : \mathbf{o} \in \mathcal{O}, \mathbf{w} \in \mathcal{W}_\sigma\} = \xi$ for some $\xi > 0$. ■

Lemma 2 (Sufficient Condition for Reachability)**.** *Consider a design $\mathbf{d}$ and path with ξ-clearance $\sigma : [0, 1] \to \mathcal{Q}^{\mathbf{d}}_{free}$, with $\sigma(1) = \mathbf{q}_{goal} \in \mathcal{Q}^{\mathbf{d}}_{free}$. Then, under any design $\mathbf{d}'$ that satisfies*

$$\sup_{t \in [0,1]} ||\text{Shape}(\mathbf{d}, \sigma(t)) - \text{Shape}(\mathbf{d}', \sigma(t))||_\infty < \xi,$$

the robot can execute the same path σ to the goal configuration $\mathbf{q}_{goal}$ without colliding with the obstacles.

Proof: By the triangle inequality, for all $\mathbf{q} \in \sigma([0, 1])$:

$$\begin{aligned} \xi &\leq \inf_{\mathbf{o} \in \mathcal{O}} ||\text{Shape}(\mathbf{d}, \mathbf{q}) - \mathbf{o}||_\infty \\ &\leq ||\text{Shape}(\mathbf{d}, \mathbf{q}) - \text{Shape}(\mathbf{d}', \mathbf{q})||_\infty + \\ &\qquad\qquad \inf_{\mathbf{o} \in \mathcal{O}} ||\text{Shape}(\mathbf{d}', \mathbf{q}) - \mathbf{o}||_\infty \\ &< \xi + \inf_{\mathbf{o} \in \mathcal{O}} ||\text{Shape}(\mathbf{d}', \mathbf{q}) - \mathbf{o}||_\infty, \end{aligned}$$

which implies that $\inf_{\mathbf{o} \in \mathcal{O}} ||\text{Shape}(\mathbf{d}', \mathbf{q}) - \mathbf{o}||_\infty > 0$. ■

Let $R^* = \max_{\mathbf{d} \in \mathcal{D}} R(\mathbf{d})$ denote the optimal objective value and let $\mathcal{D}^* = \{\mathbf{d} \in \mathcal{D} \mid R(\mathbf{d}) = R^*\}$ be the optimal set of designs with respect to the problem in equation 2. The following lemmas establish that designs from the optimal design set will be sampled infinitely often.

Lemma 3 (Positive Measure of Optimal Designs)**.** *The set of optimal designs, $\mathcal{D}^* \subseteq \mathcal{D}$, has strictly non-zero Lebesgue measure, i.e., $\mu(\mathcal{D}^*) \in \mathbb{R}_+$.*

Proof: Consider an arbitrary optimal design $\mathbf{d}^* \in \mathcal{D}$ that reaches the set of goal regions $\mathcal{G}^* \subseteq \mathcal{G}$, where $R(\mathbf{d}^*) = |\mathcal{G}^*|/|\mathcal{G}| = R^*$. By Lemma 1, this implies that for each goal region $g \in \mathcal{G}^*$, there exists a path with ξ_g-clearance, $\sigma : [0, 1] \to \mathcal{Q}^{\mathbf{d}^*}_{\text{free}}$ for some $\xi_g > 0$, with $\sigma(1) = \mathbf{q}_{\text{goal}}$ such that $\mathbf{p}_{\text{g}} \in g$, where $\mathbf{p}_g = \text{Shape}(\mathbf{d}^*, \mathbf{q}_{\text{goal}})(1)$ denotes the position of the end-effector at configuration $\mathbf{q}_{\text{goal}}$. Since the goal region g is an open set (Assumption 1), there exists a constant $\epsilon_g \in \mathbb{R}_+$ such that $B_{\epsilon_g}(\mathbf{p}_g) \subseteq g$.

Let $\epsilon'_g = \min\{\epsilon_g, \xi_g\}$ be a strictly positive constant, for some $\xi_g > 0$ as in Lemma 1. By the continuity of the Shape function (Assumption 2), we have that Shape is uniformly continuous on the compact domain $\mathcal{D} \times \mathcal{Q}$. Thus, for the constant ϵ'_g, there exists some $\delta_g \in \mathbb{R}_+$ such that for any design $\mathbf{d} \in \mathcal{D}'$, it follows that

$$||\text{Shape}(\mathbf{d}, \mathbf{q}) - \text{Shape}(\mathbf{d}^*, \mathbf{q})||_\infty < \epsilon'_g \quad \forall \mathbf{q} \in \sigma([0, 1]),$$

where $\mathcal{D}' = \{\mathbf{d}' \in \mathcal{D} \mid ||\mathbf{d}' - \mathbf{d}^*||_\infty < \delta_g\}$.

Lemma 2 implies that all designs $\mathbf{d} \in \mathcal{D}'$ can traverse the path $\sigma : [0, 1] \to \mathcal{Q}^{\mathbf{d}}_{\text{free}}$ to reach $\mathbf{q}_{\text{goal}}$ without colliding with the obstacles. Moreover, by definition of the supremum norm, the end-effector of the robot under design $\mathbf{d}$ at configuration $\mathbf{q}_{\text{goal}}$ is fully contained in goal region g. Thus, we have shown the existence of a non-empty, open set of designs $\mathcal{D}'$ that can reach goal region g by a continuous, collision-free path.

Following the same line of reasoning, there exist strictly positive constants ϵ_g and $\epsilon'_g = \min\{\epsilon_g, \xi_g\}$ for each goal region $g \in \mathcal{G}^*$. Since $\mathcal{G}^*$ is a finite set, let $\epsilon = \min_{g \in \mathcal{G}^*} \epsilon'_g$ be a strictly positive constant. It follows by generalization of the previous argument for a single goal region that there exists a non-empty, open set of optimal designs, $\mathcal{D}'' = \{\mathbf{d}'' \in \mathcal{D} \mid ||\mathbf{d}'' - \mathbf{d}^*||_\infty < \delta\}$, for some $\delta \in \mathbb{R}_+$, that is, $R(\mathbf{d}) = R^*$ for all designs $\mathbf{d} \in \mathcal{D}''$. Lebesgue measure is strictly positive on non-empty, open sets, thus $\mu(\mathcal{D}'') \in \mathbb{R}_+$. Since $\mathcal{D}'' \subseteq \mathcal{D}^*$ by definition, it follows that $\mu(\mathcal{D}^*) \in \mathbb{R}_+$. ■

Lemma 4 (Frequency of Sampling Optimal Designs)**.** *Alg. 2 will sample designs from the optimal design set $\mathcal{D}^*$ infinitely often.*

Proof: It is known that designs that are an element of any subset $\mathcal{D}' \subseteq \mathcal{D}$ with non-zero measure will be sampled infinitely often by the ASA algorithm [13, 20]. Thus, Lemma 3 yields the result. ■

C. Asymptotic Optimality

Let $\mathcal{Y}_k$ be a random variable that denotes the maximum reachable goal percentage attained over all the designs sampled in optimization iterations $1, \ldots, k$.

Theorem 5 (Asymptotic Optimality)**.** *The solution generated by Alg. 2 almost surely converges to a globally optimal design $\mathbf{d}^* \in D^*$, i.e.,*

$$\mathbb{P}\left(\lim_{k\to\infty} \mathcal{Y}_k = R^*\right) = 1.$$

Proof: Application of Lemma 4 implies that the optimal set of designs $\mathcal{D}^* \subset \mathcal{D}$ will be sampled infinitely often. Let $j \in \mathbb{N}_+$ denote the j^{th} occurrence of sampling any arbitrary optimal design $\mathbf{d}^* \in \mathcal{D}^*$ and let I_j denote the number of iterations that the RRT algorithm is executed for. Note that by the procedure used to increase the number of RRT iterations by $i_\Delta \in \mathbb{N}_+$ (Alg. 2) after each sampled design, we have that $I_j + 1 \leq I_{j+1}$ for all j and that $1 \leq I_1$.

For each occurrence of sampling an optimal design $\mathbf{d}^* \in \mathcal{D}$, a random approximation of the reachable goal percentage is generated by running the RRT algorithm for I_j iterations. Let $\hat{\mathcal{G}}_j(\mathbf{d}^*)$ and $\hat{R}_j(\mathbf{d}^*) = \frac{|\hat{\mathcal{G}}_j(\mathbf{d}^*)|}{|\mathcal{G}|}$ denote the approximation of GoalRegionsReachable$(\mathbf{d}^*)$ and $R(\mathbf{d}^*)$ for the j^{th} sampled optimal design respectively. For any arbitrary $\epsilon \in \mathbb{R}_+$, let A_j denote the event $|\hat{R}_j(\mathbf{d}^*) - R^*| \geq \epsilon$ for each j. Note that event A_j is equivalent to the event that the RRT algorithm fails to find a collision-free path to at least one goal region $g \in \mathcal{G}^* \setminus \hat{\mathcal{G}}_j(\mathbf{d}^*)$ within I_j iterations. Thus, we have

$$\mathbb{P}(A_j) = \mathbb{P}(\exists g \in \mathcal{G}^* \setminus \hat{\mathcal{G}}_j(\mathbf{d}^*)) \leq \sum_{g\in\mathcal{G}^*} \mathbb{P}(g \in \mathcal{G}^* \setminus \hat{\mathcal{G}}_j(\mathbf{d}^*))$$
$$\leq \sum_{g\in\mathcal{G}^*} ae^{-bI_j} = |\mathcal{G}^*|ae^{-bI_j},$$

for some constants a, b, where the first inequality is by the union bound and the second by RRT's exponential decay of the probability of failure to find a path after I_j iterations [15, 18].

Consider the sum of the probabilities of A_j over all j:

$$\sum_{j=1}^{\infty} \mathbb{P}(A_j) \leq \sum_{j=1}^{\infty} |\mathcal{G}^*|ae^{-bI_j} \leq \sum_{j=1}^{\infty} |\mathcal{G}^*|ae^{-bj}$$
$$= \frac{|\mathcal{G}^*|a}{e^b - 1} < \infty.$$

By the Borel-Cantelli Lemma we have that $\mathbb{P}(\limsup_{j\to\infty} A_j) = 0$, that is, the probability that A_j occurs infinitely often is 0. This implies that $\mathbb{P}(\liminf_{j\to\infty} |\hat{R}_j(\mathbf{d}^*) - R^*| < \epsilon) = 1$, which is precisely the definition of $\hat{R}_j(\mathbf{d}^*) \overset{a.s.}{\to} R^*$.

Thus, with probability 1, at least one optimal design $\mathbf{d}^* \in \mathcal{D}^*$ will be sampled and evaluated accurately as the number of optimization iterations of Alg. 2 approaches infinity. Since the best solution found thus far is cached in Alg. 2, it follows that $\mathcal{Y}_k \overset{a.s.}{\to} R^*$. ■

VI. Results

We apply our design optimization algorithm to two distinct piecewise cylindrical robots: (i) a serial manipulator and (ii) a concentric tube robot, a tentacle-like robot designed for minimally invasive medical procedures. We assess the performance of our method (ASA+MP) in computing designs with high reachable goal percentage and compare its computational efficiency and results with the following variants of our method and other state-of-the-art design optimization algorithms.

- *NM+G*: The Nelder-Mead optimization algorithm is used instead of ASA for sampling designs. For evaluation of the reachable goal percentage, a grid-based approach is used instead of motion planning; the configuration space is discretized into a grid and the robot configuration at each grid point is evaluated to determine if it is collision-free and reaches a goal region [3].
- *NM+MP*: Nelder-Mead is used for optimizing the design. In contrast to prior work using Nelder-Mead [3], we use motion planning using the same number of initial and additional RRT iterations as our algorithm to approximate the reachable goal percentage of candidate designs.
- *ASA+G*: In this simplified form of our approach, we use ASA to sample designs, but we use the grid approach (as described in NM+G) to evaluate reachable goal percentage for a candidate design.
- *RRT of RRTs*: An RRT-based algorithm is run both in the design space [31] and in the configuration space for estimating reachable goal percentage.

We emphasize that the grid-based algorithms (ASA+G and NM+G) only consider final configurations when evaluating reachable goal percentage during design optimization. This implies that grid-based evaluations generate *upper bounds* on the ground-truth reachable goal percentage, since the actual motion of the device from its start configuration to a goal region is not considered, and no motion may be feasible due to obstacles. In our results graphs, we do not display this upper bound; instead, we run a post-processing step (that is not counted towards method computation time) and estimate the reachable goal percentage of each returned design by running the RRT algorithm for 300,000 iterations.

We implemented all design optimization algorithms in C++. The experiments were conducted on a PC with two 2.40 GHz Intel Xeon E5620 processors (8 cores total) and 12 GB RAM.

A. Design Optimization of a 4-link Serial Manipulator

We consider the design optimization of a serial manipulator with 4 revolute joints and 4 straight links operating in a 2D environment. The configuration space of the robot is defined

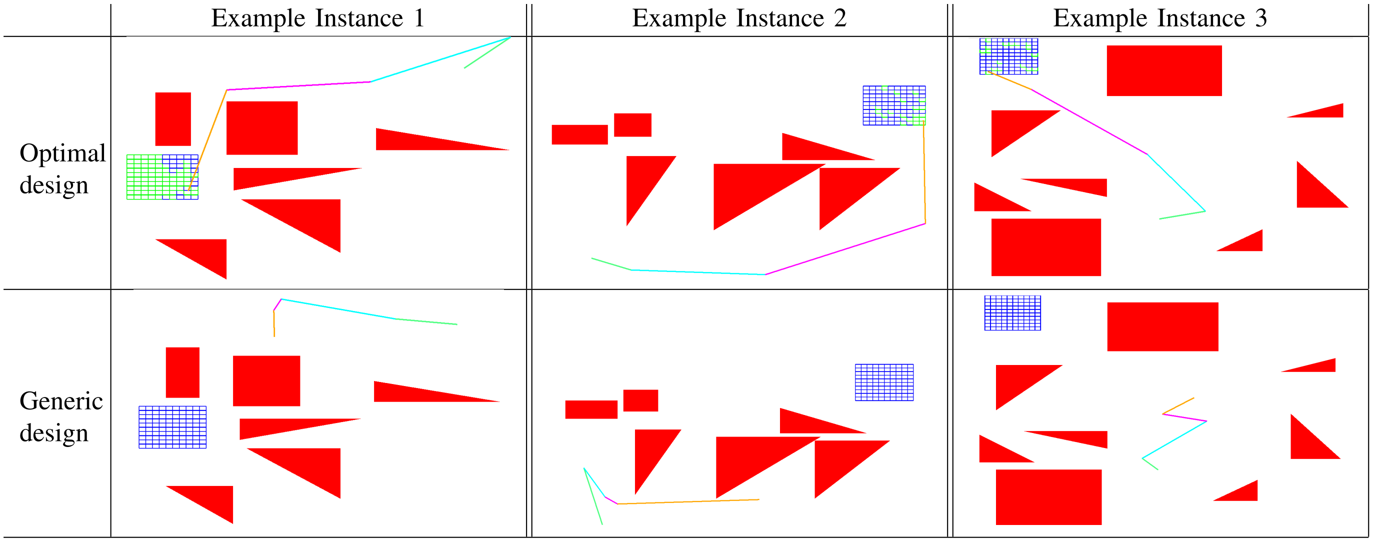

Fig. 2. First row: Example configurations of robot designs (where the links are colored green, cyan, magenta, and orange) computed by our algorithm for three randomly generated 2D environments containing obstacles (red) and a grid of goal regions colored green for grid cells reachable using the optimal design and blue for unreachable cells. Second row: In contrast to optimal designs, generic (i.e., randomly generated) robot designs operating in the same three scenarios are unable to reach the goal regions.

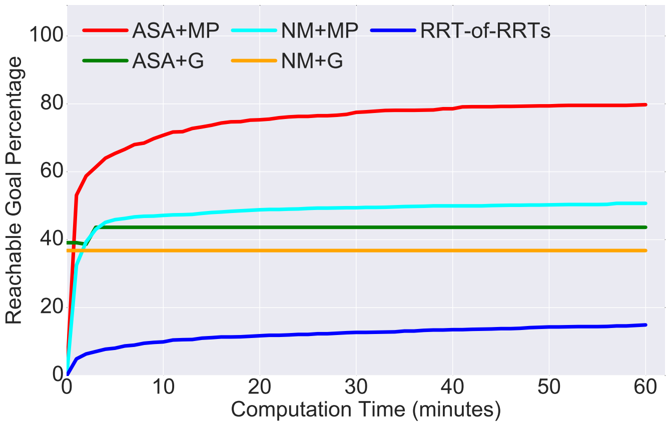

Fig. 3. The performance over time of the design optimization methods for a 4-link serial manipulator. The plot shows the reachable goal percentage of the best design found thus far with respect to computation time, averaged over 40 randomized problem instances.

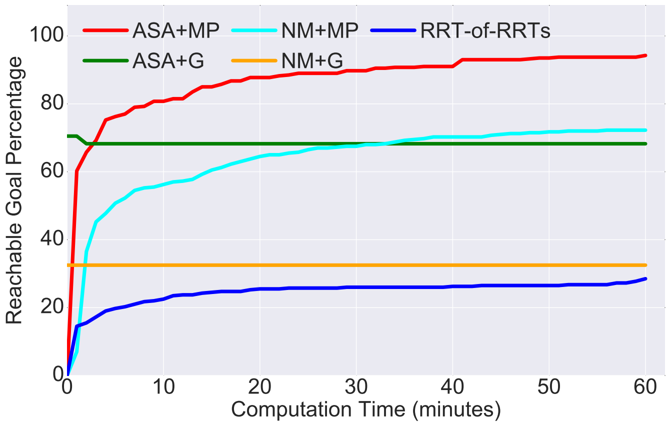

Fig. 4. Plot of the reachable goal percentage over computation time for each design optimization method run 10 times for the problem instance in Fig. 2 (right column).

by the angles of the four links, i.e., $\mathcal{Q} \subset (\mathcal{S}^1)^4$. We define a robot's design space as the length of each of the four links, thus, $\mathcal{D} \subset \mathbb{R}^4$. We evaluated each design optimization method on 40 randomized problem instances. For each instance, we randomly generated between 4 and 8 rectangular or right triangular obstacles with sides of random length and a set of 100 goal regions arranged in a regular grid and placed randomly in the workspace. The robot's start configuration $\mathbf{q}_0$ was fixed as $\mathbf{0}$ for all instances and the robot's base position was randomly placed so that the robot was collision-free. Fig. 2 depicts three examples of the problem instances.

Fig. 3 shows the reachable goal percentage (averaged over the 40 problem instances) achieved by each design optimization algorithm as a function of computation time. The robot design generated by our algorithm is capable of reaching a significantly higher percentage of the goal regions compared to the designs found by the other algorithms.

Fig. 4 depicts the performance of each design optimization algorithm for a single scenario, specifically Example Instance 3 in Fig. 2. Each line is an average over 10 runs of the corresponding algorithm. We note the methods that use grid-based evaluation of the objective function are not guaranteed to improve over time since ignoring motion planning implies they are optimizing a potentially incorrect approximation of the objective function. Our method improves the design in an asymptotically optimal manner.

Our results indicate that our approach for blending ASA for searching the design space and motion planning for design evaluation helps in attaining computational efficiency and escapes local optima via asymptotic optimality.

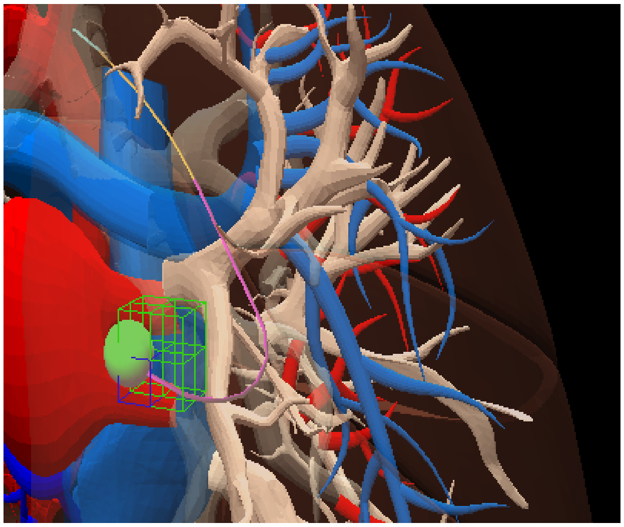

Fig. 5. A concentric tube robot (composed of cyan, yellow, and magenta tubes) has the potential to reach clinical goal regions (green and blue voxels) within the lung for early-stage lung cancer diagnosis. The figure shows the robot with an appropriate design reaching a point (green sphere) in one of the goal regions (shown as blue).

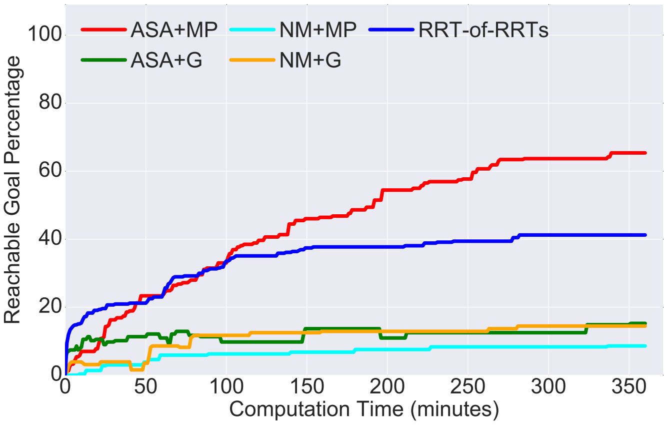

Fig. 6. The reachable goal percentage over computation time for the concentric tube robot scenario. The results are averaged across 40 different problem instances, each with randomly selected goal regions in the right lung.

B. Design Optimization of a 3-tube Concentric Tube Robot

We next apply our design optimization algorithm to a concentric tube robot [8, 11], a medical robot composed of nested nitinol tubes that can be rotated and translated independently to change the shape of the entire robot and achieve tentacle-like motion. Unlike traditional medical instruments that are constrained to straight-line paths, these robots are capable of curving around anatomical obstacles, e.g., blood vessels, to reach clinical targets in a safe, minimally-invasive manner.

We consider in simulation a concentric tube robot with 3 tubes. In configuration space, each tube adds two degrees of freedom (since each tube can be independently inserted and rotated), resulting in a 6 dimensional configuration space, i.e., $\mathcal{Q} \subset (\mathcal{S}^1)^3 \times \mathbb{R}^3$. The curvilinear shapes that the robot can achieve are highly dependent on the physical specifications of the robot's tubes, i.e., its design. In this study, each tube of the concentric tube robot is described by (i) the length of its straight section, (ii) the length of its pre-curved section, and (iii) the curvature of its pre-curved section. Thus, for our 3-tube robot the design space is 9 dimensional, i.e., $\mathcal{D} \subset \mathbb{R}^9$. To evaluate the robot's shape given its configuration, we use an accurate mechanics-based kinematic model to account for the elastic and torsional interactions between the tubes [28].

Fig. 5 illustrates a potential medical application of these devices for biopsy of suspicious nodules in the lung for early-stage lung cancer diagnosis. 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 a clinical target for biopsy. We discretized the interior volume of the right human lung into 4156 equally-sized cubic voxels each with volume ≈ 0.7 cm^3. For each trial, a subset of 8 contiguous voxels (i.e., goal regions) was randomly chosen to represent subregions of a clinical target that should be biopsied.

The results averaged over 40 trials are shown in Fig. 6. The results for this scenario follow a similar trend as the results obtained from the 4-link serial manipulator scenario. In particular, the figure illustrates our algorithm's effectiveness in finding a design with high reachable goal percentage and its tendency to efficiently improve the solution over the allotted computation time without being trapped in local optima.

VII. Conclusion

We presented a design optimization algorithm applicable to any piecewise cylindrical robot. The algorithm integrates a sampling-based motion planner in the configuration space with stochastic search in the design space to efficiently compute designs that maximize reachability to user-specified goal regions in the workspace. We proved the asymptotic optimality of our algorithm and demonstrated its computational efficiency in simulated scenarios involving serial manipulators and concentric tube robots for medical procedures.

In future work, we plan to consider a mixture of continuous and discrete design parameters and generalize our definition of goal regions to consider goal configurations and end effector poses. We also plan to physically implement the designs computed by our method and conduct experiments in testbeds based on clinically-relevant scenarios, such as lung biopsies and neurosurgery. Since the shape-set or 3D-printed robots may not precisely match our method's output, we plan to consider design uncertainty in design optimization. We also conjecture that our method and analysis can be extended beyond piecewise cylindrical robots.

Acknowledgments

We thank Robert J. Webster III for valuable discussions, Chris Bowen for his suggestions regarding the analysis, and Luis G. Torres for his insights into design optimization and software for evaluating designs. This research was supported in part by the National Science Foundation under award IIS-1149965 and by the National Institutes of Health under awards R01EB017467 and R21EB017952.

References

[1] Cenk Baykal, Luis G Torres, and Ron Alterovitz. Optimizing design parameters for sets of concentric tube robots using sampling-based motion planning. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 4381–4387, 2015.

[2] Christos Bergeles, Andrew H Gosline, Nikolay V Vasilyev, Patrick Codd, Pedro J del Nido, and Pierre E. Dupont. Concentric tube robot design and optimization based on task and anatomical constraints. *IEEE Trans. Robotics*, 31(1):67–84, 2015.

[3] Jessica Burgner, Hunter B Gilbert, and Robert J. Webster III. On the computational design of concentric tube robots: Incorporating volume-based objectives. In *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, pages 1185–1190, 2013.

[4] Jessica Burgner-Kahrs. Task-specific design of tubular continuum robots for surgical applications. In *Soft Robotics*, pages 222–230. Springer, 2015.

[5] Jessica Burgner-Kahrs, Hunter B Gilbert, Josephine Granna, Philip J Swaney, and Robert J. Webster III. Workspace characterization for concentric tube continuum robots. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, pages 1269–1275, 2014.

[6] Olivier Chocron. Evolutionary design of modular robotic arms. *Robotica*, 2008.

[7] Laurent Denarie, Kevin Molloy, Marc Vaisset, Thierry Siméon, and Juan Cortés. Combining system design and path planning. In *Proc. Workshop on the Algorithmic Foundations of Robotics (WAFR)*, 2016.

[8] Pierre E. Dupont, Jesse Lock, Brandon Itkowitz, and Evan Butler. Design and control of concentric-tube robots. *IEEE Trans. Robotics*, 26(2):209–225, 2010.

[9] Rick Durrett. *Probability: theory and examples*. Cambridge university press, 2010.

[10] Hunter B. Gilbert and Robert J. Webster III. Rapid, reliable shape setting of superelastic nitinol for prototyping robots. *IEEE Robotics and Automation Letters*, 1(1):98–105, 2016.

[11] Hunter B. Gilbert, D. Caleb Rucker, and Robert J. Webster III. Concentric tube robots: The state of the art and future directions. In *Int. Symp. Robotics Research (ISRR)*, 2013.

[12] Junhyoung Ha, Frank C Park, and Pierre E Dupont. Achieving elastic stability of concentric tube robots through optimization of tube precurvature. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, pages 864–870, 2014.

[13] Lester Ingber. Very fast simulated re-annealing. *Mathematical and Computer Modelling*, 12(8):967–973, 1989.

[14] Gangyuan Jing, Tarik Tosun, Mark Yim, and Hadas Kress-Gazit. An end-to-end system for accomplishing tasks with modular robots. In *Proc. Robotics: Science and Systems (RSS)*, 2016.

[15] Sertac Karaman and Emilio Frazzoli. Sampling-based algorithms for optimal motion planning. *Int. J. Robotics Research*, 30(7):846–894, 2011.

[16] Lalitesh K. Katragadda. *A Language and Framework for Robotic Design*. PhD thesis, Carnegie Mellon University, 1997.

[17] Steven M. LaValle. *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006.

[18] Steven M LaValle and James J Kuffner. Randomized kinodynamic planning. *Int. J. Robotics Research*, 20(5): 378–400, 2001.

[19] Chris Leger. *Automated Synthesis and Optimization of Robot Configurations: An Evolutionary Approach*. PhD thesis, Carnegie Mellon University, 1999.

[20] Marco Locatelli. Simulated annealing algorithms for continuous global optimization. In *Handbook of Global Optimization*, pages 179–229. Springer, 2002.

[21] Jean-Pierre Merlet. Optimal design of robots. In *Proc. Robotics: Science and Systems (RSS)*, 2005.

[22] Tania K Morimoto and Allison M Okamura. Design of 3-d printed concentric tube robots. *IEEE Trans. Robotics*, 32(6):1419–1430, 2016.

[23] Tania K Morimoto, Joseph D Greer, Michael H Hsieh, and Allison M Okamura. Surgeon design interface for patient-specific concentric tube robots. In *Proc. IEEE Int. Conf. on Biomedical Robotics and Biomechatronics (BioRob)*, pages 41–48, 2016.

[24] Yash Mulgaonkar, Brandon Araki, Je-sung Koh, Luis Guerrero-Bonilla, Daniel M Aukes, Anurag Makineni, Michael T Tolley, Daniela Rus, Robert J Wood, and Vijay Kumar. The flying monkey: a mesoscale robot that can run, fly, and grasp. In *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, pages 4672–4679, 2016.

[25] J. Park, P. Chang, and J. Yang. Task-oriented design of robot kinematics using the grid method. *Int. J. Advanced Robotics*, 17(9):879–907, 2003.

[26] Mark M Plecnik, Duncan W Haldane, Justin K Yim, and Ronald S Fearing. Design exploration and kinematic tuning of a power modulating jumping monopod. *Journal of Mechanisms and Robotics*, 9(1):011009, 2017.

[27] John H. Reif. Complexity of the mover's problem and generalizations. In *20th Annual IEEE Symp. on Foundations of Computer Science*, pages 421–427, 1979.

[28] Daniel Caleb Rucker. *The mechanics of continuum robots: model-based sensing and control*. PhD thesis, Vanderbilt University, 2011.

[29] Daniela Rus and Michael T Tolley. Design, fabrication and control of soft robots. *Nature*, 521(7553):467–475, 2015.

[30] Damien Salle, Philippe Bidaud, and Guillaume Morel. 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, 2004.

[31] Luis G Torres, Robert J. Webster III, and Ron Alterovitz. 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, 2012.

[32] R. Vijaykumar, K. J. Waldron, and M. J. Tsai. Geometric optimization of serial chain manipulator structures for working volume and dexterity. *Int. J. Robotics Research*, 5(2):91–103, 1986.

[33] Cornell Wright, Austin Buchan, Ben Brown, Jason Geist, Michael Schwerin, David Rollinson, Matthew Tesch, and Howie Choset. Design and architecture of the unified modular snake robot. In *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, pages 4347–4354, 2012.