On Free Velocity Cones of Narrow Passages of High-DoF Kinematic Chains

Guanfeng Liu[1], J.C. Trinkle[2], Yong Yang[1], and Junjie Li[1]
[1]School of Electromechanical Engineering, Guangdong Polytechnic Normal University
[2]Department of Computer Science, Leigh University
[1] liugf1004@gmail.com
[2] jeff.trinkle@lehigh.edu

***Abstract*—Narrow passages in free space pose great challenges to many sampling-based motion planners. In problems with high-dimensional configuration spaces (C-spaces), narrow passages in free space (C-free) can be hard to find and navigate, and sometimes even counter-intuitive. In this paper, we present an algorithm to construct cones of available instantaneous velocities, (*"free velocity cones", or briefly,* "free cones") on the boundary of C-free to facilitate finding and navigating narrow passages. This is accomplished by first developing free cones and associated measures of local C-space narrowness for a single rigid link. These results are then extended to kinematic chains (open and closed) of arbitrary degrees of freedom. It turns out that the degeneracy of the free cones dictates the existence of C-space narrow passages, and the locations and orientations of the links in the workspace relate intimately to the corresponding connected component of C-free. This observation leads us to an algorithm of combining the enumeration of topological components and random sampling of the free cones. Experimental results from applying our algorithm to several challenging examples show that our new algorithm is more efficient than some variants of probabilistic roadmap (PRM) algorithms for multi-query problems, and several variants of rapidly exploring random tree (RRT) algorithms, such as RRT-CONNECT and RRV for single-query problems.**

I. Introduction

The narrow passage problem remains open in the field of robot motion planning. The importance of the problem was noticed since the early development of sampling-based motion planners [9], [23]. The challenges come mainly from the significantly smaller volume measure of narrow passages relative to the configuration spaces (C-spaces) containing them. To date, there are no efficient methods to reliably identify narrow passages and generate enough samples for building well-behaved roadmaps or trees. Several distance or volume based indices have been proposed to characterize narrow passages in C-space, e.g., path clearance [2], ϵ-goodness [3], expansiveness [4], and k-clearance [1]. Every measure has its own advantages and drawbacks. In motion planning problems with large numbers of degrees of freedom (more than three) and complex obstacles the complexity of precisely calculating these indices makes their use for finding and navigating narrow passages impractical. In extreme situations, narrow passages could be pathologically curved or have counter-intuitive shapes, making the search for useful points in configuration space even more difficult and time consuming. For example narrow passages in C-spaces of dimension greater than 2 might form complex knots. It is also possible that a C-space component is of small volume and everywhere narrow. Fig. 6 shows such a problem; the links of a planar closed chain pass through a narrow gap between two polygonal obstacles. Similarly, in Fig 8, the links of a spatial closed chain pass through a small hole in a toroidal obstacle. For long and highly-curved narrow passages, not only are more samples required within the passage, but also each sample has to be sufficiently close to its nearest neighbor so that a simple local planner that tests sample connectivity will not be trapped by C-space obstacles (or C-obstacles).

A. Theoretical foundations of sampling-based algorithms

Sampling-based motion planning algorithms have garnered popularity and great successes in many motion planning problems. Their main principle is to learn the connectivity of C-space, through a collection of random C-space samples and collision checks. Such connectivity, depending on the categories of problems to solve, is encoded in a graph or a tree (or multiple trees). Graphs are often used to answer multi-query problems, while trees are more suited to single-query problems. The corresponding algorithms for them are respectively called probabilistic roadmap method (PRM) [9], [23] and rapidly-exploring random tree (RRT) [41].

The successes of these algorithms have fueled the research about their theoretical foundations. Although both PRM [2] and RRT [41], [46] have been shown to be probabilistic complete, the actual computational time for a specific problem can not be pre-determined. In particular we cannot exclude the possibility that the incremental growth of the graph or tree will be trapped by some extremely curved and narrow passages in a high dimensional C-space. Lindermann and LaValle [45] examined another fundamental issue related to random sampling itself. They elucidated the Voronoi-bias nature of the RRT algorithm, and then deduced that a similar bias can be generated via deterministic multi-sample RRT algorithms. A Voronoi

bias indicates the fact that the planner tends to grow the tree in largely unexplored portions of C-space. However, a Voronoi bias does not guarantee a narrow-passage bias, so it cannot prevent an RRT tree from becoming trapped in long curved narrow passages, as pointed out in [40].

B. Previous solutions to the narrow passage problem

Despite the difficulties of the problem, some progress has been made in the past decade. Earlier progresses are mainly centered on applying different heuristics to address the challenge of narrow passages, including workspace information guided sampling [14], [32], sampling boundaries of C-space obstacles [19], converging to medial-axis [5], [20], [42], dilation of the free space [23], the bridge-test methods [24], manifold sampling [1], iterative relaxation of constraints (IRC) [48], and retraction along the contact space of C-space obstacles [37]–[39]. These methods are based upon intuition that works on narrow-passage problems in low-dimensional spaces, but it is highly non-trivial to generalize them to problems in high-dimensional spaces. Techniques described in [5], [19], [23], which seem quite natural in lower-dimensional spaces, are subject to explosive growth of complexity for narrow passages in high-dimensional C-space. The bridge-test method [24] is very efficient for many high-dimensional problems, but still requires enormous numbers of bridge tests to find enough samples to cover narrow passages. Moreover, the desired lengths of the used bridges as well as the landscape for carrying out the bridge tests are hard to accurately estimate before sampling. Manifold-sampling approach [1] picks specific sub-manifolds that conforms with the narrow passage, and then generates samples on an arrangement of orthogonal sub-manifolds. However this method so far is only demonstrated to be successful in special examples. IRC method [48] solves an easier problem first by relaxing some feasibility constraints, and then deforms the solution toward the one for the final harder problem by incrementally putting back the constraints. There are, however, no unified strategies for relaxing the constraints for different categories of motion planning problems. The retraction approach in [37]–[39] enhances the standard algorithm of rapidly exploring random tree (RRT) by generating a sequence of samples on the contact space through an iterative optimization algorithm. For many narrow-passage problems, this approach often generates more samples in the entrance as well as the interior of the passage, and therefore achieves a higher success rate. However, this approach relies on the fact that there exists leaves on the RRT tree which are already near the entrance of the passage.

In recent years several new techniques have been introduced to the problem. Vonasek et al. [22] developed an iterative scaling approach to find a guiding path through narrow passages in high-dimensional C-space. Szkandera et al. [29] proposed a modified RRT algorithm that identifies and samples the exit regions of C-space cavities, and then merges the RRT trees of the exit regions with the main RRT tree of the free space. Tihrovic and Ferizbegovic [40] developed a new RRT variant, the rapidly-exploring random vines algorithm (RRV), which employs the tool of principle component analysis (PCA) for estimating the relative location of a trapped leaf node with respect to possible narrow passage. For trapped leaf nodes, PCA produces a projection map that guides the expanding toward unexplored narrow passage regions. This algorithm has been demonstrated to be more efficient than other RRT variants. However the accuracy of PCA is highly sensitive to the curvature of the passage as well as the number of sampling points around the trapped node used in the PCA. RRV can be regarded as a type of adaptive sampling algorithm based upon sampling density estimation. Other examples of such sampling can be found in [47], [49]. In particular the method of [47], Search Tree with Resolution Independent Density Estimation (STRIDE), employs a balanced tree data structure for estimating the sampling density. It shows promising performances in several narrow passage problems involving high-dimensional C-spaces.

C. Relation to closed chains

The narrow passage problem in its nature is similar to, but not completely same as, the motion planning problem with holonomic constraints, which exist in closed chains. Both problems have to address the challenges of sampling a subset of small volume measure within a large ambient space. Although the domains of both problems are semi-algebraic sets, only equality constraints appear in the latter problem. When the gap of narrow passage goes to 0, the former problem converges to the latter.

A number of strategies have been proposed for the resolution of equality constraints in closed chains [10], [11], [15], [21]. Yakey et al. [10] approximate holonomic constraints by semi-algebraic sets with inequalities, thus converting the original problem into one with an artificial narrow passage. Others derive samples either through the inverse kinematics (IK) of sub-chains [11], [15], interpolation along tangent spaces [17], continuation across multiple local coordinate charts [25]–[27], and reachable volume estimation and sampling [50], [51]. The C-space itself might contain narrow passages when obstacles are introduced into the corresponding workspace. Although narrow passages still manifest a small-volume subset of the C-space sub-manifold, the non-Euclidean topology of the latter introduces extra complexities, not only in the decomposition of the free space into connected components [28], but also in the topology of each individual component [16]. The treatment of narrow passages for closed chains is, however, largely undeveloped.

D. Contribution

In this paper we develop a novel tool, the free velocity cone, for identifying and navigating C-space narrow passages of kinematic chains of arbitrary degrees of freedom (DoFs). Our major innovations are

1) Computing the free velocity cones for a single rigid link of arbitrary geometry using its near-contact model;
2) Necessary and sufficient conditions for narrow passages of a single rigid link in terms of the degeneracy of the free velocity cones;
3) Computing the free velocity cones for a kinematic chain by assembling the free velocity cones of the set of restrained links and applying their corresponding IKs;
4) Necessary and sufficient conditions for narrow passages of a kinematic chain;
5) Directly sampling narrow passages (i.e., topological samples) by enumerating randomly the set of restrained links, aligning them with the workspace medium axis and applying the IK of subchains;
6) Developing projection maps based upon the free velocity cones for quickly expanding the RRT tree using computationally cheap samples and thus reducing the use of too many expensive topological samples.

This paper is organized as follows. In Section II we study the free velocity cones of a kinematic chain. In Section III we propose the concept of topological components and related analysis tools. In Section IV we present a new RRV-type algorithm. We present experimental results in Section V, and finally we draw conclusion in Section VI.

II. Narrow passage and free velocity cone of a kinematic chain

In this section we first provide a rigorous definition of the free velocity cone and narrow passage of a rigid body from its near-contact kinematics, and then generalize the result to narrow passages in C-space of kinematic (open or close) chains of arbitrarily DoFs. Our notations are summarized in Table I.

A. Narrow passage and free velocity cone of a rigid body

Consider first a rigid object moving among several obstacles in its workspace. Suppose it has close proximity with these obstacles, but without collision. Then the set of near-contact points $\{p_i\}$ as well as the initial gap $d_{i,0}$ at p_i measure the narrowness of the workspace around the object.

Recall a result from multi-fingered grasping [30], [31] that an infinitesimal motion (or twist) δV of the rigid object with respect to its body frame will result in the offsets $\{\delta d_i\}$ of $\{p_i\}$ along its outward normal n_i,

$$\begin{bmatrix} \delta d_1 \\ \vdots \\ \delta d_k \end{bmatrix} = G^T \delta V \tag{1}$$

$$G \triangleq \begin{bmatrix} n_1 & \cdots & n_k \\ p_1 \times n_1 & \cdots & p_k \times n_k \end{bmatrix}, \tag{2}$$

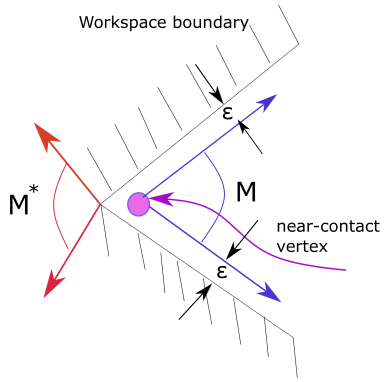

Fig. 1. Collision cone $\mathcal{M}^\star$ (in red) and free cone $\mathcal{M}$ (in blue) of a planar rigid body. The near-contact vertex of the object is drawn as a magenta disk. A red arrow represents a collision twist F_i.

where $k > 0$ is the number of near-contact points and G is referred to as the grasp matrix under frictionless contacts. To avoid collision with obstacles, δd_i has to satisfy

$$d_{i,0} - \delta d_i \geq 0, \tag{3}$$

or

$$\delta d_i \leq d_{i,0} \leq \epsilon \tag{4}$$

where $\epsilon \triangleq \max_i d_{i,0} > 0$. As a matter of fact Eqn. (1) and (2) is a first-order model of the variation of the gaps, which is only accurate for sufficiently small ϵ (hence the name "near-contact model"). Since a small positive offset δd_i of p_i along n_i might cause collision, we are interested in the set of twists for which p_i moves along $-n_i$.

Define

$$\mathcal{M} \triangleq \{\delta V \in se(3) \mid \delta d_i = \delta V^T F_i \leq 0, \forall i\}, \tag{5}$$

where $F_i \triangleq \left[n_i^T, (p_i \times n_i)^T\right]^T$ are column vectors of G. As shown in Appendix A, elements of $\mathcal{M}$ are free twists in the sense that for a finite motion of a given period of time generated by these twists, the rigid body will not collide with any obstacles. Contrarily F_i are collision twists that reduce the original gaps $d_{i,0}$ and finally lead to the collision. Next we define the dual $\mathcal{M}^\star$ of $\mathcal{M}$

$$\mathcal{M}^\star \triangleq \{F \in se(3) \mid \delta V^T F \leq 0,\ \forall \delta V \in \mathcal{M}\}. \tag{6}$$

Lemma 1. *Both $\mathcal{M}$ and $\mathcal{M}^\star$ are convex cones. $\mathcal{M}^\star =$ CONE(G), the conic combination[1] of all collision twists. $\mathcal{M}$ is the polar cone of $\mathcal{M}^\star$, and vice versa. Moreover $\mathcal{M}$ itself is a conic combination of the set of free twists $H =$*

[1]A conic combination of a set of points is defined as a linear combination of these points with non-negative coefficients.

TABLE I
LIST OF NOTATIONS USED IN THIS PAPER

Notation	Explanation
$SE(3)$	special Euclidean group in $\mathbb{R}^3$
$se(3)$	Lie algebra of $SE(3)$
exp	exponential map from $se(3)$ to $SE(3)$
δV, δV_i	elements of $se(3)$
$\mathcal{C}$, $\mathcal{C}_{\text{obst}}$	C-space, C-space obstacle,
c	element of $\mathcal{C}$
g, g_i, $\hat{g}_i$	element of $SE(3)$
$T_c\mathcal{C}$	Tangent space of $\mathcal{C}$
$\mathcal{C}_{\text{free}}$	collision-free portion of C-space
m	number of links in an open/closed chain
n	number of obstacles in workspace
$\{L_0, L_1, \cdots, L_m\}$	links in an open/closed chain
$\mathcal{O} = \{\mathcal{O}_i\}$	set of obstacles in workspace
$\{p_1, \cdots, p_k\}$	near-contact point set
n_i	normal vector of an object at p_i
G	grasp matrix
$\mathcal{M}(g)$	escaping cone in $se(3)$
$\mathcal{M}^\star(g)$	polar cone of $\mathcal{M}(g)$
$\mathcal{M}_c$	escaping cone in $T_c\mathcal{C}$
$\mathcal{M}_c^\star$	polar cone of $\mathcal{M}_c$
$\{d_{i,0}, \cdots, d_{k,0}\}$	initial gaps at near-contact point set
ϵ, $\epsilon(g)$	narrowness metric for a restrained link
δd_i	variation of gaps at near-contact point set
δV, δV_i, δV_j	twist of a rigid body in body frame
R(B), N(B)	range/null space of matrix B
F_i	collision twist in $R(G)$
g_{i_j}	configuration of link L_{i_j}
f_{i_j}	forward kinematic map from $\mathcal{C}$ to g_{i_j}
Jf_{i_j}	Jacobian matrix of f_{i_j}
L^u	upper bound of arc length parameter
L^l	lower bound of arc length parameter
L	$L^u + L^l$
γ	map of chain geometry in 3-D space
γ_c	map of chain geometry at c
γ_c^{-1}	inverse of γ_c
$\gamma_{(c,\lambda)}$	deformation of γ_c by factor λ
Π	projection map to escaping cone

$\{\delta V_j\}$, *i.e.*, $\mathcal{M} = \text{CONE}(H)$. *The decomposition of* $se(3) = \mathcal{M} + \mathcal{M}^\star$ [2] *is called a polar-cone decomposition.*

Proof: See Appendix B. □

We refer to $\mathcal{M}$ as the *free velocity cone*, while $\mathcal{M}^\star$ as the *collision cone*. Fig. 1 illustrates $\mathcal{M}$ and $\mathcal{M}^\star$ of a planar rigid body with 1 near-contact vertex.

Lemma 2. *If the set of near-contact points* $\{p_i\}$ *is* form-closure *[33], i.e., the convex hull of the column vectors of* G *strictly contains the origin of* $se(3)$*, then* $\mathcal{M} = \{0\}$.

A collision-free configuration $g \in SE(3)$ is said to be *narrow* if the associated free velocity cone generated by the set of all near contacts $\mathcal{M}(g)$ degenerates into a lower-dimensional cone, i.e.,

$$\dim \mathcal{M}(g) < 6. \tag{7}$$

For planar linkages we substitute $g \in SE(2)$ and $\dim \mathcal{M}(g) < 3$ in Eqn. (7).

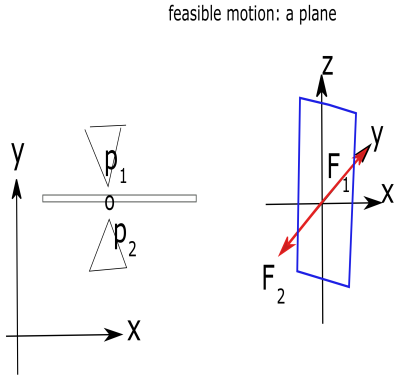

Fig. 2. An object with 2 near-contact points, for which $\mathcal{M} = \{x, z\}$ (drawn as a blue plane) includes translations along x-axis and rotations around z-axis. Red arrows are collision twists along y-axis. This configuration is narrow becaues $\dim(\mathcal{M}) < 3$.

Proposition 1. *A collision-free configuration* g *of a rigid body is narrow if and only if* $\mathcal{M}^\star(g)$ *contains a non-empty subspace.*

Proof: See Appendix C. □

A path $g(t) \in SE(3)$ is said to pass through a narrow passage if there exists at least one t_0 such that $g(t_0)$ is narrow. Eqn. (7) indicates that once the robot reaches $g(t_0)$, feasible samples in its neighborhood is limited to a lower dimensional cone $\mathcal{M}(g(t_0))$.

Example 1. 1-D and 2-D narrow passages for a planar rigid body

When $\mathcal{M}(g)$, $g \in SE(2)$ is a 3-D cone, g is certainly not narrow. The only possibilities are when $\mathcal{M}(g)$ degenerates into a plane or a line. Fig. 2 shows a narrow configuration under 2 near-contact points for which $\mathcal{M}(g)$ is a plane. If there are 3 or more non-colinear near-contact points, the only non-trivial narrow passage is given in Fig. 3, where the free velocity cone is a 1-D translation along x axis, while the collision cone is a plane drawn as a red rectangle. [3]

Example 2. 1-D narrow passage of a rigid body in 3-D space.

In 3-D space, the free velocity cone of a rigid body is

[2] Here we adopt the Minkowski sum of two given sets.

[3] If the object is circular, and moreover there are 3 or more near-contact points on its boundary whose convex hull contains the center, then the circle can only rotate around its center. However, this case is regarded as trivial here.

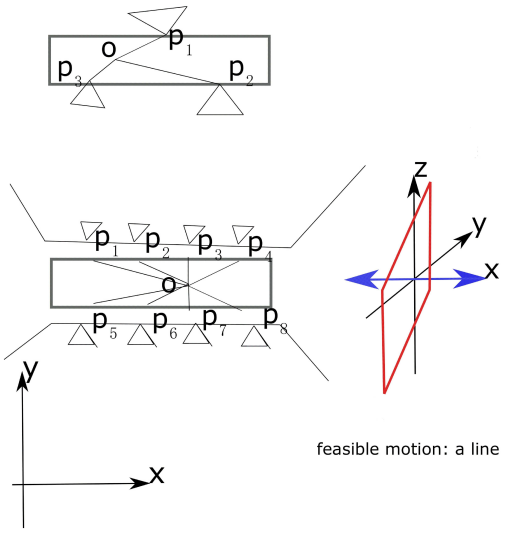

Fig. 3. An object with 3 or more non-colinear near-contact points can only translate along x-axis (drawn as a blue line).

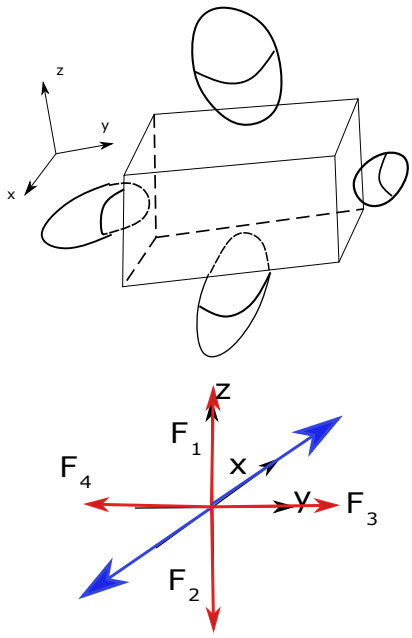

Fig. 4. An object with two pairs of near-contact points, each with their normals along the same line.

generally a 6-D cone in $se(3)$ for 1-6 near-contact points. This cone can degenerate into a lower dimensional one. One such example is given in Fig. 4, where there are two pairs of near-contact points, so the rigid body can only translate along x axis.

B. Quantitative Measures and Projection Maps

A narrow configuration g can be quantitatively measured by the dimension $\dim(\mathcal{M}(g))$ of $\mathcal{M}(g)$, and the maximal value $\epsilon(g)$ among all initial gaps $d_{i,0}$ at p_i, i.e.,

$$\epsilon(g) \triangleq \max_i d_{i,0}(g). \tag{8}$$

The former quantity is referred to as the *dimension of narrowness*, while the latter as the *degree of narrowness* [4].

Depending on the geometries of workspace obstacles and the considered rigid body, a narrow passage in C-space can assume complex and possibly continuous shape, e.g., a long narrow corridor or a maze in 2-D case. To study the geometry of such narrow passage as a whole, we associate a free velocity cone and a projection map to every narrow configuration in the passage. Let g_0 be a known collision-free narrow configuration in a narrow passage, $\mathcal{M}(g_0)$ the free velocity cone associated with g_0. Consider a small neighborhood $U_{g_0} \subset SE(3)$ of g_0 such that for every $g_1 \in U_{g_0}$ we have

$$g_1 = g_0 \exp^{\delta V}, \quad \delta V \in se(3). \tag{9}$$

Recall that $\mathcal{M}_{g_0}$ is the cone $\{\delta V \in se(3) \mid G^T \delta V \leq 0\}$. We define the following map from g_1 onto the surface of $\mathcal{M}(g_0)$

$$\Pi(g_1) = g_0 \exp^{(I - G(G^TG)^{-1}G^T)\delta V}. \tag{10}$$

Π is also a valid map even when $\mathcal{M}(g_0)$ degenerates into a subspace (see Example 1 and 2). The map $I - G(G^TG)^{-1}G^T$ has been extensively studied in [17], [18], in particular it has deep connection with the curvature of the underlying passage. It can be regarded as a precise formulation of PCA-based projection map [40], in which the map is estimated based upon a set of random samples in a neighborhood around g_0.

C. Narrow passage and free velocity cone of A Kinematic Chain

A kinematic chain is a bunch of links $(L_0, L_1, \cdots, L_m)$ connected together through joints, which are assumed here to be all revolute without loss of generality. When the chain does not contain closed loops, its C-space $\mathcal{C}$ is simply the set of all feasible combinations of joint angles and the configuration of the base link L_0. $\mathcal{C} = T^m \triangleq (S^1)^m$ if L_0 is a fixed base. Otherwise $\mathcal{C} = SE(3) \times T^m$, i.e., we treat L_0 as a free-floating link capable of taking any pose in $SE(3)$. On the other hand when the chain has closed loops, $\mathcal{C}$ becomes a sub-manifold embedded in the ambient space of the corresponding open chain without taking into account the loop-closure constraints.

Suppose a subset set of links $N_L = (L_{i_1}, \cdots, L_{i_k})$ are in narrow configurations $(g_{i_1}, \cdots, g_{i_k})$ respectively. They are simply called as the "restrained links" in this paper. To each restrained link L_{i_j}, $j = 1, \cdots, k$, there is an associated free velocity cone $\mathcal{M}_{i_j}$ and a collision cone $\mathcal{M}^\star_{i_j}$. Let f_{i_j} be the forward kinematic map from $\mathcal{C}$ to the configuration g_{i_j} of Link L_{i_j}, i.e.,

$$f_{i_j} : \mathcal{C} \to SE(3), c \to g_{i_j}, \tag{11}$$

and Jf_{i_j} the corresponding Jacobian of $f_{i,j}$. With these Jacobians, we are able to assemble the free velocity cones

[4]Notice that $\epsilon(g)$ is different from path clearance, which is often defined as $\min_i d_{i,0}$. Obviously an object with small path clearance (e.g., when it is close to an obstacle from one direction) does not mean it lies in a narrow passage.

and collision cones of N_L together and then map them into the corresponding cones, $\mathcal{M}_c$ and $\mathcal{M}_c^\star$ in $\mathcal{C}$. By simple calculation, we have

$$\mathcal{M}_c = \cap_j J f_{i_j}^{-1} \mathcal{M}_{i_j} \quad (12)$$

$$\mathcal{M}_c^\star = \sum_j J f_{i_j}^T \mathcal{M}_{i_j}^\star \quad (13)$$

assuming f_{i_j} is not singular.

Proposition 2. *Both $\mathcal{M}_c$ and $\mathcal{M}_c^\star$ are convex cones. A configuration $c \in \mathcal{C}$ of a kinematic chain is narrow if and only if N_L is not empty, and more over $\mathcal{M}_c^\star$ contains a non-empty subspace. In particular if the origin of $T_c\mathcal{C}$ is strictly contained in the interior of $\mathcal{M}_c^\star$, then the entire chain is "form closure", i.e., its dimension of narrowness is* 0.

Example 3. Open chain with 1 **restrained link**
If there is only one restrained link, say L_{i_1}, and suppose $J f_{i_1}^T$ is not singular, $J f_{i_1}^T \mathcal{M}_{i,1}^\star$ will contain a non-empty subspace as long as $\mathcal{M}_{i,1}^\star$ contains a non-empty subspace. See the case in Fig 5, where L_3 is in a narrow configuration. We conclude c of this chain is narrow based upon Proposition 2.

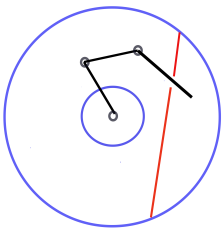

Fig. 5. Narrow configuration of a planar 3-link open chain with only 1 restrained link, L_3. Where red line lines are obstacles, and blue annular is the workspace of robot tip point.

Example 4. Narrow configurations of closed chains in 2**-D and** 3**-D spaces**
Fig. 6 illustrates a planar 9-link closed chain moving among two polygonal obstacles. The configuration g_4 of L_4 is narrow as already discussed in Section II-A and Example 1, and the corresponding free velocity cone is a 2-D subspace. Now break the chain into two subchains $\mathrm{CH}_L = \{L_0, L_1, \cdots, L_4\}$ and $\mathrm{CH}_R = \{L_9, L_8, \cdots, L_4\}$ with $L_9 = L_0$ being the fixed ground link. Let $J f_L$ and $J R_L$ be, respectively, the Jacobain of CH_L and CH_R. It is easy to check $N(J f_L^T) = \{0\}$ and $N(J f_R^T) = \{0\}$. Moreover $\mathcal{M}_c^\star = \begin{bmatrix} J f_L^T \\ J f_R^T \end{bmatrix} \mathcal{M}_4^\star$. c is certainly narrow as $\mathcal{M}_c^\star$ contains a non-empty subspace of the form $\begin{bmatrix} J f_L^T \\ J f_R^T \end{bmatrix} W$ with W a non-empty subspace of $\mathcal{M}_4^\star$. Fig. 7 gives another closed chain where the number of links is 6. In this example L_3 is in a narrow configuration, $\mathrm{CH}_L = \{L_0, \cdots, L_3\}$, $\mathrm{CH}_R = \{L_6, L_5, L_4, L_3\}$ with $L_6 = L_0$, and $N(J f_L^T) = \{0\}$ and $N(J f_R^T) = \{0\}$. So c of this chain is narrow too. Fig. 8

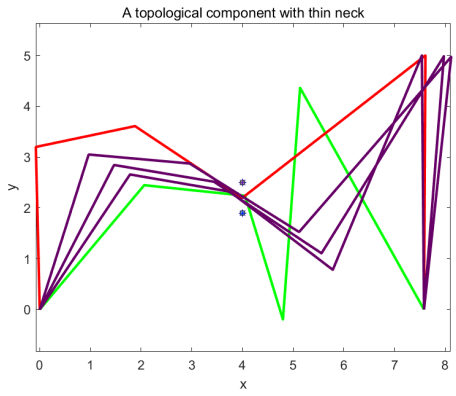

Fig. 7. Narrow configurations of a 6-bar closed chain

shows an 8-link spatial closed chain (assuming 0-thickness) passing through a toroidal obstacle of small inner-radius. As g_3 of L_3 (drawn as dashed line) is narrow, and moreover $N(J f_L^T) = \{0\}$ and $N(J f_R^T) = \{0\}$, c of the chain must be narrow.

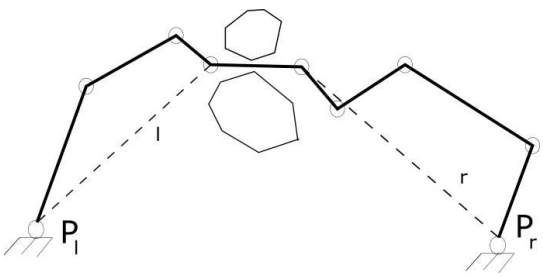

Fig. 6. Narrow configuration of a 9-link closed chain generated from a narrow configuration of link 4 (The link anchored at P_l is L_1, and that at P_r is L_8, and the link fixed to the ground is L_0).

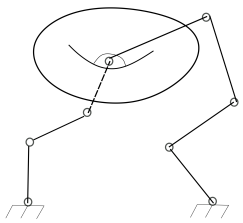

Fig. 8. Narrow configuration of an 8-link closed chain in 3-D space.

When $c_0 \in \mathcal{C}$ is narrow, it is possible to define a similar projection map that projects a sample c_1 in its

neighborhood onto the surface of the free velocity cone $\mathcal{M}_{c_0}$ at c_0. Let $\delta c \in T_{c_0}\mathcal{C}$ be a generalized velocity. It maps to a twist $Jf_{i_j}\delta c$ of a restrained link L_{i_j}. So $G_{i_j}^T Jf_{i_j}\delta c \leq 0$, for all restrained links $L_{i_j} \in N_L$, where G_{i_j} is the grasp matrix of L_{i_j}. Define $G_c \triangleq \left[Jf_{i_1}^T G_{i_1}, \cdots, Jf_{i_k}^T G_{i_k}\right]$, then we have

$$G_c^T \delta c \leq 0, \tag{14}$$

and the projection

$$\Pi(c_1) = c_0 + (I - G_c(G_c^T G_c)^{-1} G_c^T)(c_1 - c_0). \tag{15}$$

Π provides a powerful tool for exploring a long and curved narrow passage by applying projections to those samples in the neighborhood of existing narrow configurations, and growing the RRT tree inside the narrow passage.

III. Topological components of a Kinematic Chain

In this section, we propose the concept *topological components* for a kinematic chain which allows us to generate a coarse decomposition of $\mathcal{C}_{\text{free}}$ without a priori roadmap.

A. Deformation of a kinematic chain

Without loss of generality a kinematic chain can be represented as a map

$$\gamma : \mathcal{C} \times [-L^l, L^u] \to \mathbb{R}^3 \times \mathbb{R}^2, (c, s) \to (X(c, s), \Omega(s)) \tag{16}$$

where $X(c, s) \in \mathbb{R}^3$ gives the coordinates, as a function of the arclength parameter s, of every point on the 1-D backbone of the chain at c, L^l and L^u are the lower and upper bounds of s, $L = L^l + L^u$ is the length of $X(c, s)$, and $\Omega(s) \subset \mathbb{R}^2$ is a closed domain representing the cross-section of the chain at $X(c, s)$ which only depends on s. Throughout the paper, we assume $\Omega(s)$ is approximated by a circumscribing disk of radius r_s centered around $X(c, s)$, and thus the physical geometry of a kinematic chain by a surface of revolution around the backbone $X(c, s)$.[5] Moreover, we assume $L >> \max_s r_s$ so that the path planning problem for a kinematic chain is reduced to the problem for the curve X plus the collision checking using the approximation by the surface of revolution.

Given $c \in \mathcal{C}$, we define

$$\gamma_c : [-L^l, L^u] \to \mathbb{R}^3 \times \mathbb{R}^2, s \to \gamma(c, s), \tag{17}$$

and

$$\gamma_{(c,\lambda)} : [-\frac{L^l}{\lambda}, \frac{L^u}{\lambda}] \to \mathbb{R}^3 \times \mathbb{R}^2, s \to \gamma(c, \lambda s), \tag{18}$$

a deformed chain. λ is referred to as the deformation scale. The deformation from the original chain $\gamma_{(c,1)}$ all way to $\gamma_{(c,\lambda)}$ is said to be collision-free if the volume occupied by the deformed chain has no overlap with workspace obstacles for all deformation scales in $[1, \lambda]$ if $\lambda \geq 1$ (or $[\lambda, 1]$ if $\lambda \leq 1$). Sometimes we might decompose the entire chain into several subchains (i.e., divide $[-L^l, L^u]$ into several intervals), each deforms within its corresponding interval to avoid possible collision. It is always possible to choose the different scales for different subchains so that they still constitute a complete chain without any gaps after performing deformations upon individual subchains.

$\mathcal{C}_{\text{free}}$ can now be decomposed into components based upon the equivalence relation after deformations. Two configurations $c_1, c_2 \in \mathcal{C}_{\text{free}}$ are said to be in the same *topological component* if they can be broken into the same number of subchains, and moreover, there exists a motion $c(t)$ between c_1 and c_2 that moves each subchain after collision-free deformations to their desired configurations without colliding with obstacles and breaking the interconnection between consecutive subchains. The following Lemma states the relationship between topological components and the actual connected components of $\mathcal{C}_{\text{free}}$.

Lemma 3. *If $c_1, c_2 \in \mathcal{C}_{\text{free}}$ lie in two different topological components, they must belong to different connected components of $\mathcal{C}_{\text{free}}$.*

Example 5. Open chains with and without fixed base

In Fig. 9, two configurations c_1 and c_2 of a 4-DoF free-floating open chain are given. γ_{c_1} and γ_{c_2} can be deformed to a corresponding "mini-chain" $\gamma_{(c_1,\lambda)}$ and $\gamma_{(c_2,\lambda)}$ without collision for a given scale $\lambda > 1$. It is quite simple to verify that c_1 and c_2 for the mini-chain are connected. So they are in the same topological component (although they are also in the same connected component of $\mathcal{C}_{\text{free}}$).

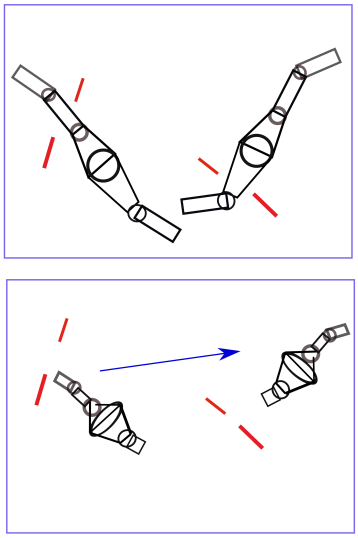

Fig. 9. Two configurations of an open chain with free-floating base in the same topological component

Fig. 10 shows three configurations of a 9-DoF free-floating open chain. The two configurations drawn as black (c_1) and blue (c_2) are in the same topological component. The configuration drawn as purple (c_3) is, however, not in the same topological component as the previous two

[5] For many industrial manipulators commercially available, this is a reasonable simplification in motion planning and control. Our path planning algorithms, developed in later sections, are even valid without this assumption.

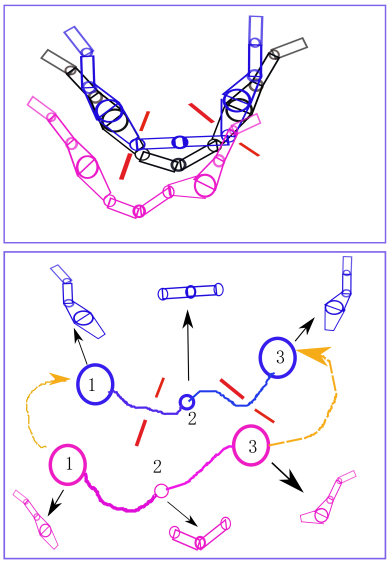

Fig. 10. Top: Three configurations of an open chain with floating base. Configurations drawn as black and blue are in the same topological component. The configuration in purple and the other two are not belong to the same topological component; Bottom: The chain is broken into three subchains, illustrated as three balls with different radii interconnected with unbreakable string.

configurations. This latter result can be deduced through a topological argument. We brake the chain into three subchains as illustrated at the bottom of Fig. 10. We represent the first and last subchain as a large ball to indicate that they can not pass the two narrow passages. The middle subchain is represented as a small ball because it can pass through both narrow passages. For every possible motion $c(t)$ (shown as yellow arrows) that moves the two large balls (i.e. the first and last subchains) from their corresponding configurations at c_3 to those at c_2, the backbones of γ_{c_3} and γ_{c_2} as well as the two yellow paths forms a closed curve that can not be retracted to a point due to the obstruction from the obstacles. Therefore, c_3 is not in the same topological component as c_1 and c_2. Appendix D shows that there are totally 23 topological components for this open chain under the deformations upon the three subchains.

The closed-loop structures of closed chains introduce new types of topological components, the analysis of which, however, requires sophisticated tools from topology which are out of the scope of this paper. Interested readers can refer to Appendix E for the related theories.

Example 6. Topological components of closed chains in 2-D space

Fig. 11 shows a 12-bar planar closed chain with zero (or very small) thickness whose workspace contains 2 point obstacles. When self-intersection is not allowed, there are only 5 topological components. Configurations in these 5 components are drawn as different colors in Fig. 11.

Example 7. Topological components of closed chains in 3-D space

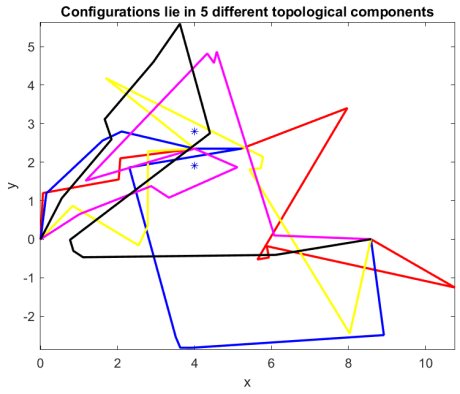

Fig. 11. A 12-bar closed chain with 2 point obstacles has 5 topological components if self-intersection is not allowed.

Fig. 12-top illustrates 4 configurations of a 10-bar spatial closed chain with spherical joints. They are labeled as c_k, c_b, c_y, c_m, respectively based upon their black, blue, yellow, and magenta colors. Applying the theories in Appendix E, these four configurations belong to four different topological components respectively. Fig. 12-bottom illustrates two configurations (yellow and blue) of a spatial 7-bar closed chain. They are in the same topological components if the links are thin enough to pass through the gap between two cylindrical obstacles. This narrow gap manifests the existence of a narrow neck in the corresponding topological component.

B. Relation to workspace medium axis and restrained links

With the simplified model of a kinematic chain as a piecewise linear backbone $X(c, s)$ and the collision model of the surface of revolution around $X(c, s)$, it is desirable for $X(c, s)$ to align with the medium axis of the workspace to avoid possible collisions with workspace obstacles, obtained, for example, from the method of Voronoi diagram. However, computation of the complete set of medium axis is expensive [42]. On the other hand complete alignment of $X(c, s)$ with the medium axis is not only impossible sometimes, but also unnecessary. Our key observations here are

- Narrow configurations c of the chain can be sampled by enumerating a subset of restrained links according to Proposition 2 and aligning them with their corresponding medium axis in the workspace;
- The medium axis for a restrained link can be obtained in the preprocessing phase of workspace obstacles by identifying small-gap vertex-face pairs based upon the known maximal radius of the link;
- Links which are not belong to the set of restrained links are free to move as long as they satisfy the kinematic constraints of the chain;
- The poses (i.e. the position and orientation) of the set of restrained links determine the topological component to which the narrow configuration c of the entire

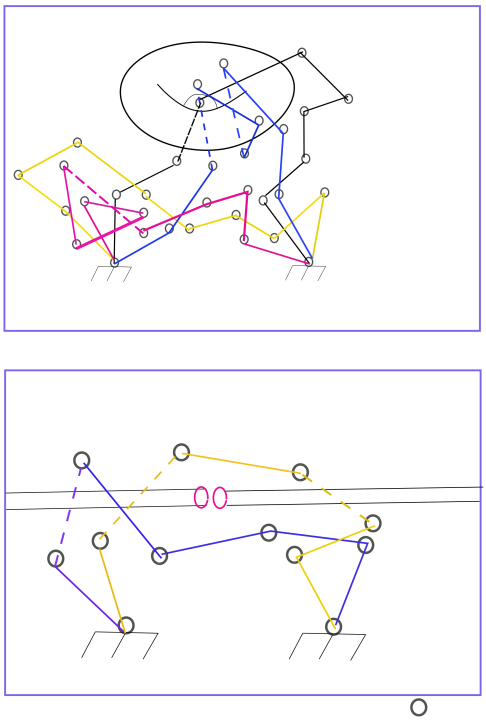

Fig. 12. Top: Four configurations of a 10-bar spatial closed chain that lie in 4 different topological components. Bottom: Two configurations of a 7-bar spatial closed chain sitting in a workspace with two cylindrical obstacles with a small gap. They are in the same topological component with a narrow neck.

chain belongs to according to our analysis in Section III-A.

- Starting from a narrow configuration c, the corresponding narrow passage in C-space can be explored by projecting nearby collision or non-collision samples onto the free velocity cone $\mathcal{M}_c$ at c via the projection map in Eqn. (15).

The above observations will be heavily used in a new path planning algorithm introduced in the next section.

IV. A new RRV-type algorithm based upon free velocity cones and topological components

In this section we propose a new random sampler that combines a high-level subchain combinatorics and a low-level sampling algorithm of the narrow configurations for the set of restrained links. These samples and the associated projection maps are then plug into the framework of the PRM and RRT algorithm to obtain efficient new PRM-type and RRT-type algorithms.

A. Identify restrained links

Throughout the paper we assume workspace obstacles are convex or non-convex polyhedral objects. As non-convex polyhedral obstacles can be decomposed into convex polyhedral pieces [12], [13], it is assumed here that we are given a list convex polyhedral obstacles $\{\mathcal{O}_j\}$.[6]

Suppose the maximal radius of L_i is r_i and except for the constraints imposed by the robot kinematics (e.g. the loop closure constraints) each link is free to rotate. Possible workspace narrow passages for L_i can be identified by finding small-gap vertex-face pairs, e.g. up to $(1+\epsilon_r)r_i, \epsilon_r > 0$, each from a different obstacle respectively. The searching for small-gap vertex-face pairs is achieved by the k-nn search algorithm, as given in Algorithm 1. Notice that this algorithm only needs to be carried out once in the pre-processing phase.

Algorithm 1: Workspace narrow passage identification of L_i

Require: convex polyhedra
$FV_j \triangleq \{FV_j.vertices, FV_j.faces\}$ representation of $\mathcal{O}_j$
Ensure: find all vertex-face pairs with gaps less than $(1+\epsilon_r)r_i$
for Every obstacle $\mathcal{O}_j$ **do**
  for Every vertex v_a^k in $FV_j.vertices$ **do**
    Using k-nn search algorithm to find the closest set of vertices $\{v_b^l\}$ from non-adjacent polyhedra.
    Compute the distance between v_a^k and every face that contains v_b^l.
    Keep the pairs for which the distance is less than $(1+\epsilon_r)r_i$.
  end for
end for

B. Sampling narrow configurations of a restrained link L_i

For each narrow vertex-face pair for L_i, the workspace medium axis can be approximated by the portion S of the plane which equally separates the pair while contained in the tetrahedron formed by the pair. S is called a central triangle here. Therefore the center p_c^i of link L_i can be randomly chosen from S, and the backbone of L_i be chosen as parallel to the face in the pair. See Fig. 13. In case that two pairs of narrow vertex-face pairs overlap, we compute the intersection of their respective central triangles and often a line segment (called central line segment) is obtained. Then p_c^i is randomly picked from the segment, and the backbone of L_i is made aligning with the line segment. Such random narrow configuration of L_i is then passed to the reachability test using the kinematics of the chain. The final algorithm for sampling reachable narrow configurations of L_i is given in Algorithm 2.[7]

C. Random enumeration of restrained links

Topological components of a given kinematic chain is closely related to the set of restrained links, as discussed

[6] Pairs of convex polyhedral obstacles might be adjacent along a common boundary face.

[7] For kinematic chains whose link geometries are not surface of revolution, p_c^i is sampled from the neighborhood of the central triangle or line, while R_i is randomly picked from $SO(3)$.

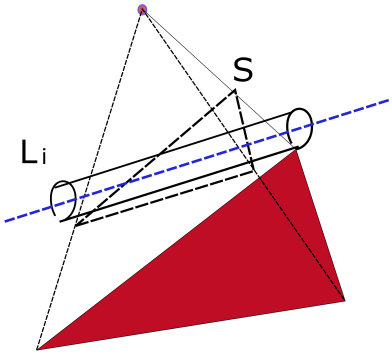

Fig. 13. Narrow configuration of L_i. A narrow vertex-face pair is drawn as a magenta disk and a red triangle; The backbone of L_i is aligned with the central triangle S drawn in dashed thick black lines.

Algorithm 2: A random sampler of the narrow configurations of link L_i

Require: The set of narrow vertex-face pairs between obstacles, and the kinematic model of all related subchains that determines the reachable set of L_i
Ensure: Generate a set of random samples of L_i which are both reachable and narrow.
for Every isolated narrow vertex-face pair **do**
Randomly pick p_c^i from the central triangle between the vertex-face pair
Pick a random orientation R_i of the backbone of L_i so that it is parallel to the face in the pair
Check the reachability of $g_i = (p_c^i, R_i)$ and discard it if it is unreachable
end for
for Every pair of overlapping narrow vertex-face pairs **do**
Randomly pick p_c^i from the central line segment, and set R_i so that the backbone of L_i aligns with the segment
Check the reachability of $g_i = (p_c^i, R_i)$ and discard it if it is unreachable
end for

in Section III, and Example 5-7. We might randomly pick k link indices $\{i_1, \cdots, i_k\}$ from a candidate set A_L of links, and associate them randomly in one-to-one correspondence with the set of narrow vertex-face pair. Consider the kinematic constraints of the chain, $A_L = \{L_0, \cdots, L_m\}$ for free-floating open chains, $A_L = \{L_{n_0}, \cdots, L_m\}$ for fixed-base open chains, and $A_L = \{L_{n_0}, \cdots, L_{m-n_0}\}$ for closed chains with $n_0 = 3$ for 2D chains, and 6 for 3D chains. Moreover, $i_{k+1} - i_k \geq n_0$ to have feasible IK solutions for the subchain from L_{i_k} to $L_{i_{k+1}}$. Hence $k \leq |A_L|/n_0$, where $|A_L|$ denotes the cardinality of A_L. These k links satisfying the above constraints are referred to as an *admissible link set*. Combining the enumeration of admissible link set, the corresponding set of narrow vertex-face pairs, and Algorithm 2, we obtain a new random sampler specifically for generating narrow configurations of a kinematic chain, as given in Algorithm 3. We remark here that the most expensive step in Algorithm 3 is computing the IK for $k+1$ subchains given the configurations of k restrained links. For spatial kinematic chains, in worst case the IK requires solving the roots of a 16-degree polynomial, although there are numerical algorithms available [43], [44].

Algorithm 3: A new random sampler based upon the free velocity cones and topological components

Require: A kinematic chain $(L_0, \cdots, L_m)$ and n obstacles $\mathcal{O}_i$, $i = 1, \cdots, n$
Require: A set of links A_L which might be at narrow configurations
Require: A set of narrow vertex-face pairs between pairs of obstacles
Ensure: Generate N narrow configurations for the chain
Initialize numSample=0
while numSample $< N$ **do**
Randomly choose an admissible link set $\{i_1, \cdots, i_k\}$ from A_L, and assign them to a corresponding set of non-repeating narrow vertex-face pairs (also chosen randomly).
Break the chain into $k+1$ subchains $\{i_j, \cdots, i_{j+1}\}$, $j = 0, \cdots, k$, $i_0 = 0$, and $i_{k+1} = m$.
Randomly sample the narrow configurations of every link in $\{i_1, \cdots, i_k\}$ using Algorithm 2.
Fixed the configurations of these links, call the IK of the $k+1$ subchains to obtain the configurations of the remaining links
end while

D. A new RRV-type algorithm

We might combine the samples from Algorithm 3, called topological samples, with those by purely random methods, called regular samples, and then plug them into any RRT-type algorithm, e.g., the RRT-CONNECT algorithm [41]. However, topological samples are expensive compared with the regular ones. In case of a long and pathologically curved narrow passage, a substantial amount of topological samples are required to reasonably cover the passage. To reduce the computation time, we follow the RRV algorithm [40] to project any samples (either topological, or regular samples in $\mathcal{C}_{\text{free}}$ or $\mathcal{C}_{\text{obst}}$), for which the nearest leaf node is a narrow configuration and more over the extension from the leaf toward the sample itself traps locally, onto the free velocity cone of the leaf node using the map in Eqn. (15). Afterwards, a new extension is carried out from the same leaf node toward the projected sample. As the forward Jacobian involved in Eqn. (15) is easy to compute for a known configuration [33], more

samples can be quickly generated in the neighborhood of a existing narrow sample. A key step, Extend(q_{near}, q_{rand}), of this algorithm is summarized in Algorithm 4, in which RRT-Extend() is the extension function in the original RRT algorithm [41], [46]. This new RRV-type algorithm is named M-RRV algorithm in Section V.

Algorithm 4: Extend(q_{near}, q_{rand})

```
if RRT-Extend(q_near, q_rand) is not trapped then
  Insert q_new=RRT-Extend(q_near, q_rand) into the tree
else
  if q_near is narrow then
    q_new = Π(q_rand)
    if RRT-Extend(q_near, q_new) is not trapped then
      Insert q_new1=RRT-Extend(q_near, q_new) into the
      tree
    end if
  end if
end if
```

Similarly, we might enhance the existing PRM algorithms, e.g., the OBPRM [19], by including the topological samples and applying the projection map (15) over all samples whose connection with their close neighbors fails, as given in Algorithm 5. Notice that the CONNECT function stands for the local planner. It returns a free milestone up to a given distance (similar to the RRT algorithm[8]) from q^i_{near} if it fails to connect q_{rand} to q^i_{near}. This new milestone as well as its projection under Π are both added into the set of unprocessed samples, for which the CONNECT function toward their close neighbors will be applied later on.

Algorithm 5: CONNECT($\{q^i_{near}\}$, q_{rand})

```
for each q^i_near do
  if q_new1=CONNECT(q^i_near, q_rand) succeeds then
    Insert q_rand into the set of vertices of the PRM
    graph if it is not in the set yet
    Insert [q^i_near, q_rand] into the set of edges of the
    PRM graph
  else
    Add q_new1 to the set of unprocessed samples
    if q^i_near is narrow then
      Add q_new2 = Π(q_new1) to the set of unprocessed
      samples
    end if
  end if
end for
```

V. Experimental results

In this section we present the simulation results for two challenging problems: a 12-bar closed chain moving among two pairs of polygonal obstacles with the PRM-type algorithms, and a 6-DoF planar robot climbing in a vertical plane with the RRT-type algorithms. Please refer to https://github.com/guanfengliu/closechain for the codes and animations of these two examples as well as many other examples. Our software is implemented in Matlab, and run under Windows 10 and Intel Core $i7$.

Example 8. A 12-bar closed chain moving among two pairs of polygonal obstacles

A 12-bar closed chain has 9 DoFs, and so can continuously move between two configurations which are pinched by two pairs of polygonal obstacles, as shown in Fig. 14, where the link length vector is [1.2000, 2.0000, 0.5512, 1.9457, 1.2131, 2.9482, 4.5684, 0.3000, 0.3000, 5, 2.5130, 8.5815]. All links are assumed to have same width 0.046. The coordinates of the centers of the two pairs of obstacles are, respectively, $[2.5, 1.6]^T$, $[2.5, 2.8]^T$, $[4.51.0]^T$, and $[4.5, 1.9]^T$. These obstacles are all squares with same side length $0.24/\sqrt{2}$. It takes 0.21 second in average for preprocessing the set of obstacles, and finding all narrow vertex-edge pairs. We apply Algorithm 3 for generating the topological samples and Algorithm 5 for building the adjacency graph. As we choose joint angles for parameterizing the C-space, standard Euclidean metric is subject to the issue of discontinuity in determining the set of near neighbors of a given sample. In this and next examples, we adopt the metric

$$d(c_1, c_2) = (\sum_j \|p^j(c_1) - p^j(c_2)\|^2)^{1/2}, \forall c_1, c_2 \in \mathcal{C} \quad (19)$$

where $p^j(c)$ denotes the Cartesian coordinates of the tip of link j at c, $j = 1, \cdots, m-1$ of an m-link closed chain. The resulting roadmap successfully find a path between the start configuration [1.0631, 0.3865, 1.4542, − 1.2802, − 2.5472, 0.5913, − 1.2102, 0.5527, 2.4609, 0.3069, 2.7463, 3.1416] and the goal configuration [2.2410, 0.8287, 1.0704, − 0.2046, − 1.6302, 0.2022, − 2.2898, 1.1214, 1.6813, − 0.2606, 1.1309, 3.1416].

The performance of Algorithm 5 is compared with that of OBPRM + KBPRM for the same problem, as given in Table II. In this experiment we maintain a fixed number of regular samples based upon a uniform distribution on a subset of the joints of the closed chain, and then applying the IK to obtain the joint angles of the remaining subchains. Certainly, some of them will have no IK solution, others will be rejected in collision checking. Until a solution path is found, we gradually increase the number of samples near $\mathcal{C}_{\text{obst}}$ for OBPRM + KBPRM, while the number of topological samples for Algorithm 5. The average number of samples in each category, the average number of connected components in the resulting roadmap, as well as the average running time for finding a solution, are all summarized in Table II. Notice that q_{new1} generated by a failed CONNECT function of Algorithm 5 can be regarded as a type of samples near $\mathcal{C}_{\text{obst}}$, while the projected sample q_{new2} is

[8]Recall the projection map Π is only valid in the neighborhood of a narrow configuration.

often a narrow configuration. This problem is challenging because the entire connected component to which the start and goal configurations belong has a small volume measure using the same argument as Example 4.

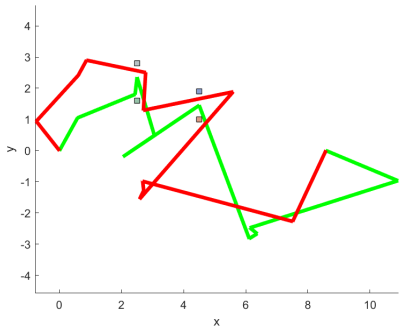

Fig. 14. Start and goal configurations confined by two pairs of polygonal obstacles with small gaps

TABLE II
PERFORMANCE COMPARISON BETWEEN DIFFERENT PRM ALGORITHMS

Algorithms	ORPRM+KBPRM	Algorithm 5
Regular samples	1386.3	1306.7
Near $\mathcal{C}_{\text{obst}}$ samples	134820.8	14253.5
Topological samples	0	40048.6
Components of roadmap	2714.8	2016.2
Run time	8005.8s	2267.3s

Example 9. A 6-DoF robot climbing on a vertical plane

Consider a 6-DoF planar robot climbing within a vertical plane with a set of holds and a number of polygonal obstacles, as shown in Fig. 15. The robot starts with an initial hold, and tries to find a sequence of holds toward a given goal hold, and the corresponding paths between any two consecutive holds in the sequence. When the robot grabs the goal hold, it continues moving until a goal configuration is achieved. We call the motion between two consecutive holds and the last motion after reaching the goal hold as a phase. Within each phase, robot base and tip links, and kinematic model keep constant. Upon its tip grabs a local "goal" hold, the robot gets into the next phase and a new kinematic model is used with the reversing order of links.

The setting of this problem is given as follows. The link length vector of the robot is simply set as $[1, 1, 1, 1, 1, 1]$. All links are assumed to have same width 0.8. The robot workspace contains three polygonal obstacles, and 10 holds whose coordinates are respectively $[2, 1.8]^T$, $[0.8, 3.5]^T$, $[-2.5, 4.5]^T$, $[-4.7153, 3.9781]^T$, $[-6.6131, 2.3139]^T$, $[-3.6058, 0.5766]^T$, $[1.0073, 5.9927]^T$, $[-1.2993, 7.6131]^T$, $[-4.3650, 8.2051]^T$, and $[4.6861, 7.1898]$.

The pre-processing phase of our algorithm calculates all vertex-edge pairs whose gap is less than 1.5 (a reasonable choice considering that the link width is 0.8), and yields two pairs $\{q_1^1, \overline{q_2^1 q_3^1}\}$ (drawn as blue disk and edge) and $\{q_1^2, \overline{q_2^2 q_3^2}\}$ (drawn as red disk and edge), both illustrated in Fig. 15. The distances between the vertex and the edge of both pairs are calculated. The minimal one is 1.1784, a value close to the link width 0.8, which makes the problem quite challenging to solve. Next, it creates a hold adjacency graph that connects pairs of holds whose distances are less than the maximal reach of the robot. With the graph, it searches a list of candidate paths between the initial and goal holds, arranged in the order of total path costs (in terms of the sum of the distances between consecutive holds). The entire pre-processing phase takes in average $0.47s$ including the time for finding all narrow vertex-edge pairs among obstacles, as well as that for searching for possible hold sequences. The run-time phase takes a path from the list of candidate paths, among which the shortest one is always executed first. When there is no feasible motion between any of the two consecutive holds in the sequence, the algorithm automatically switches to the next available hold sequence.

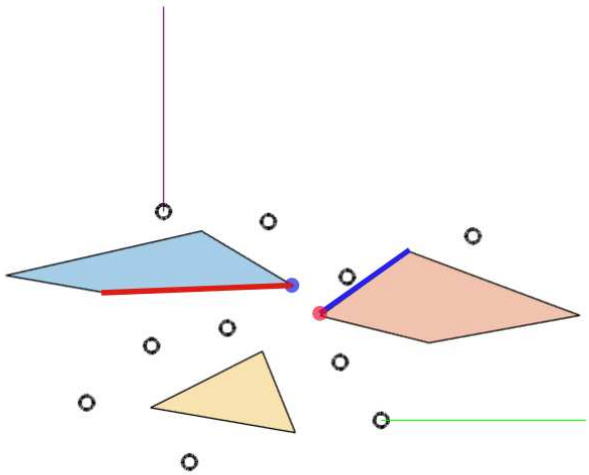

Fig. 15. A 6-DoF robot climbing in a vertical plane with 3 polygonal obstacles and 10 holds. The initial and goal configurations are respectively drawn as green and red. The pair of blue disk and edge, and the pair of red disk and edge are the two narrow vertex-edge pairs.

The motion between a pair of holds is generated by different variants of the RRT algorithm to compare their performances. The implemented algorithms include RRT-CONNECT, RRV, M-RRT-CONNECT, M-RRV (Algorithm 4), where M-RRT-CONNECT and M-RRV stand for the modified version of RRT-CONNECT and RRV by replacing a given portion of regular samples by topological samples from Algorithm 3. Furthermore, instead of using principal component analysis (PCA) method, M-RRV employs the free velocity cone for projecting rejected samples and extending the nearest leaf node toward the projected sample. In the experiments, we count the total

number of iterations, number of collision checks, number of rejected samples, number of accepted samples, and overall computation time. The results are summarized in Table III, in which all values are obtained from averaging the corresponding data from 10 experiments, and the results for M-RRV is obtained using 50% topological samples and 50% regular samples.

Remark 1. *For easy gait phases $1 \rightarrow 2$, $8 \rightarrow 9$, and $9 \rightarrow$ goal, the RRT-CONNECT algorithm is often the fastest one. This is due to the fact that topological samples and related projections are computationally expensive compared with regular samples, and at the mean time they do not help expedite the exploring process in open space.*

Remark 2. *Phase $2 \rightarrow 7$ is the phase that robot enters into a narrow passage from open space. Table III shows that M-RRT-CONNECT is about 8 times faster than RRT-CONNECT, RRV is about 5 times faster than RRT-CONNECT, and M-RRV is about 25 times faster than RRT-CONNECT. However, RRV is not as faster as M-RRT-CONNECT, which might be due to the fact that the C-space obstacles for the climbing robot are highly curved in T^6, and the estimation accuracy of the projection map by the PCA method is not as good as the Jacobian-based method in M-RRV.*

Remark 3. *Phase $7 \rightarrow 8$ is the phase that robot navigates through a narrow passage and finally exits the passage. Table III shows that M-RRT-CONNECT only improves marginally on RRT-CONNECT. This is because this phase starts from a configuration which is already inside a narrow passage, and topological samples themselves do not help to significantly expedite the navigation through the passage. RRV, due to the same reason of estimation accuracy, is not faster than RRT-CONNECT. M-RRV is 4 times faster than RRT-CONNECT, demonstrating the efficacy of Jacobian-based projection maps in joining and extending samples inside the narrow passage. Topological samples still play an important role here in M-RRV to spread out samples inside the narrow passage, which in turn helps to project rejected samples to the free velocity cone of the closest leaf node already inside the passage.*

Within the algorithm of M-RRV, we carried out experiments under different parameters such as the largest step size ϵ_Θ in extending a leaf node toward a newly generated sample, as well as the percentage P_t of topological samples. The results are given in Table IV and V. We see that for a fixed ϵ_Θ, in general the time for finding a feasible path reduces as the percentage of topological samples increases from 20% to 50%. However after the percentage goes above 50%, the computation time does not reduce anymore, but grows instead. This is because sampling of topological samples as well as the projection of rejected samples onto the free velocity cone are expensive operations compared with regular samples. The step size ϵ_Θ also affects the computation time of the algorithm. When $\epsilon_\Theta = 0.35(20°)$, Table V shows relatively larger computation time for Phase $2 \rightarrow 7$ and $7 \rightarrow 8$ for any percentage of topological samples. A reason for this is that the narrow passage involved in these two phases is highly curved in T^6, and a larger step size in extending a leaf node toward a newly generated sample (either a topological sample or the projection from a rejected sample) often leads to an invalid new leaf node. However, too small step size might cause slow extending of the RRT tree.

One might be also interested in the resulting paths by the M-RRV algorithm and whether they contain topological samples. These are confirmed in Fig. 16 , in which robot configurations arising from standard samples are drawn in brown, while those from topological samples are drawn in cyan. The degree of narrowness for all path milestones is depicted in Fig. 17-(a). As a reference, we also show the path clearance (i.e., minimal distance from obstacles) of all milestones in Fig. 17-(b). It is as expected that path narrowness is different from path clearance. There are totally 10 narrow samples on the path which are coming from topological samples, two of which are shown in Fig. 18. In fact further examining of the data shows that most of the way points on the path with close-to-zero narrowness are coming from topological samples, their local extensions, or from the projection of samples in collision to the free velocity cones of the samples of the previous two categories. This demonstrates the importance and effectiveness of topological samples and the projection maps onto their free velocity cones.

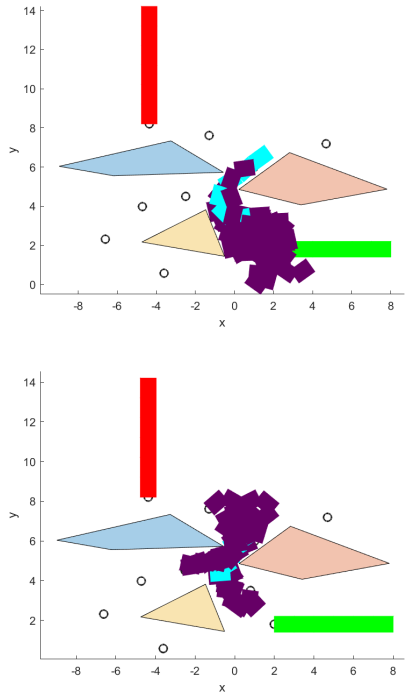

Fig. 16. Robot climbing path (link width=0.8) for two hard phases. Narrow topological samples are shown as cyan. (a): Phase $2 \rightarrow 7$; (b): Phase $7 \rightarrow 8$;

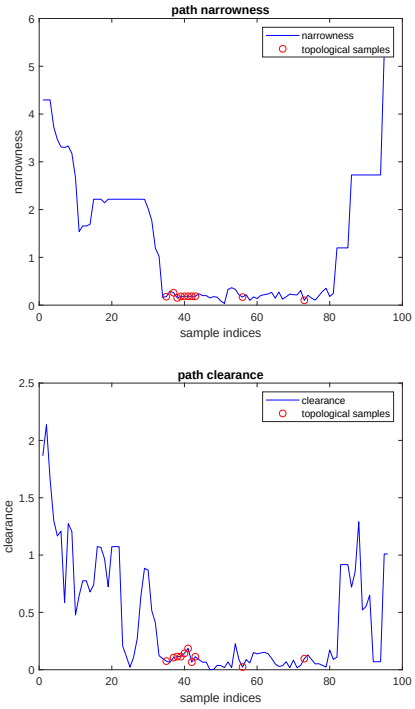

Fig. 17. (a): Narrowness of the milestones on the robot path; (b): Path clearance of the milestones on the robot path.

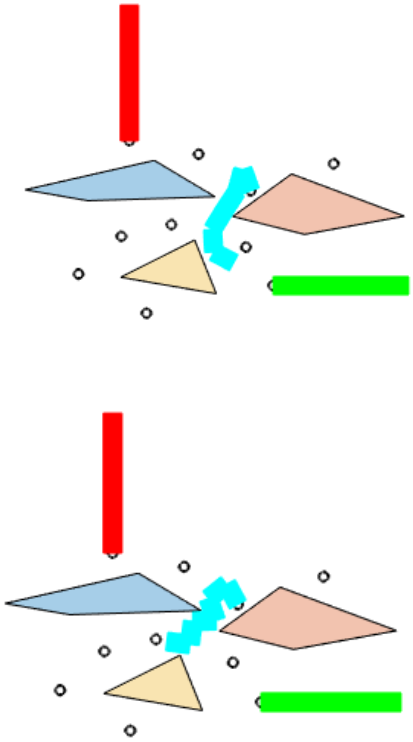

Fig. 18. Two samples (among 10) on the path which are topological samples. (a): link 3 is in narrow configuration; (b): link 4 is in narrow configuration.

VI. Conclusion

This paper presents the method of free velocity cones for exploring and navigating C-space narrow passages of a kinematic chain with arbitrarily DoFs. It is shown that narrow configurations of a single rigid link as well as the entire chain are completely determined by the degeneracy of the corresponding free velocity cones. Long curved narrow passages can be continuously explored from existing narrow configurations by projecting samples in their neighborhood onto the free velocity cones associated with them. This reduce the C-space narrow passage problem into a subproblem of sampling narrow configurations of a set of restrained links, and another subproblem of expanding known narrow configurations through projections. The former subproblem is solved by aligning the set of restrained links with workspace medium axis, and applying IK to deduce the configurations of the remaining links. While the latter subproblem is solved with a new PRM-type and RRT-type algorithm. The efficiency of our algorithm is demonstrated by comparing its performance with several existing variants of the PRM-type and RRT-type algorithms.

Acknowledgment

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

Appendix A
Geometric meaning of the twists in $\mathcal{M}$ and $\mathcal{M}^\star$

For a given twist δV of the rigid body, we consider the finite motion $\exp^{\delta V s}$, $s > 0$, generated by this constant twist, where exp denotes the exponential map on $SE(3)$, and s the arclength parameter of the curve $\exp^{\delta V s}$. Using the first order approximation of the exponential map, the following Lemma holds.

Lemma 4. *For any twist* $\delta V_i = \begin{bmatrix} n_i \\ p_i \times n_i \end{bmatrix}$, $i = 1, \ldots, k$, *there exist a small number* $s_0 > 0$ *such that the rigid body after a finite motion* $\exp^{\delta V s_0}$ *incurs a collision with at least one workspace obstacle. In contrast to these column twists of* G, *the twist* $\delta V_i \in \mathcal{M}$ *satisfies,*

$$\{\delta V \in se(3) \mid \delta d_i = n_i^T(\exp^{\delta V s} p_i - p_i) \leq 0, \forall s \in [0, s_1], \forall i\} \tag{20}$$

where s_1 *is significantly larger than* s_0.

Proof: For a given twist $\delta V = \begin{bmatrix} v \\ \omega \end{bmatrix}$, with $v, \omega \in \mathbb{R}^3$, and a near-contact point p_i, we have

$$\delta p_i = \exp^{\delta V s} p_i - p_i \approx (\omega \times p_i + v)s,$$

and then

$$\begin{aligned} \delta d_i &= n_i^T \delta p_i \approx n_i^T(\omega \times p_i + v)s = (F_i^T \delta V)s \\ F_i &= \begin{bmatrix} n_i \\ p_i \times n_i \end{bmatrix}. \end{aligned}$$

Therefore, if $\delta V = F_i$, we have $F_i^T \delta V = \|F_i\|^2 > 0$ and $\delta d_i \approx (F_i^T V)s > d_{i,0}$ for $s > s_0 \triangleq d_{i,0}/\|F_i\|^2$. On the contrary, if $F_i^T \delta V \leq 0, \forall i$, then we have $\delta d_i \leq 0, \forall i$ for $s \in [0, s_1]$, $s_1 > 0$. Notice that when the maximal gap $\epsilon = \max_i d_{i,0}$ is sufficiently small, the above first order approximation is accurate enough to claim that the twists in $\mathcal{M}$ are free twists, while those in $\mathcal{M}^\star$ are collision twists, and more over, $s_1 > s_0$.

Appendix B
Proof of Lemma 1

First, given $\delta V_j \in \mathcal{M}$, $\alpha_j \geq 0$, and each contact wrench F_i, we have

$$(\sum_j \alpha_j \delta V_j)^T F_i = \sum_j \alpha_j \delta V_j^T F_i \leq 0.$$

So $\sum_j \alpha_j \delta V_j \in \mathcal{M}$. Use the same argument, we see that $\mathcal{M} + \mathcal{M} \subset \mathcal{M}$. Therefore $\mathcal{M}$ is a convex cone. Similarly $\mathcal{M}^\star$ is a convex cone.

Second Eqn. (5) indicates that $\mathcal{M}$ is a polyhedral cone, i.e., the intersections of a finite set of half-spaces. Therefore it must be finitely generated by a finite set of twists $H = \{\delta V_j\}$, i.e., $\mathcal{M} = \text{CONE}(H)$ according to the Minkowski-Weyl theorem [6], [7]. Then $\mathcal{M}^\star$ in Eqn. (6) is equivalent to

$$\mathcal{M}^\star = \{F \in se(3) \mid \delta V_j^T F \leq 0, \forall j\}.$$

In other words, $\mathcal{M}^\star$ is a polyhedral cone too. Again applying the Minkowski-Weyl theorem yields that $\mathcal{M}^\star$ must be finitely generated too. To see $\mathcal{M}^\star = \text{CONE}(\{F_i\})$, we use the fact that $\mathcal{M} = \text{CONE}(\{F_i\})^\star$ based upon the definition of polar cone [6] and Eqn. (5), and the fact that $(\text{CONE}(\{F_i\})^\star)^\star = \text{CONE}(\{F_i\})$.

Appendix C
Proof of Proposition 1

First, if $\mathcal{M}^\star(g)$ contains a non-empty subspace $\mathcal{S}$, we choose $F, -F \in \mathcal{S}$. Then for any $\delta V \in \mathcal{M}(g)$, we have

$$\delta V^T F \leq 0, \quad \delta V^T F \geq 0,$$

so $\delta V^T F = 0$. Therefore $\dim(\mathcal{M}(g)) < 6$ and g must be narrow.

Second, if $\dim(\mathcal{M}(g)) < 6$, then there exists a subspace $\mathcal{S}$ which is orthogonal to $\mathcal{M}(g)$. Obviously $\mathcal{S} \subset \mathcal{M}(g)^\star$, as its elements satisfy Eqn. (6).

Appendix D
Number of topological components of a planar 9-DoF free-floating open chain

The number of possible topological components for the planar free-floating open chain in Fig. 10 can be enumerated based upon the locations of balls (left or right of each hole). Assume the chain can not pass through the narrow hole twice, there are 12 possible combinations without considering the order of ball 1 and 3, as shown in Fig. 19. The final number is 23 when the order is taken into account. The last configuration in Fig. 10 doesn't change its component if ball 1 and 3 get swapped.

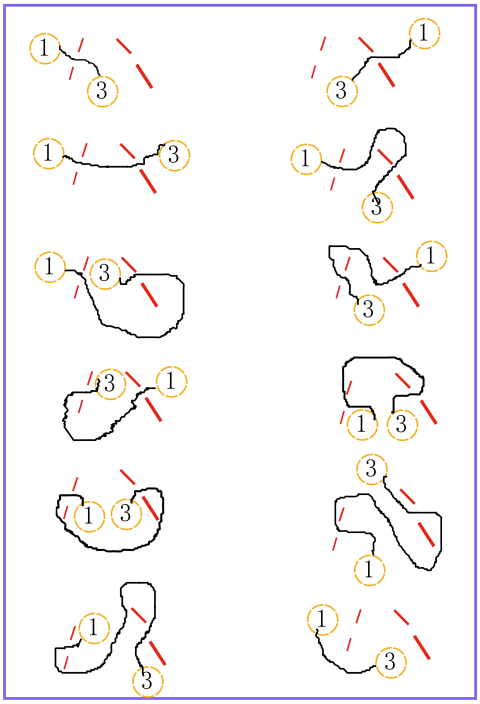

Fig. 19. 12 different combinations of two large balls interconnected with a string (representing three subchains of the planar open chain in Fig. 10).

Appendix E
Topological components of closed chains

We define the inverse γ_c^{-1} of γ_c as $\gamma_c^{-1}(s) = \gamma_c(L^u - L^l - s)$. It is easy to check $\gamma_c^{-1}(-L^l) = \gamma_c(L^u)$ and $\gamma_c^{-1}(L^u) = \gamma_c(-L^l)$. We have

Proposition 3. *Given $c_1, c_2 \in \mathcal{C}_{\text{free}}$, if the backbone of $\gamma_{c_1}^{-1} \circ \gamma_{c_2}$ can not be retracted to $\gamma_{c_2}^{-1} \circ \gamma_{c_2}$ without colliding with obstacles or self intersection, then they are not in the same topological component. In turn they are not in the same connected component of $\mathcal{C}_{\text{free}}$.*

Proof: Let $c(t), t \in [0, 1]$ be any collision-free path between c_1 and c_2. Then $\gamma_{c(t)}^{-1} \circ \gamma_{c_2}$ is a closed curve without collision and self intersection for all t. It starts from $\gamma_{c_1}^{-1} \circ \gamma_{c_2}$ and then gradually shrinks to $\gamma_{c_2}^{-1} \circ \gamma_{c_2}$, which has volume 0. When such retraction is impossible due to the obstruction of obstacles or self intersection, then any path $c(t)$ between c_1 and c_2 is not collision-free. So they must lie in two different topological components. □

When ignoring obstacles the only reason that prevents $\gamma_{c_1}^{-1} \circ \gamma_{c_2}$ from deforming into $\gamma_{c_2}^{-1} \circ \gamma_{c_2}$ is that the former closed curve forms a nontrivial knot. [9]

Corollary 1. *If $\gamma_{c_1}^{-1} \circ \gamma_{c_2}$ forms a nontrivial knot, then c_1, c_2 must lie in two different topological components.*

References

[1] O. Salzman, M. Hemmer and D. Halperin, *On the Power of Manifold Samples in Exploring Configuration Spaces and the Dimensionality of Narrow Passages.* IEEE Transactions on Automation Science and Engineering, Vol. 12, No. 2, pp. 529-538, April, 2015

[2] L. E. Kavraki, M. N. Kolountzakis, and J.C. Latombe, *Analysis of Probabilistic Roadmaps for Path Planning.* IEEE Transactions on Robotics, Vol. 14, No. 1, pp. 166-171, February, 1998.

[3] L. E. Kavraki, J.-C. Latombe, R. Motwani, and P. Raghavan, *Randomized query processing in robot path planning.* J. Comput. Syst. Sci, Vol. 57, No. 1, pp. 50-60, August 1998.

[4] D. Hsu, J. Latombe, and R. Motwani, *Path planning in expansive configuration spaces.* Int. J. Comp. Geo. and App., Vol. 4, pp. 495-512, 1999.

[5] M. Saha, J. Latombe, Y. Chang, Lin, and F. Prinz, *Finding narrow passages with probabilistic roadmaps: the small step retraction method.* Intelligent Robots and Systems, vol. 19, no. 3, pp. 301-319, Dec 2005.

[6] S.P. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge University Press, ISBN 978-0-521-83378-3, 2004.

[7] G. Ziegler, *Lectures on polytopes*, Springer-Verlag, 1997.

[8] J. Meng, G.F. Liu, and Z.X. Li, *A Geometric Theory for Synthesis and Analysis of Sub-six DoF Parallel Manipulators.* IEEE Transactions on Robotics, Vol. 23, No. 4, PP. 625-649, 2007.

[9] L.E. Kavraki, P. Švestka, J.C. Latombe, and M.H. Overmars, *Probablistic Roadmaps for path planning in high-dimensional configuration space.* IEEE Transactions on Robotics and Automation, 12(4):566-580, 1996.

[10] J. Yakey, S. M. LaValle, and L. E. Kavraki, *Randomized path planning for linkages with closed kinematic chains.* IEEE Transactions on Robotics and Automation, 17(6):951–958, December 2001.

[11] J. Cortes, *Motion planning algorithms for general closed-chain mechanisms.* Doctorat, Institut National Polytechnique, Toulouse, December 16, 2003, 160p.

[12] B. Chazelle, D. Dobkin, N. Shouraboura, and A. Tal. *Strategies for polyhedral surface decomposition: An experimental study.* Computational Geometry: Theory and Applications, 7:327-342, 1997.

[13] S. Ehmann and M. C. Lin. *Accurate and fast proximity queries between polyhedra using convex surface decomposition.* Computer Graphics Forum (Proc. of Eurographics'2001), 20(3):500-510, 2001.

[14] M. Zucker, J. Kuffner and J. A. Bagnell, *Adaptive workspace biasing for sampling-based planners*, IEEE International Conference on Robotics and Automation, PP. 3757-3762, 2008.

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

[16] J.C. Trinkle and R.J. Milgram, *Complete Path Planning for Closed Kinematic Chains with Spherical Joints.* International Jounral of Robotics Research, 21(9):773-789, December, 2002.

[17] Beobkyoon Kim, Terry Taewoong Um, Chansu Suh and F. C. Park, *Tangent bundle RRT: A randomized algorithm for constrained motion planning.* Robotica, 34, pp 202-225 doi:10.1017/S0263574714001234.

[18] G.F. Liu and Z.X. Li, *A Unified Geometric Approach to Modeling and Control of Constrained Mechanical Systems*, IEEE Transactions on Robotics and Automation, Vol. 18, No. 4, PP. 574-587, 2002.

[9] A knot is the embedding of a circle into $\mathbb{R}^3$. A knot is nontrivial if it can not be deformed into a circle without breaking itself.

[19] N. Amato, O. Bayazit, L. Dale, C. Jones, D. Vallejo, *OBPRM: An obstacle-basedPRM for 3D workspaces.* in: P.K. Agarwal, L.E. Kavraki, M.T. Mason (eds.), Robotics: The algorithmic perspective, A.K. Peters, Natick, 1998, pp. 155-168.

[20] C. Holleman and L.E. Kavraki, *A Framework for Using Workspace Medial Axis in PRM Planners*, Proceedings - IEEE International Conference on Robotics and Automation, PP. 1408-1413, 2000.

[21] D. Berenson, S. Srinivasa, D. Ferguson and J. Kuffnerr, *Manipulation Planning on Constraint Manifolds.* Proceedings - IEEE International Conference on Robotics and Automation, PP. 625-632, 2009.

[22] V. Vonasek, J. Faigl, T. Krajnik, L. Preucil, *A sampling schema for rapidly exploring random trees using a guiding path.* Proceedings of the 5th European Conference on Mobile Robots, vol. 1, pp. 201-206, 2011.

[23] D. Hsu, L.E. Kavraki, J.C. Latombe, R. Motwani, and S. Sorkin, *On finding narrow passages with probabilistic roadmap planners.* Proceedings of the third workshop on the algorithmic foundations of robotics: the algorithmic perspective, PP. 141-153, 1998.

[24] Zheng Sun, D. Hsu, T. Jiang, H. Kurniawati and J. H. Reif, *Narrow passage sampling for probabilistic roadmap planning.* IEEE Transactions on Robotics, vol. 21, no. 6, pp. 1105-1115, Dec. 2005.

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

[26] Jaillet L., Porta J.M., *Path Planning with Loop Closure Constraints Using an Atlas-Based RRT.* In: Christensen H., Khatib O. (eds) Robotics Research. Springer Tracts in Advanced Robotics, Vol. 100, PP. 345-362, 2017.

[27] C. Voss, M. Moll and L.E. Kavraki. *Atlas + X: Sampling-based Planners on Constraint Manifolds.* (2017) https://hdl.handle.net/1911/96421.

[28] G.F. Liu, J.C. Trinkle, Y. Yang, and S.M. Luo, *Motion Planning of Planar Closed Chains With Structural Sets*, IEEE Access, Vol. 8, PP. 117203-117217, 2020.

[29] J. Szkandera, I. Kolingerova, M. Manak, *Narrow Passage Problem Solution for Motion Planning.* In: V. Krzhizhanovskaya et al. (eds) Computational Science - ICCS 2020. Lecture Notes in Computer Science, Vol. 12137, PP. 459-470, Springer, Cham.

[30] A. Bicchi, *On the Closure Properties of Robotic Grasping.* International Journal of Robotics Research, Vol. 14, No. 4, PP. 319-334, 1995.

[31] M.Y. Wang, *An optimum design for 3-D fixture synthesis in a point set domain.* IEEE Transactions on Robotics and Automation, Vol. 16, No. 6, PP. 839-846, 2000.

[32] H. Kurniawati and D. Hsu, *Workspace importance sampling for probabilistic roadmap planning.* In Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 1618-1623, 2004.

[33] R.M. Murray, Z.X. Li, and S.S. Sastry, *A Mathematical Introduction to Robotic Manipulation*, CRC Press, 1994.

[34] E.G. Gilbert, D.W. Johnson, and S.S. Keerthi.*A Fast Procedure for Computing the Distance Between Objects in Three-Dimensional Space*, IEEE Transactions on Robotics and Automation, PP. 193-203.

[35] M. Farber, *Topological Complexity of Motion Planning*, Discrete Comput. Geom. 29:211-221,2003.

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

[37] L. Zhang and D. Manocha, *An efficient retraction-based RRT planner.* IEEE International Conference on Robotics and Automation, PP. 3743-3750, 2008.

[38] J. Lee, O.S. Kwon, L. Zhang, and S.E. Yoon, *A selective retraction-based RRT planner for various environments*, IEEE Transactions on Robotics, Vol. 30, No. 4, PP. 1002-1011, 2011.

[39] J. Pan, L. Zhang, and D. Manocha, *Retraction-based RRT planner for articulated models*, IEEE International Conference on Robotics and Automation, PP. 2529-2536, 2010.

[40] A. Tahirovic and M. Ferizbegovic, *Rapidly-Exploring Random Vines (RRV) for Motion Planning in Configuration Spaces with Narrow Passages*, IEEE International Conference on Robotics and Automation, PP. 7055-7062, 2018.

[41] J. J. Kuffner and S. M. LaValle, *RRT-connect: An efficient approach to single-query path planning*, IEEE International Conference on Robotics and Automation, PP. 995-1001, 2000.
[42] S.A. Wilmarth, N.M. Amato, and P.F. Stiller, *MAPRM: A Probabilistic Roadmap Planner with Sampling on the Medial Axis of the Free Space*, IEEE International Conference on Robotics and Automation, PP. 1024-1031, 1999.
[43] R. J. Milgram, G.F. Liu, and J.C. Latombe, *On the Structure of the Inverse Kinematics Map of a Fragment of Protein Backbone*, Journal of Computational Chemistry, Vol. 29., No 1, PP. 50-68, 2008.
[44] A.A. Canutescu and R.L. Dunbrack Jr. *Cyclic coordinate descent: A robotics algorithm for protein loop closure*. Protein Science, 12:963-972, 2003.
[45] S. R. Lindemann and S. M. LaValle, *Incrementally reducing dispersion by increasing Voronoi bias in RRTs*. IEEE International Conference on Robotics and Automation, PP. 3251-3257, 2004.
[46] 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, pages 293-308. A K Peters, Wellesley, MA, 2001.
[47] B. Gipson, M. Moll and L. E. Kavraki, *Resolution Independent Density Estimation for motion planning in high-dimensional spaces*. IEEE International Conference on Robotics and Automation, PP. 2437-2443, 2013.
[48] O. B. Bayazit, Dawen Xie and N. M. Amato, *Iterative relaxation of constraints: a framework for improving automated motion planning*. IEEE/RSJ International Conference on Intelligent Robots and Systems, PP. 3433-3440, 2005.
[49] S. Dalibard and J.-P. Laumond, *Linear dimensionality reduction in random motion planning*, International Journal of Robotics Research, Vol. 30, No. 12, PP. 1461-1476, 2011.
[50] T. McMahon, S. Thomas, and N. M. Amato. *Sampling based motion planning with reachable volumes: Theoretical foundations*. IEEE International Conference on Robotics and Automation, PP 6514-6521, 2014.
[51] T. McMahon, S. Thomas, and N. M. Amato. *Sampling based motion planning with reachable volumes: Application to manipulators and closed chain systems*. IEEE/RSJ Internal Conference on Intelligent Robots and Systems, PP. 3705-3712, 2014.

TABLE III
PERFORMANCE COMPARISON BETWEEN DIFFERENT VARIANTS OF RRT ALGORITHMS

Phase	performance index	RRT-CONNECT	M-RRT-CONNECT	RRV	M-RRV
$1 \rightarrow 2$	iterations	2873.4	1538	1549.5	6404.8
	number of collision checking	4242.6	1289.7	6164.5	6555.7
	number of rejected samples	2633.0	1485.2	1414.7	6151.4
	number of accepted samples (regular/topological)	284/0	71.5/29.4	242.3/0	300.8/100.1
	computation time (s)	10.1	5.6	7.7	17.9
$2 \rightarrow 7$	iterations	831819.7	114718.5	118071.6	31324.0
	number of collision checking	922496.0	106338.5	185725.9	31613.8
	number of rejected samples	800164.1	112154.4	113623.3	30834.5
	number of accepted samples (regular/topological)	33750.8/0	2329.4/1726.0	4839.3/0	624.5/464.7
	computation time (s)	664.4	82.3	121.1	27.21
$7 \rightarrow 8$	iterations	1982225.0	1403327.6	361395.5	327837.3
	number of collision checking	2053421.9	1287168.0	1627906.4	326867.6
	number of rejected samples	1981294.4	1402607.3	361149.5	350371.1
	number of accepted samples (regular/topological)	69932.5/0	24350.0/716.0	12775.0/0	5692.0/166.3
	computation time (s)	781.8	627.7	804.8	178.45
$8 \rightarrow 9$	iterations	160.0	930.2	167.8	411.0
	number of collision checking	208.0	987.5	1023.4	511.4
	number of rejected samples	153.2	910.0	159.4	397.0
	number of accepted samples (regular/topological)	10.3/0	35.3/13.0	15.0/0	18.3/15.6
	computation time (s)	1.9	3.4	3.5	2.2
$9 \rightarrow goal$	iterations	103.5	16.2	29.0	287.1
	number of collision checking	124.0	16.0	157.2	201.5
	number of rejected samples	101.5	15.0	27.3	284.4
	number of accepted samples (regular/topological)	5.0/0	1.0/0	3.0/0	8.0/0
	computation time (s)	1.0	1.2	1.4	1.4

TABLE IV
PERFORMANCE OF M-RRV ALGORITHM UNDER DIFFERENCE PERCENTAGES P_t OF TOPOLOGICAL SAMPLES AND STEP SIZE $\epsilon_\Theta = 0.175$

Phase	performance index	$P_t = 20\%$	$P_t = 30\%$	$P_t = 50\%$	$P_t = 70\%$
$2 \rightarrow 7$	iterations	51107.0	197204.6	31324.0	41222.0
	number of collision checking	59711.6	212976.0	31613.8	36171.0
	number of rejected samples	49570.9	192268.8	30834.5	40813.5
	number of accepted samples (regular/topological)	1642.5/312.0	5544.4/1781.8	624.5/464.7	459.0/873.0
	computation time (s)	42.6	135.9	27.21	25.5
$7 \rightarrow 8$	iterations	1261628.7	88669.9	327837.3	1629177.7
	number of collision checking	1535638.4	93088.1	326867.6	1560695.3
	number of rejected samples	1501869.2	95073.0	350371.1	1752484.8
	number of accepted samples (regular/topological)	35771.0/241.8	2184.3/22.0	5692.0/166.3	17258.0/1136.5
	computation time (s)	812.4	52.5	178.45	836.2

TABLE V
PERFORMANCE OF M-RRV ALGORITHM UNDER DIFFERENT PERCENTAGES P_t OF TOPOLOGICAL SAMPLES AND STEP SIZE $\epsilon_\Theta = 0.35$

Phase	performance index	$P_t = 20\%$	$P_t = 30\%$	$P_t = 50\%$	$P_t = 70\%$
$2 \rightarrow 7$	iterations	95202.3	107296.6	81932.8	138118.6
	number of collision checking	100762.0	107751.4	76652.2	115902.0
	number of rejected samples	92323.8	104608.5	80277.0	137248.7
	number of accepted samples (regular/topological)	3117.7/574.1	3038.0/995.0	1672.7/1261.5	1623.2/2865.0
	computation time (s)	86.6	78.7	61.41	72.3
$7 \rightarrow 8$	iterations	1003793.9	1029721.5	920553.5	1032175.0
	number of collision checking	1223096.0	1059590.1	957782.0	993864.6
	number of rejected samples	1201492.4	1086748.9	1031699.1	1118121.7
	number of accepted samples (regular/topological)	28473.5/178.7	25637.7/302.3	15931.7/468.5	11009.0/744.2
	computation time (s)	706.5	505.1	462.7	535.0