IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier

Motion Planning of Planar Closed Chains Based on Structural Sets

GUANFENG LIU[1], JEFF TRINKLE[2], YONG YANG[1], AND SHAOMING LUO[1]
[1]School of Mechatronics Engineering, Guangdong Polytechnic Normal University, China, 510665.
[2]Department of Computer Science, Lehigh University, USA.

Corresponding author: Yong Yang (e-mail: yy2008@gpnu.edu.cn).

This work was supported by Provincial Natural Science Foundation of Guangdong under Grant 2015A030313662 and 2015A030308011, and the Frontier and Key Technology Innovation Special Funds of Guangdong under Grant 2017B090910008.

ABSTRACT This paper investigates the motion planning problem of planar m-link $(m \geq 4)$ closed chains among point obstacles with extension to arbitrary convex 2-D obstacles. The configuration space (C-space) of closed chains is embedded into two copies of m-3 dimensional tori. Two structural sets, the C-boundaries and the C-obstacles, are analyzed based upon the C-spaces of recursively constructed lower-dimensional closed chains. They contain essential structural information about the connectivity of the collision-free portion (C-free) of the C-space. By approximating each workspace obstacle by a set of points on the boundary after dilation, its corresponding C-obstacle is guaranteed to be covered by the C-obstacle of the convex hull of the point set. This permits a resolution-complete roadmap algorithm that puts specific bias for sampling the structural sets. Several benchmark examples are presented that compare the performance between our algorithm and the traditional algorithms. Animation videos and source codes are also provided which demonstrate the effectiveness of our method for closed chains of up to 20 links.

INDEX TERMS Path planning, closed chains, boundary variety, bifurcation, narrow passages, sampling algorithm

I. INTRODUCTION

Closed-chain manipulators and mechanisms have recently received lots of interests because of their potential advantages over their serial counterparts. They have appeared in many application domains, including parallel machining tools, multipod robots, spatial robotic arms, and humanoids. However, their closed-loop structure gives rise to joint variable dependencies, which manifest in a topologically complex configuration space. An important consequence of this is that, in general, the C-space cannot be globally (and smoothly) parameterized by a single set of d variables (for example a subset of the joint displacements), where d is the degrees of freedom (DoFs) of the manipulator. In other words, any d-dimensional atlas of the C-space will contain multiple charts. This fact generally makes motion planning more challenging for closed-chain manipulators than it is for serial manipulators.

A. PREVIOUS WORK

It is well known that general exact motion planning algorithms for serial manipulators are highly complex [1]–[4], [10]. In fact, the most efficient exact planning algorithm is Canny’s, whose worst-case time complexity is exponential in the dimension of C-space [2]. In principle, exact algorithms can be applied to systems with holonomic equality constraints such as those imposed by the kinematic loops in closed-chain manipulators “by defining convenient charts and managing them" (see [3], page 411). However, the difficulty of implementing exact algorithms for general systems fueled a paradigm shift to sampling-based algorithms, in which the probabilistic roadmap (PRM) algorithm and the rapidly-exploring random tree (RRT) algorithm are the two most widely adopted methods [5]–[7].

Sampling-based algorithms build a graph that approximates the global structure of C-free. The graph has nodes that correspond to selected points of C-free and arcs between nodes that indicate path connectedness between the corresponding points. The graph can be thought of a network of highways, or a roadmap, of C-free. The roadmap becomes suitable for motion planning when the following two attributes are attained: (1) there is a one-to-one correspondence between components of the graph and those of C-free, and (2) given an arbitrary point in C-free, it is easy to find a path connecting it to the graph. At this point, motion planning is essentially reduced to graph searching.

Sampling-based algorithms have been quite successful for systems whose C-space can be parameterized by a single chart with number of coordinates equal to the DoFs of the system, but less successful otherwise [8], [11], [23]. Even though one can always generate an ambient space parametrizable by a single chart by choosing more parameters than the dimension of C-space, the number of sample points needed to construct a good roadmap for C-space submanifolds grows significantly (if not exponentially) with the dimension of the ambient space

To overcome this difficulty, one needs an efficient algorithm that generates well distributed samples on the C-space submanifold, and a fast local planner for joining pairs of samples for constructing the roadmap. Two different strategies have been proposed to solve the challenge. The first one [11] breaks a closed chain into several open chains, and then randomly generates the configurations for one of the chains (called master chain), and those of the remaining chains (called slave chain) by inverse kinematics (IK). This method guarantees that the generated samples precisely stay on the submanifold, but is subject to the draw back of significant failure rate in IK because a large portion of the workspace of the master chain might not be reachable by the slave chains. The second route samples the ambient space first and then projects them onto the submanifold through iterations. Yakey, Lavalle, and Kavraki [8] proposed a randomized gradient decent algorithm (RGD) for closing the loop gap. However, this algorithm relies on parameter tuning and might take long time to converge for some samples in the ambient space. In [21] it has been argued that the RGD algorithm is not as efficient as Jacobian pseudo-inverse method, which is widely used for manipulation planning on constraint manifolds [23], [24] and protein loop-closure problem [26]. In [15] Kim et al. observed that projections of points in a neighborhood of the ambient space do not always create new samples on the submanifold due to the curved structure of C-space. They devised a different algorithm that samples the tangent bundle of the C-space submanifold and uses the heuristics of C-space curvature for determining the sampling radius on each tangent space. As the result the number of projections is highly reduced. However, as the computation of tangent spaces and local curvature relies on the local parametrization, the difficulties of managing different charts remain. Moreover projections are still required to find a sequence of intermediate tangent spaces between start and goal configurations, and to transform the collection of milestones on these tangent spaces into the final path on the C-space submanifold. Similar ideas have been adopted by Porta and Jaillet [30], [31] who proposed an atlas-based RRT algorithm which approximates the manifold using a sequence of coordinate charts, and generates samples directly from the charts for expanding a random tree. This method does not build a complete atlas, but only a subset guided by the RRT itself. However, this algorithm might require many coordinate charts to achieve smooth covering of the manifold, and extra computation time spending on high-dimensional continuation between neighbor charts and projections between the charts and the C-space submanifold.

When a C-space manifold has to be covered by multiple coordinate charts, there exists singular loci for every single coordinate chart [17]–[19] which are crucial for establishing the adjacency between different charts. Traditional sampling-based approach will often predict wrong results if they are not properly handled. Although the random loop generator (RLG) method [9] improves the successful rate of the IK based sampling algorithms by estimating the range of each joint in the linkage that obeys the loop closure constraint, the algorithm might fail to sample some critical regions in C-space, e.g., the singular loci. The atlas-based RRT [30] applies a parametrization continuation method which essentially computes the overlapping area between neighboring charts, and then creates a linear boundary that bridges the two charts based upon artificially designed constraints. However, this continuation method only applies to single-query problems and can be time consuming if a chart is adjacent to many others.

When robot workspace contains multiple obstacles, narrow passages might develop around C-obstacles. Geometrically these are local subsets of C-free whose thickness along a codimension-1 hyper-surface is less than a small positive number $\epsilon > 0$. Narrow passages pose great challenges to most of the traditional algorithms. The most successful methods to handling narrow passages are the obstacle-based PRM algorithm developed by Amato et al. [22], and the penetration-after-dilation method of Hsu et al. [25]. The former method applies the heuristics that narrow passages only exist when the distance between a pair of C-obstacles is sufficiently small and so samples in narrow passages must be close to at least one C-obstacle, while the latter method dilates C-free by shrinking workspace obstacles or robot links, and then identifies the possible penetrations between original links and workspace obstacles. In general the former method generates more samples around C-obstacles and is suited for multi-query problems, while the latter one is only applicable to single-query problems and might fail to give a solution if penetrations can not be eliminated by local resampling. Methods for handling both narrow passages and closed-loop constraints have not been seen in any literature.

The difficulties associated with applying sampling-based motion planning methods to closed-chain manipulators and the availability of new results in topology led to renewed interest in exact planning algorithms for closed kinematic chains (see Fig. 1) [12]–[14]. Trinkle and Milgram derived some global topological properties of the C-space (the number of components and the structures of the components) of single-loop closed chains with spherical joints in a workspace without obstacles [12]. They show that the C-space is the union of manifolds that are products of spheres and intervals. An efficient path planning algorithm can be carried out by iteratively moving those joints corresponding to the spheres to their goals, while the remaining links follow accordingly

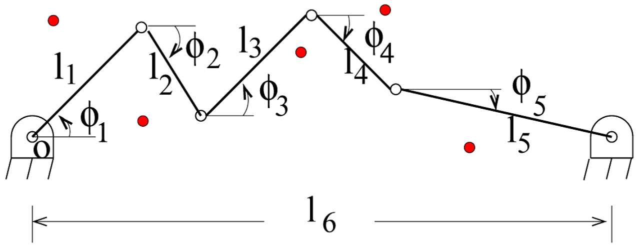

FIGURE 1: A closed 6-chain among point obstacles (shown as small discs).

so that the loop can be closed.

B. CONTRIBUTION

In this paper, we employ the two structural sets, the C-boundaries and C-obstacles, for solving the challenges from both C-space bifurcations and narrow passages. Our innovations are summarized as the following.

1) Minimal parametrization of C-space with only two coordinate charts, each embedded in an (m-3)-dimensional tori;
2) Characterize C-space bifurcations in terms of boundary varieties;
3) Cover each workspace obstacle by the convex hull of a set of point obstacles; Identify and sample narrow passages through computing configurations for which the closed chain intersects a point obstacle in the set obtained above;
4) Sample both coordinate charts, the boundary varieties, and the narrow passages through an algorithm that precisely computes the feasible range of each individual joint angle;
5) Strong experimental evidences that demonstrate the importance of the structural sets in solving challenging motion planning problems;

II. BASIC NOTATION AND TERMINOLOGY

Imagine a planar serial chain of $m-1$ links connected by revolute joints, with one end free, and the other connected to the ground. The ground is regarded as link m and is referred to as the base of the chain. Relative to the base, the open chain has $m-1$ degrees of freedom and its C-space is simply a product of $m-1$ circles, (i.e., $(S^1)^{m-1}$). A closed m-chain can be constructed by attaching the distal end of the open chain to the base as shown in Fig. 1. Mathematically, this attachment imposes two algebraic equality constraints, which cause the C-space $\mathcal{C}$ of the closed chain to become a compact, closed, real, variety of dimension $m-3$. This variety is a manifold as long as the distance between the two anchors is not equal to one of the 2^{m-2} critical lengths [12].

III. C-SPACE OF PLANAR CLOSED CHAINS

Here we summarize two results from topological approaches to motion planning that are crucial to the work presented here. The first result is about the connectivity of $\mathcal{C}$ of a planar closed chain. We need a concept called long links [12] which is defined as a subset $\mathcal{L}$ of the links such that the sum of the lengths of every pair of distinct links in $\mathcal{L}$ is strictly greater than half of the sum of the lengths of all m links. Note that $\mathcal{L}$ may not be unique. Let $\mathcal{L}^\star$ be a set with maximal cardinality. Then the number of long links of a closed chain is defined as $|\mathcal{L}^\star|$. Due to the strict inequality in the definition, the number of long links $|\mathcal{L}^\star|$ must be 0, 2, or 3. If $|\mathcal{L}^\star|$ is equal to 3, $\mathcal{C}$ has two components; otherwise, it has one.

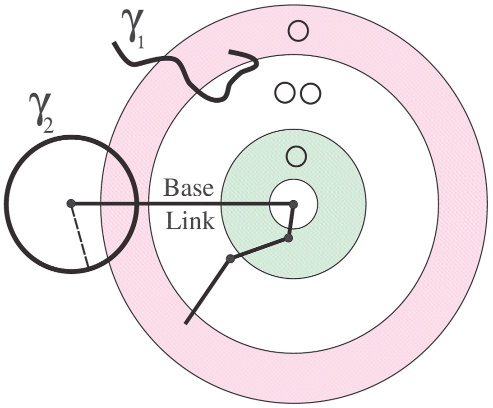

FIGURE 2: Construction of C-space of closed chains via singular circles of an open chain.

The second result gives the topology of $\mathcal{C}$. It says that for given link lengths $\{l_1, \cdots, l_{m-1}\}$ and base length l_m that is generic with respect to those $m-1$ lengths, $\mathcal{C}$ is the boundary of a manifold with boundary, which is given as the union of submanifolds of the form $(S^1)^k \times I^{m-2-k}$ [12], where I^d denotes the interval of dimension d.

To clarify the above conclusion, consider Fig. 2, which shows a horizontal base link and three moveable links anchored at the center of four concentric circles. These circles are the singular circles (not drawn to scale) of the open 3-chain. If the end point of the 3-chain is anchored at any point it can reach, its C-space $\mathcal{C}$ is that of a closed 4-chain. If the anchor point is on one of the singular circles, the three links in the chain can be arranged to be colinear and $\mathcal{C}$ is either a single point or a figure eight, while if the anchor point is interior to one of the three reachable annuli, then $\mathcal{C}$ is one circle or two disjoint circles (as indicated by the small circles at the 12 o'clock position of the concentric circles). Assume that the end point of the open 3-chain is constrained to a one-dimensional curve γ. This effectively converts the open chain into a closed chain whose C-space can be constructed by "gluing" together all the C-spaces at each point as we move along γ. For example, begin at the left end of curve γ_1 and traverse it to its other end. Initially, the C-space over each point of γ_1 is empty, since the open 3-chain cannot reach those points. At the intersection with the outer-most circle, $\mathcal{C}$ of the closed chain is a point, but the workspace segment lying inside the outer-most annular region generates a tube. At the point where γ_1 intersects the next singular circle, $\mathcal{C}$ of the closed 4-chain is a figure eight. This signifies a bifurcation of the tube into two tubes. The two tubes coalesce into a single tube at the next crossing of the same singular circle. Finally, at the end of the curve, $\mathcal{C}$ is a circle. Thus $\mathcal{C}$ of the mechanism with the end of the open 3-chain constrained to lie on γ_1 is a tube pinched closed at one end, open at the other, and with a hole through the tube somewhere between the two ends. Applying the same logic to the closed 5-chain that would result from connecting the end of the open 3-chain to points on γ_2, one finds that $\mathcal{C}$ of the closed 5-chain is a sphere. This result shows that $\mathcal{C}$ of an m-link closed chain can be obtained by gluing the C-spaces of $(m-1)$-link closed chains in a recursive manner.

IV. PARAMETRIZATION OF C-SPACE, BOUNDARY AND COLLISION VARIETIES

In this section we propose a minimal parametrization of $\mathcal{C}$ which employs only two coordinate charts. Since $\mathcal{C}$ of a generic closed m-chain is an (m-3)-dimensional manifold, it can be locally parameterized by a set of $m-3$ joint angles. However, fixing the orientations of $m-3$ links (including the fixed base angle) does not fix the configuration of the closed chain. Returning to Fig. 1, fixing ϕ_3, ϕ_4 and ϕ_5 still allows *elbow-up* and *elbow-down* postures of links 1 and 2.

This last result suggests partitioning $\mathcal{C}$ into an *elbow-up* piece and an *elbow-down* piece as follows. Break the closed chain at the third joint, thus creating an open 2-chain CH_1 with link lengths $\{l_1, l_2\}$ and an open (m-3)-chain CH_2 with link lengths $\{l_3, \cdots, l_{m-1}\}$ based at the point $(l_m, 0)$. The C-space of CH_2 is the (m-3)-dimensional torus. For an arbitrary point in this space, the chain can be closed in 0, 1, or 2 configurations of CH_1. When there are two configurations, they are labeled *elbow-up* and *elbow-down*. Since there are never more than two configurations that close the loop, two copies of the torus suffice to represent $\mathcal{C}$ of the closed chain. When there is only one configuration, the *elbow-up* and *elbow-down* configurations have converged, so at these points, the tori are connected. These configurations form a variety referred to as the *boundary variety*, which plays a key role in stitching together the C-space patches in both tori.

***Example* 1: Minimal coordinate charts of a closed 5-chain in three different cases**

Consider a closed 5-chain, with link lengths, $[1,\ 1.3,\ 4,\ 4,\ 5]^T$. There are two point obstacles in the workspace $p_1 = (5.5, 1)$ and $p_2 = (2, 2)$. The C-space of this chain has two components, since it has three long links. Obviously, we could choose any two joint angles among $(\phi_1, \cdots, \phi_4)$ as parameters and then embed the C-space into the elbow-up and elbow-down 2-D tori. However, depending on the chosen joint angle parameters, the portion of the C-space on each tori may or may not contain boundary varieties. When the C-space is parametrized by (ϕ_3, ϕ_4), it does not cover the entire elbow-up and elbow-down tori, but two separate irregular annuli each enclosed by two boundary curves as shown in Fig. 3-(b). The place and size of the C-space regions inside both torus are exactly same so only one torus is drawn here. The corresponding pair of annuli in the elbow-up and elbow-down tori are glued together along their boundaries to form a 2-D torus. In this case, the C-space is two disjoint tori, a conclusion which can also be deduced using the argument in Section III. When the C-space is parameterized by (ϕ_1, ϕ_2), we recall a result from [12] that when a closed chain has three long links, its C-space is the products of circles corresponding to the short links [1]. Therefore, the C-space covers the entire two disconnected tori without boundaries, as shown in Fig. 4-(a) and 4-(b).

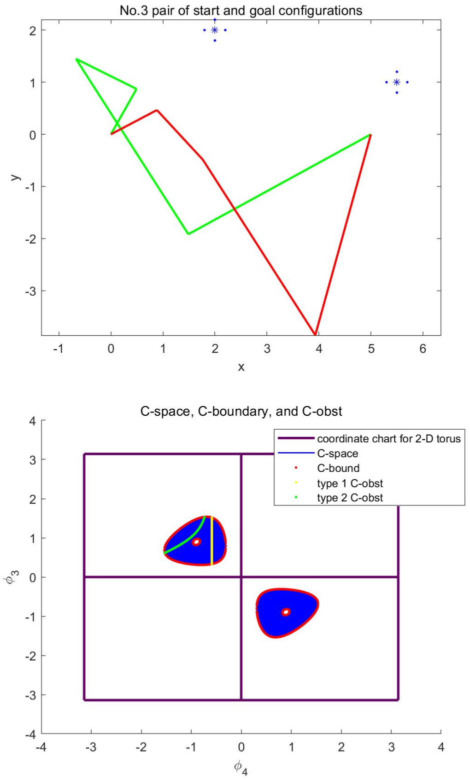

FIGURE 3: (a): Two configurations of a 5-link closed chain, 2 point obstacles, and 8 dilated point obstacles; (b): C-space regions embedded in both elbow-up and elbow-down (ϕ_3, ϕ_4) tori are exactly same, and composed of two disconnected irregular annuli (in blue) which are bounded by two curves (in red).

When a closed chain has only 0 or 2 long links, its C-space has only one component. Then there is a patch in both tori. The two patches pinch together along the boundary varieties. Consider another closed chain with link lengths $[2, 2.3, 2.35, 3.15, 5]^T$. It has 2 long links. Fig. 5-(a) and Fig. 5-(b) show the C-space patch in each torus. Fig. 6-(a) and Fig. 6-(b) show the C-space patch (in each torus) of a close chain with 0 long links, whose vector of link lengths is $[2, 2.3, 2.6, 2.9, 3]^T$.

[1] For planar closed chains with 3 long links, the parametrization using the joint angles of short links are referred to as canonical parametrization.

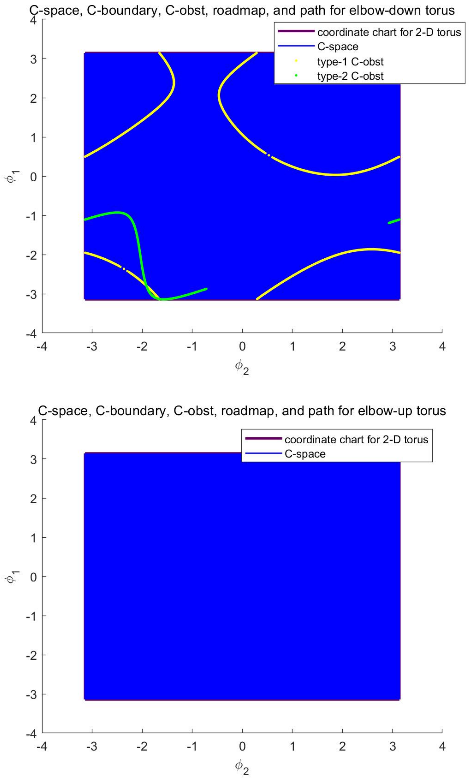

FIGURE 4: (a): C-space covers two entire (ϕ_1, ϕ_2) tori without boundaries; (b): The elbow-up (ϕ_1, ϕ_2) torus contains no collision varieties because the obstacles can only be reached by l_3 and l_4 in elbow-down configuration

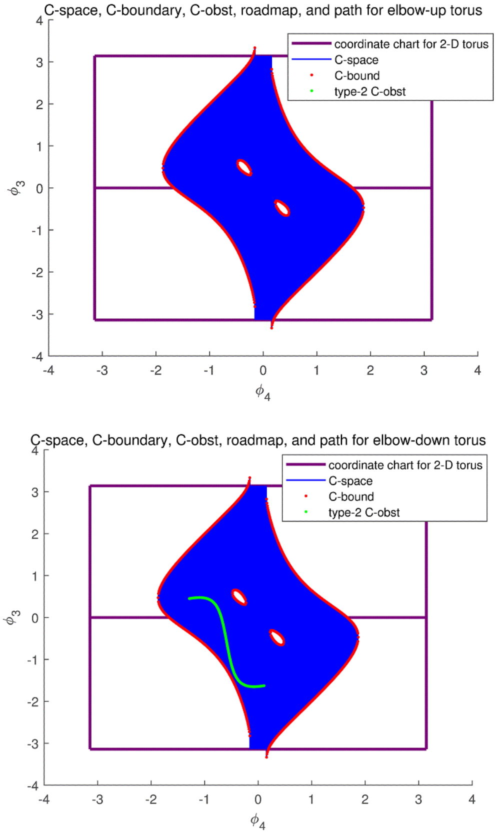

FIGURE 5: A closed chain with 2 long links: (a): C-space region in elbow-up torus; (b): C-space region in elbow-down torus

A. BOUNDARY VARIETY

Boundary variety B arises from covering $\mathcal{C}$ with multiple local coordinate charts of exactly same dimension as $\mathcal{C}$, in our case which are simply two copies of $m-3$ dimensional tori. Its size and shape depend on the chosen local parameters, as shown in Example 1. In the simplest case that the closed chain has three long links, the boundary variety becomes empty when canonical parametrization is adopted. In this subsection, we show that B is the union of C-spaces of two lower-dimensional closed chains.

Refer to Fig. 1, let us break the closed m-chain into an open 2-chain CH_1 , and an open (m-3)-chain CH_2. The boundary variety B is the set of configurations for which the endpoints of CH_1 and CH_2 can be connected when the links of CH_1 are collinear. With the constraint of collinearity, the possible end point locations Σ of CH_1 in the workspace is a pair of concentric circles of radii $l_1 + l_2$ and $|l_1 - l_2|$ centered at the origin. The boundary variety can now be defined by

$$B = f_2^{-1}(\Sigma). \tag{1}$$

Note that B is the union of the C-space of the two closed (m-1)-chains M_1 and M_2 with link lengths $\{l_1+l_2,\, l_3,\, \cdots,\, l_m\}$ and $\{|l_1-l_2|,\, l_3,\, \cdots,\, l_m\}$, respectively. If B is empty, as is the case when $\|L^\star\| = 3$, $\mathcal{C}$ is not connected.

Topologically, the unreachable portion of both tori are clipped out by B, and the remaining reachable patches, when glued together along B, forms $\mathcal{C}$.

***Example* 2: Boundary varieties for a 5-bar closed chain** As shown in Fig. 3-(b), B for the closed chain with link lengths $[1, 1.3, 4, 4, 5]^T$ are two pairs of circles (in red) on both elbow-up and elbow-down tori. Gluing the corresponding pairs of circles from both tori together yields two disconnected torus, which is the topology of the C-space of the closed chain. B for the closed chain in Fig. 5 and 6 are 4 circles on both elbow-up and elbow-down tori. Among the 4 circles, two (denoted C_1, C_2) lie in the interior, and the other two (C_3, C_4) are formed by identifying the top and bottom ends (recall they are 2π apart) of the two curve segments that enclose the blue interior. Without considering the two interior holes, the C-space patch in each torus is simply a cylinder by identifying the middle-top and middle-bottom line segments (both in blue, and are 2π apart). The two circles C_3, C_4 on both ends of the cylinder in the elbow-up torus are pinched together with those in the elbow-down torus, respectively. This yields a 2-D torus. The pairs of interior circles C_1, C_2 in the elbow-up torus, when gluing with their counterparts in the elbow-down torus, create another two holes on the previous 2-D torus. We conclude that $\mathcal{C}$ of the 5-bar closed chain is a genus-3 surface.

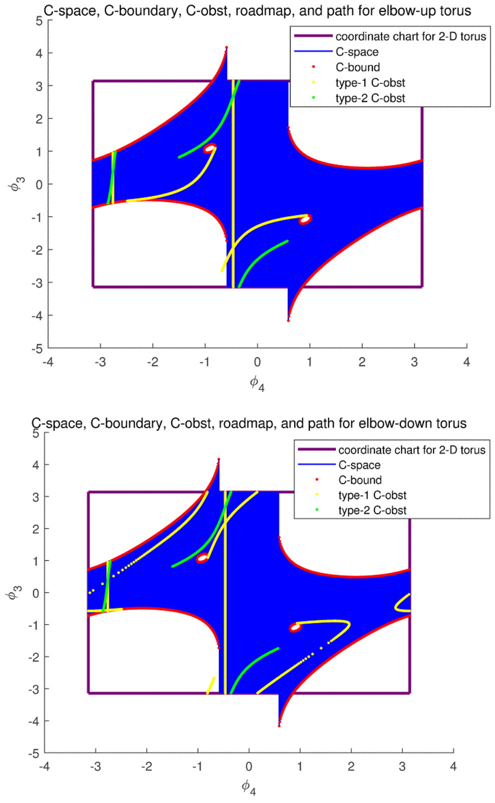

FIGURE 6: A closed chain with 0 long links: (a): C-space region in elbow-up torus; (b): C-space region in elbow-down torus

B. COLLISION VARIETIES

Consider first the case when the workspace has a set of point obstacles $\mathcal{O} = \{p_i \in \mathbb{R}^2 \mid i = 1, \cdots, n\}$. Let $V^j_{p_i}$, $j = 1, \cdots, m-1$, denote the (m-4)-dimensional variety corresponding to p_i lying on link j. The union of these varieties over all links gives the contribution of p_i to the obstacles $\mathcal{C}_{obst}$ in C-space (or simply called C-obstacles):

$$V_{p_i} = \bigcup_{j=1}^{m-1} V^j_{p_i}.$$

As shown in Fig. 7-(a), $V^j_{p_i}$ is a fiber bundle, whose base manifold and fiber are the C-space of the two closed chain CH_b, CH_f respectively. Notice that CH_b contains one prismatic joint. The following lemma eliminates the annoying prismatic joint in CH_b through the C-space of a $j+2$ fully revolute-joint closed chain, which is useful for sampling $\mathcal{C}_{free}$ with a unified random loop generator in Section VI.

Lemma 1: The C-space of CH_b in the calculation of $V^j_{p_i}$ is equivalent to that of a $(j+2)$-link closed chain with only revolute joints, whose link length vector is $(l_1, \cdots, l_{j-1}, 1/2l_j, 1/2l_j, l_b)$, where l_b is the length of the new base link connecting the origin to p_j.

Proof: See Appendix A.

Example 3: **C_{obst} of point obstacles**
We introduce two point obstacles, $p_1 = (5.5, 1)$ and $p_2 = (2, 2)$ to the example in Fig. 3. Note that it is impossible for link 1 or 2 to touch either point, so the corresponding varieties are empty. Consequently, only the contact varieties of links 3, and 4 appear in green and yellow respectively in 3-(b)[2]. If we adopt the canonical parametrization (ϕ_1, ϕ_2) then the contact varieties only appear in the elbow-down torus because the two points can only be reached by link 3 and 4 in elbow-down configuration (elbow-up configuration cannot be closed by link 1 and 2). Same point obstacles are introduced in the example in Fig. 5. While in the example in Fig. 6, the point obstacles are $p_1 = (5.5, 1)$ and $p_2 = (1, 1)$.

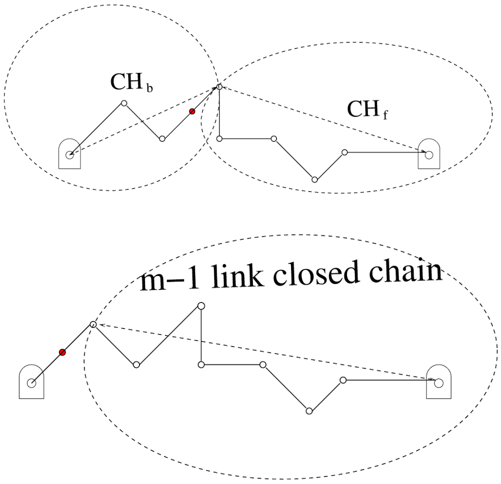

FIGURE 7: (a): $V^j_{p_i}, 1 < j < m-1$ is a fibration of the C-spaces of two closed chains. CH_b with one prismatic joint generates the base manifold and CH_f generates the fiber space for each configuration of CH_b; (b): $V^j_{p_i}$, $j = 1, m-1$, is the C-space of an $m-1$ link closed chain

V. RESOLUTION-COMPLETE ROADMAP ALGORITHM

Our analysis in the previous sections leads to the structure sets B and $\{V^j_{p_i}\}$ which permits an exact motion planning algorithm either through cell decomposition or roadmap methods. However, it is well known that these algorithms have high complexity which makes their implementation formidable. Moreover, the exact roadmap computed for the motion planning problems with point obstacles will inevitably contain segments on the boundary of C-obstacles, and is therefore not acceptable in practice. To solve this problem, we might compute a resolution complete roadmap algorithm by dilating each point obstacle in robot workspace by a factor $\epsilon > 0$ (i.e. replacing each point obstacle by a small disk of radius ϵ), and approximating each disk by a convex polygon of a set of points on the boundary circle. Here ϵ could be chosen as a positive number less than a quarter of the minimal distance among all point obstacles. The following fact is quite obvious.

[2] We use "type 1 C-obst" for $\mathcal{C}_{obst}$ by fixed-base links, and "type 2 C-obst" by free-floating links.

Proposition 1: If a given point obstacle p_i is approximated by a set of point obstacles $Q_i^\epsilon = \{q_k^i \mid q_k^i \in \mathbb{R}^2\}$ lying on the boundary after dilation by a factor ϵ such that p_i lies in the strict interior of the convex hull $CONV(Q_i^\epsilon)$ of Q_i^ϵ, then $V_{p_i}^j \subset V_{CONV(Q_i^\epsilon)}^j$, the corresponding C-obstacle for which link j intersects $CONV(Q_i^\epsilon)$.
As a result, a resolution-complete complete roadmap is readily computed.

Proposition 2: A resolution-complete roadmap for an m-link closed chain moving among $\mathcal{O} = \{p_i\}$ can be obtained from that of the same chain moving among $\mathcal{O}^\epsilon \triangleq \cup_i CONV(Q_i^\epsilon)$, by applying the collision checking between the closed chain and the strict interior of the convex hulls $CONV(Q_i^\epsilon)$.
It is obvious that as $\epsilon \to 0$, $\mathcal{O}^\epsilon \to \mathcal{O} = \{p_i\}$, and the roadmap based upon $\mathcal{O}^\epsilon$ converges to that base upon $\mathcal{O}$. Proposition 1 can be extended to the cases with arbitrary convex obstacles in the workspace.

Proposition 3: If a convex obstacle $\mathcal{O}_i \subset \mathbb{R}^2$ dilates by a factor $\epsilon > 0$, and let $Q_i^\epsilon = \{q_k^i \mid q_k^i \in \mathbb{R}^2\}$ be a set of points on the boundary after dilation such that $\mathcal{O}_i$ lies strictly in the interior of the convex hull $CONV(Q_i^\epsilon)$ of Q_i^ϵ, then the collision variety $V_{\mathcal{O}_i}^j \subset V_{CONV(Q_i^\epsilon)}^j$.

The point set $Q^\epsilon \triangleq \cup Q_i^\epsilon$ is very useful for generating milestones in $\mathcal{C}_{free}$ which are near $\mathcal{C}_{obst}$. For any configuration of a link that touches a convex hull $CONV(Q_i^\epsilon)$, there is a corresponding configuration with same orientation angle such that the link intersects a point in Q_i^ϵ. Moreover, these two configurations only differ from each other by a translation smaller than the length of an edge in $CONV(Q_i^\epsilon)$. Therefore, one can always achieve good coverage of the region near $\mathcal{C}_{obst}$ by increasing the number of points in Q_i^ϵ.

VI. SAMPLING-BASED ALGORITHM BASED ON STRUCTURAL INFORMATION

The results about the structure sets B and $\mathcal{C}_{obst}$ are useful for generating sampling points which not only stays in the C-space submanifold, but also are collectively representing the connectivity of $\mathcal{C}_{free}$ faithfully.

With our parametrization, $\mathcal{C}_{free}$ can simply be sampled by taking points in the two copies of $(S^1)^{m-3}$, while throwing away those unreachable by CH_1 and those colliding with obstacles. However, there are still two obstructions to overcome when constructing graphical representation of $\mathcal{C}_{free}$ based upon the generated milestones. One is the bifurcation of $\mathcal{C}$ itself along B. It implies that any motion joining a pair of milestones from the two different tori must cross B at least once. Instant jumping between the two tori is unavoidable if no points on B get sampled and the motion of CH_1 is calculated using IK. This jumping is already known to be dangerous and unacceptable by researchers in robot kinematics [27]–[29].

The other difficulty is the existence of possible narrow passages around $\mathcal{C}_{obst}$, which has significantly smaller thickness at least locally along a codimension-1 hyper-surface compared with the overall thickness of $\mathcal{C}$. The difficulty for sampling this region could be comparable to that on B (which is exactly codimension-1). Our overall sampling-based algorithm in pseudo code is given in Algorithm 1.

Algorithm 1 A PRM algorithm for m-link Closed Chains

```
Require: A vector of link lengths $(l_1, \cdots, l_m)$, and a set of
    convex obstacles $\mathcal{O}$ (including point obstacles as special
    case), and the start and goal configurations $\phi_{init}$ and $\phi_{goal}$
Ensure: Plan a collision-free path $\phi(t)$, $t \in [0,1]$ such that
    $\phi(0) = \phi_{init}$, and $\phi(1) = \phi_{goal}$ or report failure otherwise
    Step 1: Dilate each obstacle $\mathcal{O}_i \in \mathcal{O}$ and sample its
    boundary to obtain a set of points $Q_i^\epsilon$ such that $\mathcal{O}_i \subset CONV(Q_i^\epsilon)$
    Step 2: Randomly sample the reachable portion of both
    elbow-up and elbow-down $(m-3)$-dimensional tori
    Step 3: Randomly sample the boundary variety $B$
    Step 4: Randomly sample regions which are close to $\mathcal{C}_{obst}$
    Step 5: Check collision of obtained samples and retain
    collision-free ones
    Step 6: Repeat Steps 2-5 until the desired number of
    samples are obtained
    Step 7: Build a roadmap (a graph) by using a local planner
    to join pairs of samples which are only apart up to a given
    distance limit
    Step 8: Connect $\phi_{init}$ and $\phi_{goal}$ to vertices on the roadmap
    by the same local planner and distance limit
    Step 9: Search the graph for a sequence of milestones
    between $\phi_{init}$ and $\phi_{goal}$ or report failure
    Step 10: Turn the sequence to a path $\phi(t)$ by applying the
    local planner again
```

Algorithm 1 can be divided into three key components, a random loop generator, a sampler of regions near $\mathcal{C}_{obst}$, and a local planner with collision checking. They are in turn presented in the following subsections.

A. RANDOM LOOP GENERATOR THAT RESOLVES C-SPACE BIFURCATIONS

In this subsection we present an algorithm (Algorithm 2) that efficiently generates samples on the interior of the reachable portion of both $(m-3)$-dimensional tori as well as their boundary variety B. The major advantages of this algorithm include (1) it never samples invalid region on the tori for which the loop can not be closed; (2) the generated samples are well distributed. Algorithm 2 employs a special probability measure function $g(\phi_3, \cdots, \phi_{m-1})$ for sampling the C-space submanifold. From the way we pick ϕ_i given a fixed vector $(\phi_{i+1}, \cdots, \phi_{m-1})$, the probability measure $g(\phi_i \mid \phi_{i+1}, \cdots, \phi_{m-1})$ in sampling ϕ_i is a uniform distribution on a collection of intervals (also called a fiber) whose lengths depend on $(\phi_{i+1}, \cdots, \phi_{m-1})$. So $g(\phi_3, \cdots, \phi_{m-1})$ can be calculated recursively which goes all way back to $g(\phi_{m-1})$. Although $g(\phi_{m-1})$ is a constant whose value solely depends on the vector of link lengths, $g(\phi_i, \cdots \phi_{m-1})$ is a complex function of $(\phi_{i+1}, \cdots, \phi_{m-1})$. Furthermore, not only every

Algorithm 2 A random loop generator that samples B and the interior of $\mathcal{C}$

```
Require: A vector of link lengths $(l_1, \cdots, l_m)$, the left and right anchor points, $P_l = [0,0]^T$ and $P_r = [l_m, 0]^T$, and a positive integer $n > 0$
Ensure: Generate $n$ well distributed samples over the interior of both tori which satisfy the loop-closure constraint
Ensure: Generate $n$ well distributed samples over the boundary variety $B$
  Step 1: Initialize $i = m - 2$
  Step 2: Compute the radii of the minimal and maximal critical circles of the open chain $(l_1, \cdots, l_i)$ based at $P_l$
  Step 3: Calculate the intersections between the above two critical circles and the circle based at $P_r$ of radius $l_{(i+1)}$
  Step 4: Identify the interval on the latter circle contained in the workspace of the open chain based at $P_l$ using circle-circle intersection
  Step 5: Uniformly sample $\phi_{(i+1)}$ from these intervals and fixing it, update the coordinates of right anchor
  Step 6: Set $i = i - 1$ and repeat Step 1 to 5 until $i = 3$
  Step 7: Apply the IK of the $(l_1 + l_2, l_3)$ chain for $(\phi_1, \phi_2 = \phi_1, \phi_3)$, or the IK of the $(|1_1 - l_2|, l_3)$ chain for $(\phi_1, \phi_2 = \phi_1 + \pi, \phi_3)$; these are points on $B$
  Step 8 Repeat Step 2 to 5 for $i = 2$
  Step 9: Apply the IK of the $(l_1, l_2)$ chain for $(\phi_1, \phi_2)$; these are points on the interior of both tori
  Step 10: Repeat Step 1 to 9 until $n$ samples on both the interior and boundary are generated
```

sample from Algorithm 2 is valid, but also at each step in recursion a fiber of ϕ_i over a given set of $(\phi_{i+1}, \cdots, \phi_{m-1})$ will generate the number of samples probabilistically proportional to the overall length of the collection of intervals contained in the fiber.

B. SAMPLING REGIONS NEAR $\mathcal{C}_{OBST}$

Narrow passages might form when there are obstacles in robot workspace. These are regions of $\mathcal{C}_{free}$ for which its thickness locally along a codimension-1 hyper-surface is very small compared with other regions. It is unlikely for a random sampling algorithm to quickly find samples in the narrow passages. If none of them gets sampled the resulting roadmap is incomplete and will often lead to wrong predictions for many queries.

It has been observed by Amato et al. [20], [22] that samples near $\mathcal{C}_{obst}$ are very helpful for improving the visibility of robots if narrow passages develop around $\mathcal{C}_{obst}$. Because of the small thickness of the narrow passages, every sample in them is close to at least one of the C-obstacles. Such closeness manifests in a small gap between a workspace obstacle and at least one link of the closed chain. Recall that Q_i^ϵ comes from the boundary of a slightly-dilated $\mathcal{O}_i$, and more over we know that $\mathcal{C}_{obst} \subset \cup_{i,j} V^j_{CONV(Q_i^\epsilon)}$ from Proposition 1, 2, and 3. Therefore configurations in $\mathcal{C}_{free}$ for which one of the links intersects a point obstacle in Q^ϵ must be near $\mathcal{C}_{obst}$. The sampling algorithm is provided in Algorithm 3.

Algorithm 3 Sampling regions near $\mathcal{C}_{obst}$

```
Require: A vector of link lengths $(l_1, \cdots, l_m)$, the left and right anchor points, $P_l = [0,0]^T$ and $P_r = [l_m, 0]^T$, a set of $n_1$ point obstacles in the dilated point obstacle set $Q^\epsilon = \{p_i\}$, and a positive integer $n_2 > 0$
Ensure: Generate $n_2$ samples on each collision variety $V^j_{p_i}$ if it is not empty
  for $i = 1$ to $n_1$ do
    for $j = 1$ to $m - 1$ do
      Break the closed chain into a left open chain $(l_1, \cdots, l_j)$ based at $P_l$ and a right open chain $(l_{(j+1)}, \cdots, l_{(m-1)})$ based at $P_r$
      if $p_i$ can be reached by the left open chain then
        Left open chain forms a closed chain $CH_b$ when link $j$ insects $p_i$
        while number of generated samples $< n_2$ do
          Use Algorithm 2 (and also the result in Lemma 1) to sample $CH_b$, and for each such sample, the right open chain forms a closed chain $CH_f$ (refer to Fig. 7)
          Use Algorithm 2 to sample $CH_f$
          Discard the sample if it is in collision; and keep valid one
        end while
      end if
    end for
  end for
```

C. COLLISION CHECKING AND LOCAL PLANNER

Both Algorithm 2 and 3 rely on collision checking to eliminate invalid configurations. By triangulating $CONV(Q_i^\epsilon)$ and all robot links, collision checking can be performed by triangle-triangle intersection checking.

The next step after filtering out invalid samples is to construct a roadmap based upon the set of generated collision-free samples. This requires an efficient local planner to connect nearby samples. Although $\mathcal{C}$ might not be convex, our parametrization of $\mathcal{C}$ as a pair of elbow-up and elbow-down $(m-3)$-D tori allows us to construct an efficient local planner based upon a minimal-distance joint-interpolation motion of CH_2 based at $P_r = [l_m, 0]^T$, and an accordion motion of CH_1 based at $P_l = [0, 0]^T$.

Given a pair of configurations $\phi^1 = (\phi_1^1, \phi_2^1, \psi^1)$ and $\phi^2 = (\phi_1^2, \phi_2^2, \psi^2)$ of the original closed chain where $\psi^i = (\phi_3^i, \cdots, \phi_{m-1}^i)$, $i = 1, 2$, are the corresponding pair of configurations for CH_2. If $\phi_2^1 - \phi_1^1$ has same sign as $\phi_2^2 - \phi_1^2$ (i.e., they belong to the same torus), a local planner for CH_2 is constructed as

$$\psi(t) = \psi^1 + \delta\psi * t, \quad t \in [0, 1]\,, \tag{2}$$

where $\delta\psi$ is the vector $\psi^2 - \psi^1$ projected (through modular operation) to the interval $[-\pi, \pi]^{m-3}$. Under $\psi(t)$ the end-effector of CH_2 follows a smooth curve $\gamma(t) = f_2(\psi(t)) \in \mathbb{R}^2$ between $\gamma(0) = f_2(\psi^1) \in \mathbb{R}^2$ and $\gamma(1) = f_2(\psi^2) \in \mathbb{R}^2$ (Recall that f_2 is the forward kinematic map of CH_2). If every point in $\gamma(t)$ can be closed by CH_1, then we might apply the IK map f_1^{-1} of CH_1 to compute an interpolation which is "compliant" with $\gamma(t)$. $f_1^{-1}(\gamma(t))$ is referred to as an accordion move if the same sign as (ϕ_1^1, ϕ_2^1) (also (ϕ_1^2, ϕ_2^2)) is adopted in the calculation of the IK for every point in $\gamma(t)$.

Theorem 1: Let $r_{\min} = \|l_1 - l_2\|$, and $r_{\max} = l_1 + l_2$. Suppose the start and end points of $\gamma(t)$ lies in the strict interior of the workspace of CH_1, i.e., there exists $0 < \delta r < \frac{1}{2}(r_{\max} - r_{\min})$ such that $r_{\min} + \delta r < \|\gamma(0) - P_l\| < r_{\max} - \delta r$ and $r_{\min} + \delta r < \|\gamma(1) - P_l\| < r_{\max} - \delta r$. Then there exists an $\epsilon > 0$ such that as long as $\|\delta\psi\| < \epsilon$, $r_{\min} < \|\gamma(t) - P_l\| < r_{\max}$ for all $t \in [0, 1]$. If only one end of $\gamma(t)$ lies in the strict interior while the other end lies on a critical circle, then the same inequality holds for all $t \in (0, 1)$.

Proof: See Appendix B. □

Theorem 1 shows that as long as ϕ^1 and ϕ^2 have same sign (i.e. belong to the same torus), and ψ^1 and ψ^2 are sufficiently close, then there is always an accordion move for CH_1 that closes the loop at every $t \in [0, 1]$. Conversely if ϕ^1 and ϕ^2 have different sign, then the above local planner will fail. Fortunately, the milestones generated from Algorithm 2 contain samples exactly on the boundary variety B, which can be used for bridging pairs of configurations in different tori. For example ϕ^1 and ϕ^2 in different tori (but their components ψ^1 and ψ^2 are sufficient close) can be joined by two accordion moves, from ϕ^1 to a point $\phi^3 \in B$, and then from ϕ^3 to ϕ^2.

VII. EXPERIMENTAL RESULTS

Our method was implemented in Matlab and tested for many planning problems (with number of links from 6 to 19). All Matlab programs were run under Windows 10 and Intel Core $i7$. Our software can be downloaded from https://ai.stanford.edu/~liugf/closedchain.html, in which animation videos can be found for all the examples discussed below.

A. PATH PLANNING BETWEEN TWO CONFIGURATIONS IN DIFFERENT TORI

Our first set of experiments is centered around the challenges of C-space bifurcation caused by local coordinate charts. First we apply Algorithm 1 (denoted as Alg-1) and the traditional PRM algorithm without boundary samples (denoted as Alg-2) to the problem of a 5-bar closed chain in Fig. 8-(a), in which the start and goal configurations lie in the two different tori. This chain has same link length vector as the example in Section IV and Fig. 3, but with different set of point obstacles, $(1, 1)$ and $(1, 1.5)$. Both tori are parametrized by (ϕ_3, ϕ_4). The experimental results are summarized in Table 1, where regular samples simply mean they lie neither on B nor near $\mathcal{C}_{obst}$. It shows that the traditional PRM algorithm takes much longer computation time and requires more samples before obtaining the correct answer. Notice that here we regard any instantaneous jump of the elbow angle $(\phi_2 - \phi_1)$ of more than 0.04 radian (around 2 degree) as invalid due to the reason explained in the beginning of Section VI where we present the first challenge of C-space bifurcation. Samples on B help quickly find the path as shown in Fig. 8-(b) and 8-(c). Next we apply our algorithm to a 6-bar closed chain moving among 6 point obstacles in Fig. 9. The start and goal configurations

$$(-0.6363, -1.2183, 0.0416, 1.9416, -0.1416, \pi)$$
$$(-0.9063, 0.8648, 0.0416, -2.0416, 0.3416, \pi)$$

lie in two different tori. Our software generates 535 random samples (in less than a minute) which satisfy the loop-closure constraints and are collision-free. Among which 100 samples

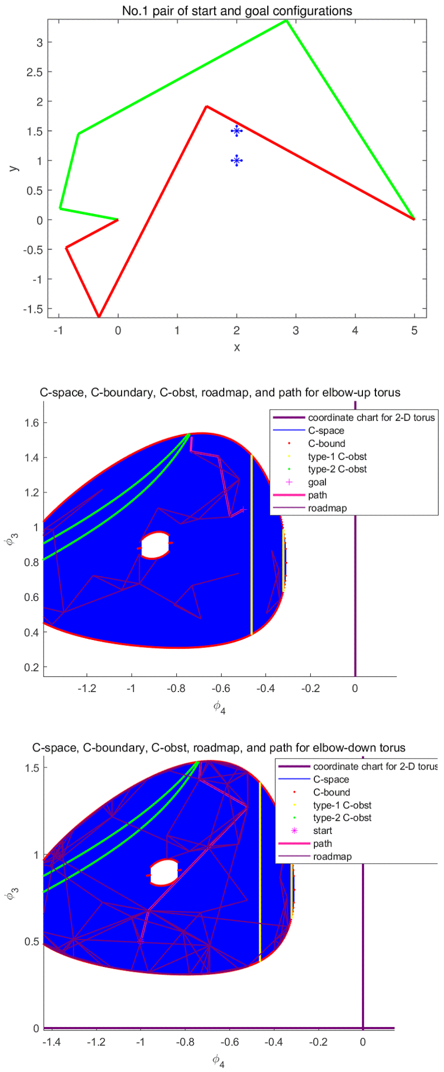

FIGURE 8: (a): Start and goal configurations at two different tori; (b): Boundary and collision varieties, roadmap and path segment on the elbow-up torus; (c): Boundary and collision varieties, roadmap and path segment on the elbow-down torus

TABLE 1: Performance comparison between algorithms with and without boundary sampling

Algorithms	Alg-2	Alg-1
No. of regular samples	2000	100
No. of B samples	0	100
No. of components	14	8
Computation time	16.44s	4.54s
prediction	failure	success

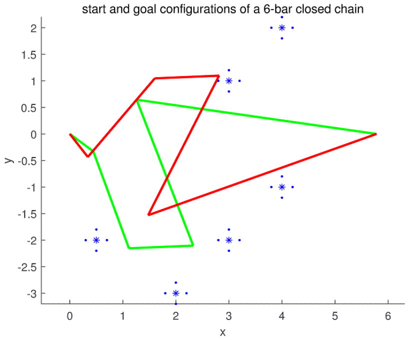

FIGURE 9: start (in green) and goal (in red) configurations of a 6-bar closed chain, original point obstacle set (blue star) and the dilated point obstacle set (blue dot)

are regular, 100 samples on B, and the remaining ones near $\mathcal{C}_{obst}$. Fig. 10, 11, and 12 show the resulting path. In the middle of path segment 5 in Fig. 12-(a), the robot finishes a motion that crosses B from the elbow-up torus to the elbow-down torus, through a sample on B.

B. NARROW PASSAGE PROBLEM

Our second sets of experiments are specifically designed so that the C-space of the target closed chain contains narrow passages. Fig. 13-(a) shows a 5-bar closed chain moving among two sufficiently close point obstacles. The link length vector of this chain is $[1, 1.3, 4, 4, 5]^T$, and the two point obstacles are $(1, 1.1)$ and $(1, 1.4)$. The start and goal configurations are $[-2.4, 0.75, 0.8847, -0.9727, \pi]^T$ and $[2.1, 0.15, 0.7503, -1.2415, \pi]^T$ respectively. At both configurations link 3 is confined between the two point obstacles. We apply Alg-1 with both non-canonical (ϕ_3, ϕ_4) and canonical parameters (ϕ_1, ϕ_2). The generated roadmaps in different parameter sets are respectively shown in Fig. 13-(a), Fig. 13-(b), Fig. 14-(a), and Fig. 14-(b). Both roadmaps yield correct paths through the samples near $\mathcal{C}_{obst}$.

The performance comparison between the algorithms with (Alg-1) and without (Alg-2) samples near $\mathcal{C}_{obst}$ is given in Table 2. It clearly demonstrates the importance of samples near $\mathcal{C}_{obst}$.

Next we compare the performance of both algorithms to a narrow passage problem of a 12-bar closed chain, also with two tightly arranged point obstacles, $(4, 1.9)$ and $(4, 2.5)$, as shown in Fig. 15. The link length vector for this mechanism is $[1.2000,\ 2.0000,\ 0.5512,\ 1.9457,\ 1.2131,\ 2.9482,\ 4.5684,\ 0.3,\ 0.3,\ 5,\ 2.5130,\ 8.5815]^T$, and the start and goal configurations are, respectively, $[-0.3800,\ 0.5708,\ 2.2956,\ 0.6335,\ 0,\ -0.3809,\ -1.2904,\ 3.91134,\ 2.3753,\ 1.1734,\ 3.6964,\ 3.1416]^T$ and $[0.2388,\ 1.7032,\ 1.4992,\ -0.3222,\ 0,\ -1.0564,\ -0.5959,\ 1.1011,\ 0.6490,\ 1.6947,\ 4.5497,\ 3.1416]^T$. Our algorithm can correctly plan a path that move links of the chain through the workspace narrow passages to their desired locations. The traditional algorithm turns out to be very difficult to find a valid path, see Table 3. Videos for both these two narrow-passage problems can be found in our website.

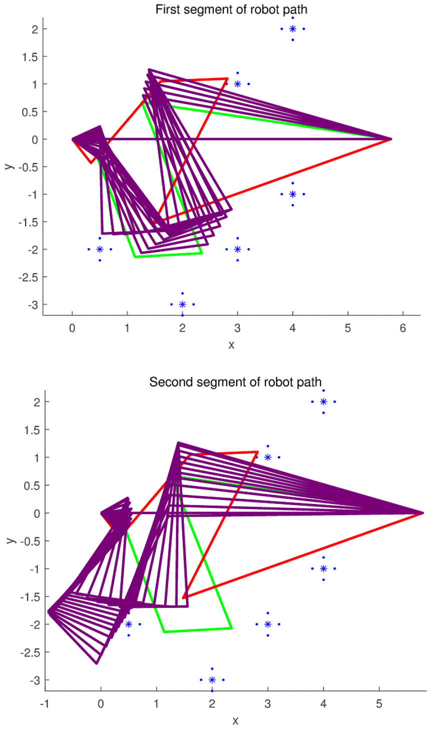

FIGURE 10: (a): First segment of robot path; (b): Second segment of robot path

TABLE 2: Performance comparison between algorithms for a 5-bar closed chain moving through narrow passages

Algorithms	Alg-2	Alg-1
No. of regular samples	2000	100
near $\mathcal{C}_{obst}$ samples	0	740
No. of components	18	21
Computation time	40.95s	4.97s
prediction	failure	success

C. HIGH-DIMENSIONAL CLOSED CHAINS IN GENERAL

In general situations when $\mathcal{C}$ has no narrow passages, $\mathcal{C}$ tends to be more expansive as the number links increase. Both

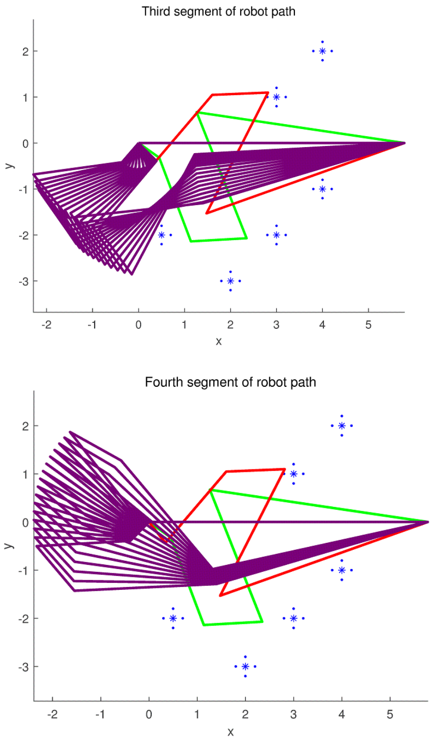

FIGURE 11: (a): Third segment of robot path; (b): Fourth segment of robot path

TABLE 3: Performance comparison between algorithms for a 12-bar closed chain moving through narrow passages

Algorithms	Alg-2	Alg-1
No. of regular samples	40000	1000
near $\mathcal{C}_{obst}$ samples	0	6322
No. of components	3222	2432
Computation time	>5 hours	3009 s
prediction	failure	success

the traditional PRM algorithm and our algorithm can yield reasonable solution.

For example we apply our algorithm to the example of a 10-bar closed chain moving among 10 point obstacles. The link lengths are $\{1.2, 2.0, 0.5512, 1.9457, 1.2131, 2.9482, 4.5684, 0.3, 0.3, 8.5815\}$, and the start and goal configurations are $\{0.6669, -0.3802, -0.6014, -1.1834, 0.0765, 1.9765, -0.1067, -0.3255, -2.7811, 3.1416\}$ and $\{-0.6669, 0.3802, -0.9412, 0.8299, 0.0067, -2.0765, 0.3067, 0.3255, 2.7811, 3.1416\}$, respectively. Our algorithm generates over 3000 milestones (2000 samples are near $\mathcal{C}_{obst}$) for the roadmap, and the resulting path is shown in Fig. 17, 18, and 19. In this example the mechanism crosses the boundary variety in Fig. 17-(a), and pass through several points in the dilated point obstacle set in Fig. 18 and 19 toward far left for adjusting the relative angle between link 6 and 7 before arriving the goal configuration. We remark that the traditional PRM algorithm can also solve this problem because $\mathcal{C}$ contains no narrow passages. The only difference is that the solution of our algorithm might include milestones near $\mathcal{C}_{obst}$.

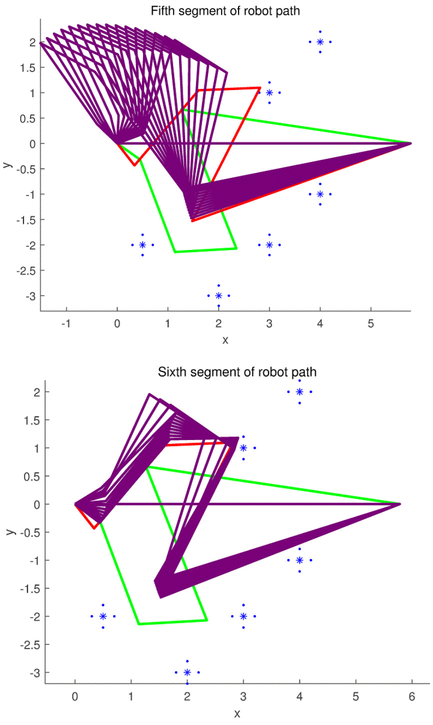

FIGURE 12: (a): Fifth segment of robot path; (b): Last segment of robot path

Although it is hard to compare the performances of both algorithms to a specific path query problem without narrow passages. A meaningful comparison can be carried out by counting the successful rate of two equal-size roadmaps generated by Alg-1 and Alg-2, respectively, under a set of randomly generated start-goal pairs. Table 4 gives the comparison results for an 18-bar closed chain moving among 10 obstacles, and a 20-bar closed chain moving among 12 obstacles. Appendix C provides an estimation of the number of components under different conditions. In our case the

TABLE 4: Successful rate of roadmaps under a set of start-goal pairs

closed chain	Alg-2	Alg-1
18 links 10 obstacles	8000 vertices 12/20 successful rate 2797 components	8000 vertices 15/20 successful rate 1336 components
20 links 12 obstacles	8000 vertices 13/20 successful rate 3726 components	8000 vertices 19/20 successful rate 2252 components

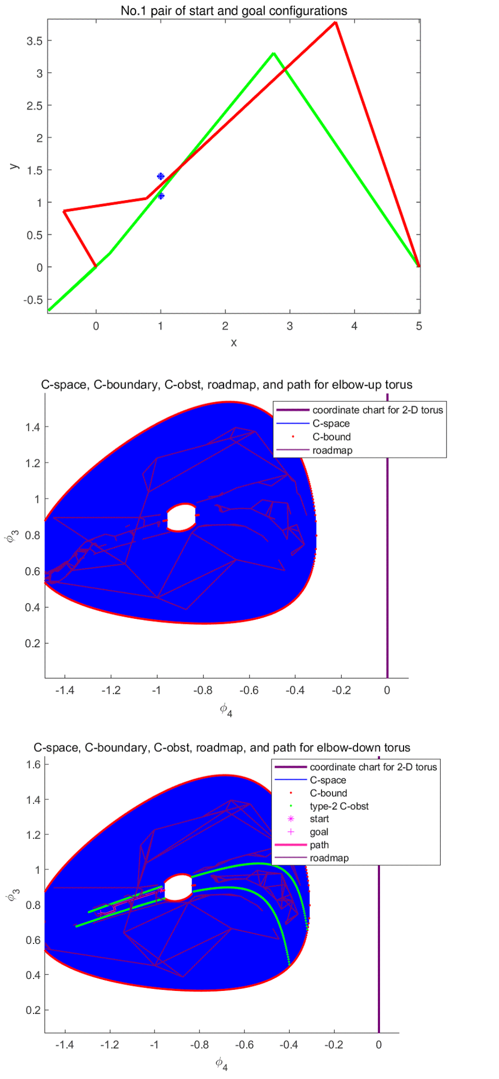

FIGURE 13: (a): Start and goal configurations confined by two close point obstacles; (b): Roadmap on the elbow-up torus (using non-canonical parametrization); (c): Roadmap and path on the elbow-down torus (using non-canonical parametrization)

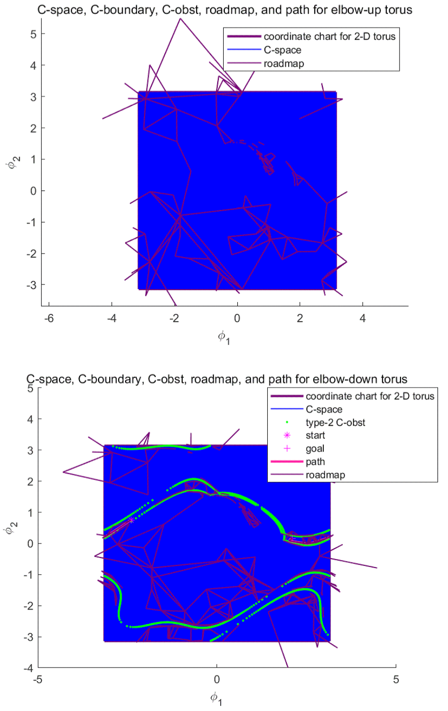

FIGURE 14: (a): Roadmap on the elbow-up torus (using canonical parametrization); (b): Roadmap and path on the elbow-down torus (using canonical parametrization)

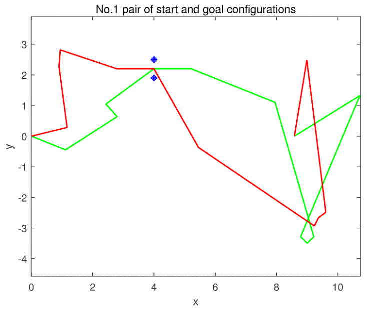

FIGURE 15: Start and goal configurations of a 12-bar closed chain with two point obstacles, whose C-space contains a narrow passage

elbow-up and elbow-down tori are connected (using the test of the long link set introduced in Section III). Moreover the number of components for the 10-bar and 20-bare cases is estimated to be 1024 and 2048 respectively. Therefore Alg-1 yields a better roadmap with high successful rate in path queries, and more accurate estimation on the number of connected components of $\mathcal{C}_{free}$.

VIII. CONCLUSION

This paper presents a framework that employs the structural sets of $\mathcal{C}$ for solving the motion planning problems of planar closed chains moving among point obstacles with extension to arbitrary convex 2-D obstacles. This framework combines results on the topology of $\mathcal{C}$, its boundary variety B and the collision varieties $\mathcal{C}_{obst}$. First, $\mathcal{C}$ is covered by two charts, each embedded in an $(m$-$3)$-dimensional torus. The connectivity between the two tori is completely determined by B. Second, the structure of B and $\mathcal{C}_{obst}$ is analyzed using the C-spaces of lower-dimensional closed chains. We then present a resolution-complete sampling-based algorithm by approximating workspace obstacles as a set of point obstacles. This algorithm generates well-distributed samples in the interior of $\mathcal{C}$, B, and the regions near $\mathcal{C}_{obst}$. Compared with traditional algorithms, this new algorithm not only avoids massive time-consuming projections from the ambient space to $\mathcal{C}$, but also is able to solve problems for which $\mathcal{C}$ contains bifurcations and narrow passages, as demonstrated by examples.

.

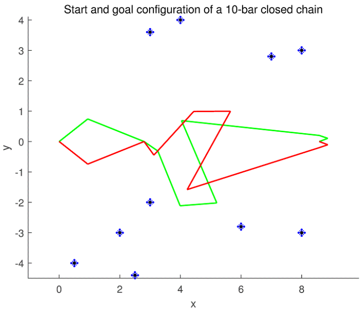

FIGURE 16: Start (in green) and goal (in red) configurations of a 10-bar closed-chain moving among 10 point obstacles

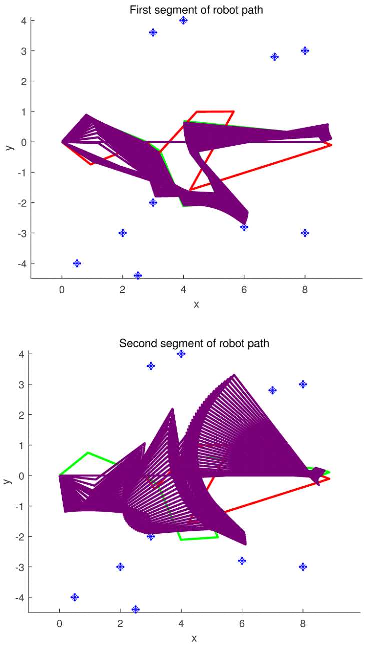

FIGURE 17: (a): First segment of robot path; (b): Second segment of robot path

APPENDIX A PROOF OF LEMMA 1

When link $j, 1 < j < m-1$ is in contact with point obstacle p_i, the feasible range of the proximal end of link j lies in the disk of radius l_j centered at p_i. This workspace is equivalent to that of a 2-bar mechanism based at p_i with link lengths $(1/2l_j, 1/2l_j)$. Therefore $V_{p_i}^j$ is equivalent to the C-space of the closed chain $(l_1, \cdots, l_{j-1}, 1/2l_j, 1/2l_j, l_b)$, where $l_b = \|p_i\|$.

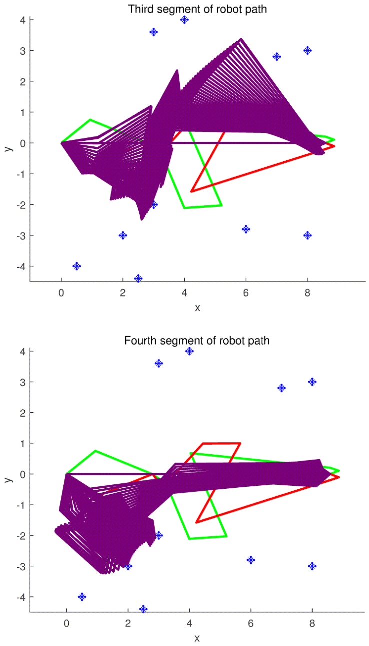

FIGURE 18: (a): Third segment of robot path; (b): Fourth segment of robot path

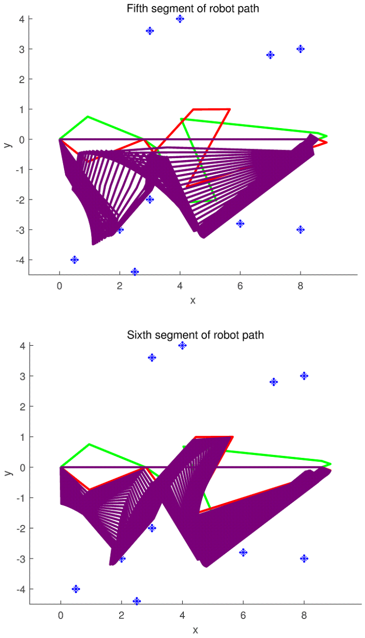

FIGURE 19: (a): Fifth segment of robot path; (b): Last segment of robot path

APPENDIX B PROOF OF THEOREM 1

We first prove that $r_{\min} < \|\gamma(t) - P_l\|$, and the fact $\|\gamma(t) - P_l\| < r_{\max}$ can be proved in the same manner. We have

$$\begin{aligned} \|\gamma(t) - P_l\| &= \|f_2(\psi(t)) - P_l\| \\ &= \|f_2(\psi(t)) - f_2(\psi^1) + f_2(\psi^1) - P_l\| \\ &\geq \|f_2(\psi^1) - P_l\| - \|f_2(\psi(t)) - f_2(\psi^1)\| \\ &\geq r_{\min} + \delta r - \|Jf_2(\psi(t_1))\| \|\psi(t) - \psi^1\| \end{aligned}$$

for some $t_1 \in [0,1]$. Because the interval $[0,1]$ is compact, $\|Jf_2(\psi(t_1))\|$ is bounded from above by a positive value $\eta > 0$. Therefore as long as $\|\psi(t) - \psi^1\| \leq \|\psi^2 - \psi^1\| < \epsilon < \frac{\delta r}{\eta}$, then $r_{\min} < \|\gamma(t) - P_l\|$. If without generality we assume $\|\gamma(0) - P_l\| = r_{\min}$, then the inequality $r_{\min} < \|\gamma(t) - P_l\|$ is still valid for all $t \in (0,1]$, and the equality only holds at $t = 0$.

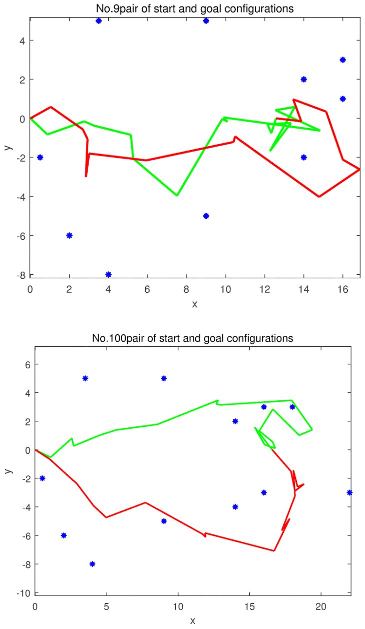

FIGURE 20: (a): Disconnected configurations of an 18-bar chain; (b): Disconnected configurations of a 20-bar chain

APPENDIX C ESTIMATE THE NUMBER OF COMPONENTS OF $\mathcal{C}_{FREE}$

For a planar m-link closed chain among n convex obstacles, although the exact number of components of $\mathcal{C}_{free}$ is very hard to compute (with almost same complexity as computing a complete roadmap), it is possible to put a rough estimation using topological argument. Consider the connectivity between a pair of configurations of the chain. Geometrically these two configurations form a closed loop in $\mathbb{R}^2$. The number of components of $\mathcal{C}_{free}$ clearly depends on the number of 360-degree loops (or winding number in mathematics) of the former loop and the subsets of obstacles contained in each 360-degree loop. The latter are in turn determined by the overall distribution of obstacles and their distances to the two fixed end points p_l and p_r. Let the entire plane be divided into the top and bottom half-planes along the line P_lP_r. Consider the convex hull of the obstacles as well as the two fixed end on each half-plane. Let L_t and L_b be, respectively, the length of the perimeter of the top and bottom convex hulls, and $L_0 = \max(L_t, L_b)$. If $k < \frac{\sum_i l_i}{L_0} < k+1$, then the number N of connected components roughly satisfies

$$(2^n)^k < N < b(2^n)^{(k+1)},\ b \in \{1,2\}. \tag{3}$$

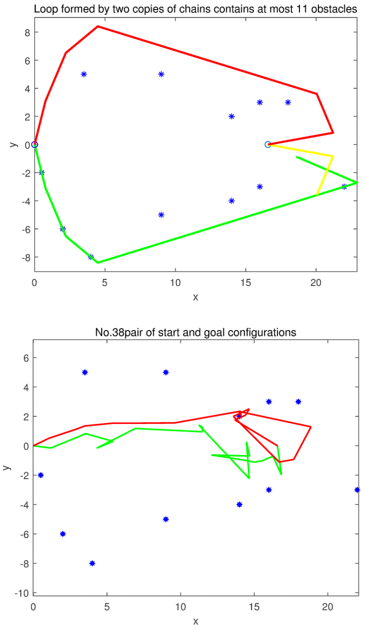

FIGURE 21: (a): Loop formed by the two copies of a 20-bar chain contain only up to 11 obstacles; (b): $O(n^2)$ exceptional cases with internal loops that contain obstacles

We deduce this result as follows. Refer to Fig. 20, the loop formed by the two copies of the chains, one at start configuration (green) and the other at goal configuration (red), might contain up to n obstacles. Obviously, as long as there are obstacles inside the loop the two configurations are disconnected. The feasible number of connected components can be counted by fixing the start configuration, and let the goal configuration vary so that the loop encloses different sets of obstacles. Each such set leads to a new component in $\mathcal{C}_{free}$. As a result, there are at least $\sum_{i=0}^{n} C_n^i = 2^n$ components if the loop can wind around the n obstacles once. Considering the fact that $\mathcal{C}$ itself might contain two components, the maximal value of b is 2. This leads to Eqn. (3). If the chain is longer enough so that the loop can wind around the set of obstacles twice, then the number of components can reach $2(2^n)^2$. Eqn. (3) tends to be accurate as the number of links increases, because the more links the chain has, the closer the chain behaves like a segment of a curve with two fixed ends. For the example in Fig. 20-(b) and the second case in Table 4, $\frac{\sum_i l_i}{L_0} < 1$. This is because the green configuration, as shown in Fig. 21-(a), is not long enough to contain all obstacles in the bottom half plane, and only 11 obstacles can be enclosed inside the loop formed by the red and the half-green-half-yellow configurations. Although there are exceptional cases where a configuration contains internal loops (see Fig. 21-(b)). It is not difficult to verify that with the given link length vector, the number of such exceptional cases is at most $O(n^2)$. Therefore, the number of components for the 20-bar case in Table 4 is close to $2^{11} = 2048$. Similarly for the 18-bar example in Fig. 20-(a) and also in the first case of Table

4, the number of components is close to $2^{10} = 1024$ because $1 < \frac{\sum_i l_i}{L_0} < 2$.

ACKNOWLEDGMENT

The authors would like to thank Jim Milgram for introducing them to many of the ideas that led to the results obtained.

REFERENCES

[1] J. Hopcroft and G. Wilfong, On the motion of objects in contact. International Journal of Robotics Research, Vol. 4, No. 4, PP. 32-46, 1986.

[2] J.F. Canny, The Complexity of Robot Motion Planning. MIT Press, Cambridge, MA, 1988.

[3] J.C. Latombe, Robot Motion PLanning. Kluwer Academic Publishers, New York, 1991.

[4] J. Schwartz, M. Sharir, and J. Hopcroft, Planning, Geometry, and Complexity of Robot Motion. Ablex Publishing Corp., 1986.

[5] L.E. Kavraki, P. $\check{S}$vestka, J.C. Latombe, and M.H. Overmars, Probablistic Roadmaps for path planning in high-dimensional configuration space. IEEE Transactions on Robotics and Automation, Vol. 12, No. 4, PP. 566-580, 1996.

[6] S. M. LaValle and J. J. Kuffner, Rapidly-exploring random trees: Progress and prospects. In B. R. Donald, K. M. Lynch, and D. Rus, editors, Algorithmic and Computational Robotics: New Directions, PP. 293-308, AK Peters, Wellesley, MA, 2001.

[7] S. R. Lindemann and S. M. LaValle, Current issues in sampling-based motion planning. In P. Dario and R. Chatila, editors, The 11th International Symposium on Robotics Research. Springer Tracts in Advanced Robotics, Vol. 15, PP. 36-54, Springer, Berlin, Heidelberg, 2005.

[8] J. Yakey, S. M. LaValle, and L. E. Kavraki, Randomized path planning for linkages with closed kinematic chains. IEEE Transactions on Robotics and Automation, Vol. 17, No. 6, PP. 951-958, 2001.

[9] J. Cortes, Motion planning algorithms for general closed-chain mechanisms. Doctorat, Institut National Polytechnique, Toulouse, December 16, 2003.

[10] J.T. Schwartz and M. Sharir, On the piano movers II. General techniques for computing topological properties on real algebraic manifolds. Adv. Appl. Math., Vol. 4, No. 3, PP. 298-351, 1983.

[11] L. Han and N.M. Amato, A kinematics-based probabilistic roadmap method for closed chain systems. In B.R. Donald, K.M. Lynch, and D. Rus, editors, Algorithmic and Computational Robotics: New Directions, PP. 233-246, AK Peters, Wellesley, 2001.

[12] J.C. Trinkle and R.J. Milgram, Complete Path Planning for Closed Kinematic Chains with Spherical Joints. International Jounral of Robotics Research, Vol. 21, No. 9, PP. 773-789, 2002.

[13] G.F. Liu, J.C. Trinkle, R.J. Milgram, Complete Path Planning for Planar 2-R Manipulators With Point Obstacles. Proceedings of the IEEE International Conference on Robotics and Automation, PP. 3263-3269, 2004.

[14] G.F. Liu, J.C. Trinkle, R.J. Milgram, Complete Path Planning for Planar 3R-Manipulators Among Point Obstacles. In: M. Erdmann, M. Overmars, D. Hsu, F. van der Stappen, editors, Algorithmic Foundations of Robotics VI. Springer Tracts in Advanced Robotics, Vol. 17, PP. 329-344, Springer, Berlin, Heidelberg, 2005.

[15] B. Kim, T. T. Um, C. Suh, and F. C. Park, Tangent bundle RRT: A randomized algorithm for constrained motion planning. Robotica, Vol. 34, No. 1, PP. 202-225, 2014.

[16] G.F. Liu and J.C. Trinkle, Complete Path Planning for Planar Closed Chains Among Point Obstacles. In Robotics: Science and Systems, PP. 33-40, 2005.

[17] G. Liu, Y. Lou, and Z. Li, Singularities of parallel manipulators: a geometric treatment. IEEE Transactions on Robotics and Automation, Vol. 19, No. 4, PP. 579-594, 2003.

[18] F. C. Park and J. W. Kim, Singularity analysis of closed kinematic chains. Journal of Mechanical Design, Transactions of the ASME, Vol. 121, No. 1, PP. 32-38, 1999.

[19] N. Shvalb, M. Shoham, H. Bamberger, and D. Blanc, Topological and kinematic singularities for a class of parallel mechanisms. Mathematical Problems in Engineering, Vol. 2009, Article ID 249349, 12 pages, 2009.

[20] S. Rodriguez, X. Tang, J. Lien, and N. Amato, An obstacle-based rapidly-exploring random tree. Proceedings of the IEEE International Conference on Robotics and Automation, PP. 895-900, 2006.

[21] M. Stilman, Task Constrained Motion Planning in Robot Joint Space. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, PP. 3074-3081, 2007.

[22] N. Amato, O. Bayazit, L. Dale, C. Jones, and D. Vallejo, OBPRM: An obstacle-basedPRM for 3D workspaces. In P.K. Agarwal, L.E. Kavraki, M.T. Mason, editors, Robotics: The algorithmic perspective, PP. 155-168, AK Peters, Natick, 1998.

[23] D. Berenson, S. Srinivasa, D. Ferguson and J. Kuffner, Manipulation Planning on Constraint Manifolds. Proceedings of the IEEE International Conference on Robotics and Automation, PP. 625-632, 2009.

[24] D. Berenson, S. Srinivasa and J. Kuffner, Task space regions: A framework for pose-constrained manipulation planning. International Journal of Robotics Research, Vol. 30, No. 12, PP. 1435-1460, 2011.

[25] D. Hsu, L.E. Kavraki, J.C. Latombe, R. Motwani, and S. Sorkin, On finding narrow passages with probabilistic roadmap planners. In P.K. Agarwal, L.E. Kavraki, M.T. Mason, editors, Robotics: The algorithmic perspective, PP. 141-153, AK Peters, Natick, 1998.

[26] E.A. Coutsias, C. Seok, M.P. Jacobson, and K.A. Dill, A kinematic view of loop closure. Journal of Computational Chemistry, Vol. 25, No. 4, PP. 510-528, 2004.

[27] M.G. Carmichael, D.K. Liu and K.J. Waldron, A framework for singularity-robust manipulator control during physical human-robot interaction. International Journal of Robotics Research, Vol. 36, No. 5-7, PP. 861-876, 2017.

[28] C. Wampler and L. Leifer, Applications of damped leastsquares methods to resolved-rate and resolved-acceleration control of manipulators. Journal of Dynamic Systems, Measurement, and Control, Vol. 110, No. 1, PP. 31-38, 1988.

[29] Y. Nakamura and H. Hanafusa, Inverse kinematic solutions with singularity robustness for robot manipulator control. Journal of dynamic systems, measurement, and control, Vol. 108, No. 3, PP. 163-171, 1986.

[30] J.M. Porta and L. Jaillet, Path Planning on Manifolds Using Randomized Higher-Dimensional Continuation. In: D. Hsu, V. Isler, JC. Latombe, M.C. Lin, editors, Algorithmic Foundations of Robotics IX. Springer Tracts in Advanced Robotics, Vol. 68, PP. 337-353, Springer, Berlin, Heidelberg, 2010.

[31] L. Jaillet and J.M. Porta, Path Planning with Loop Closure Constraints Using an Atlas-Based RRT. In: H. Christensen, O. Khatib, editors, The 15th International Symposium on Robotics Research. Springer Tracts in Advanced Robotics, Vol. 100, PP. 345-362, Switherland, 2017.

GUANFENG LIU was born in WuXi, JiangSu, China, in 1975. He received the B.E. degree in electrical engineering from ZheJiang University, China in 1998, and the Ph.D. degree in electrical engineering from Hong Kong University of Science and Technology, Hong Kong in 2003.

From 2003 to 2005, he was a postdoc with the Computer Science Department, Rensselaer Polytechnic Institute, Troy, NY. From 2005 to 2007, he was a research fellow with the Computer Science Department, Stanford University, Stanford, CA. From 2009 to 2011, he was a senior robotics engineer at Solyndra Inc., CA. From 2012-2017 he was an adjunct professor with the School of Mechatronical Engineering, Guangdong University of Technology, Guangzhou. Since 2017, he has been an adjunct professor with the School of Mechatronics Engineering, Guangdong Polytechnic Normal University, China. He is the author of more than 50 articles. His research interests include robotics, manipulation, motion planning, and protein modeling.

JEFF TRINKLE received bachelor's degrees in Physics (1979) and Engineering Science and Mechanics (1979) from Ursinus College and Georgia Institute of Technology, respectively. In 1987, he received his PhD from the Department of Systems Engineering at the University of Pennsylvania, where he was a research assistant in the GRASP Laboratory.

Since 1987, he has held faculty positions in the Department of Systems and Industrial Engineering at the University of Arizona and the Department of Computer Science at Texas A&M University. From 1998 to 2003 he was a visiting research scientist at Sandia National Laboratories in Albuquerque New Mexico. He moved to Rensselaer Polytechnic Institute in Troy, New York, in 2003, where he served as Chair of the Computer Science Department until 2009. He was also Professor of Computer Science, Director of the CS Robotics Lab, and Faculty Dean of Residential Commons at RPI. Dr. Trinkle's primary research interests lie in the areas of robotic manipulation, robot motion planning, and physical simulation.

Dr. Trinkle has served on the editorial boards of IEEE Transactions on Robotics and Automation, IEEE Transactions on Automation Science and Engineering, and Robotica. He is the recipient of the 1985 IBM Graduate Research Fellowship, the 1989 Research Initiation Award from the National Science Foundation, the 1994 Texas A&M Center for Teaching Excellence Award, the 1998 Plank Company Faculty Fellowship, the 2004 Kayamori Best Automation Paper of the IEEE International Conference on Robotics and Automation, and the 2009 Humboldt Research Prize.

YONG YANG was born in Hunan, China, in 1968. He received the B.E., degree in mechanical engineering from ChongQing University, China, in 1990, and the M.E., degree in mechatronics engineering from Hunan University, China, in 1997, and the Ph.D. from Central South University, China, in 2006.

He studied in Linkoping University, Sweden, as a visiting scholar in 2005. He is currently a professor with the School of Mechatronics Engineering, Guangdong Polytechnic Normal University, China. He published more than 100 research papers on National and International Journals and Conferences. His research interests include mechatronics, robotics, automation, and advanced manufacturing engineering.

Dr. Yang is a member of IEEE, a vice president of Guangdong Mechinery Industry Association, and the executive director of Guangdong Institute of Mechanical Engineering.

SHAOMING LUO was born in JiaYu, HuBei, China, in 1966. He received the B.E. degree in Pressure Processing from Chongqing University, China in 1988, and M.E. degree in Pressure Processing from Chongqing University, China in 1991, and Ph.D. degree in Solid Mechanics from Chongqing University, China in 1998.

From 1999 to 2001, he was a postdoc with the State Key Laboratory of Mechanical Transmission of Chongqing University, China. From 2012 to 2017, he was a professor and president with Zhongkai University of Agricultural Engineering, Guangzhou, China. He is currently a professor and president with Guangdong Polytechnic Normal University, Guangzhou, China.

He is the vice chairman of agricultural aviation branch of China Agricultural Machinery Society(CAMS), vice chairman of Guangdong agricultural machinery society(GAMS), academic committee member of International Joint Research Center of Aviation Application Technology of National Precision Agriculture, director of Guangdong Research and Development Center of Intelligent Equipment Technology in Precision Agriculture. He is the author of more than 100 articles. His research interests include smart agricultural engineering, robotics, mechatronics, and advanced manufacturing technology.