A Survey on Topology in Motion Planning

Dongpeng Liu and Jiashuai Lu

December 8, 2020

Abstract

Motion Planning is a computational problem that, given an object and a set of obstacles, how can we move the object from a source point to a destination point without colliding with any obstacle. The problem has many robotic applications, such as autonomy, automation, artificial intelligence, etc. It is studied by both computational geometry community and robotics community. In this paper we discuss several applications of topology on motion planning. We study in detail the configuration space of motion planning problem, and some numerical homotopy invariant of any path-connected topological space X. We discuss how those invariant measure the "complexity of the problem of navigation in X" and determines the structure of motion planning algorithms in X. We introduce the useful properties of topological map for motion planning and how to build a "good" topological map. The interest of having multiple homotopically distinct paths into consideration also suggest us to study the approximate path similarity metrics.

1 Introduction

At every moment, we are making plans of our actions for the very next instant. Planning is involved in everywhere in our daily life, for example, which route we use to get to work, how to make a drone able to get back safely by itself. Motion planning, is a problem that given a start position s of a object X and a goal position t, possibly with a set of obstacles, computes a path that moves X from s to t without colliding with any obstacle. It is also know as the piano mover's problem. Imagine that we are given a computer-aided design (CAD) model of a house and a piano, and the goal is determine a way to move the piano from one room to another in the house without hitting anything [25]. This problem has many applications such as robotics, computational geometry, computer games, bioinformatics, virtual reality, and virtual prototyping [24]. Especially in the progress of autonomous robotics, it makes a critical role in the problem of enable the robotics to make decision for their own actions based on different cases [30]. These applications usually involve planning in planning under uncertainty [19], multi-agent systems [2], and dynamically changing environment [38].

While there are multiple versions of this problem according to how the obstacle information is given, researchers usually focus on the most simple version that all description of obstacles are given and fixed through the planning. The problem in this setting is often referred as the basic motion planning problem, and it is usually solved by first building a

graph to model the geometric structure of the environment and then finding a connected component that contains both the start and target positions. There are three common approaches to solve this basic setting problem: the roadmap approach, the cell decomposition approach, and the potential field approach [30]. The first two of them are both using the concept of *configuration space*, *free space* and the goal is finding a *free path* where the configuration space is a transformation from the realistic space of the robot of certain shape and size into a space that the robot is shrinked to a point.

We first put topology away for a second and go through some motion planning algorithms. Schwartz and Sharir showed how the cylindrical cell decomposition can solve any motion planning problem with freedom degree k, maximum degree of the n polynomial representation of the contact surface d in expected time $O(nd^{3^k})$ [31]. Using the roadmap technique, Canny gave an algorithm for any motion planning problem in $O(n^k \log nd^{O(k^4)})$ deterministic time and $O(n^k \log nd^{O(k^2)})$ expected time [3]. While the object is convex polygon and the degree of freedom is low, there exists some more efficient algorithms. For example, motion planning for a convex object B translating amidst m convex and pairwise disjoint obstacles can be performed in $O(m \log m)$ time [33].

Inspired by the theory of topological complexity of algorithms for solving polynomial equations developed by Smale [35] and Vassiliev [39], Farber introduced a system of topological approaches for robot motion planning algorithms in 2003. They proposed the topological complexity $\mathbf{TC}(X)$ of a topological space X. As a homotopy invariant, $\mathbf{TC}(X)$ measures of the difficulty to plan a continuous motion of a robot in X if one considers the topological space as the configuration space of the mechanical system in a motion planning problem. Their theory also introduced another notion of the topological complexity that analyzes the dependency of the instabilities of motion planning algorithms on homotopy properties of X. In the following years, applications of topological complexity and related theories to problems in robotics developed to an independent area of research. Farber *et al.* study the problem of moving a line in $\mathbb{R}^{n+1}$ by reducing it to a topological problem of calculating the topological complexity of the real projective space $\mathbf{TC}(\mathbb{R}\mathrm{P}^n)$ (recall the points of $\mathbb{R}\mathrm{P}^n$ represent lines through the origin of $\mathbb{R}^{n+1}$) as well the immersion problem in $\mathbb{R}\mathrm{P}^n$ [16]. In 2004, Farber and Yuzvinsky explored the topological complexity on designing collision free motion planning algorithms in Euclidean spaces. In the next year, Farber extended the result to the case of graphs [13]. and on graphs. Moreover, there are also some research on topological complexity but less relative to our topic, the topology in motion planning. Alexander Dranishnikov extend the theory to exploit the topological complexity of map in 2014 [10]. Later on, Pavešić applied it to measure the manipulation complexity of robotic devices in kinematic maps [27, 29] and showed how the topological complexities of covering projections approximate the topological complexity of the base space [28].

This first topic that we discuss (Section 2) is the topological complexity of the topological space, which is also the main topic in this paper. Before we talk about the actually homotopy invariants of topological spaces, we begin with briefly introducing several settings of the motion planning problem and give the general definition of configuration spaces under those settings. After that, we introduce four different priori notions of topological complexity and what topology properties they can measure. More specifically, for one of those topological complexities, we further introduce some lower bounds of it for various spaces.

A topological map is a graph with its vertices represent distinct positions and edges

represent a path between two distinct vertices. The concept is first proposed by Kuipers and Byung in 1991 [23] when they first tried to explore the topological approach to develop a robot exploration and mapping strategy by build a topological map to represent the environment. Kortenkamp and Weymouth (1994), Thrun and Bücken (1996) both studied practical strategies for producing topological maps [22, 36]. Ulrich and Nourbakhsh utilized the topological map to design a algorithm for topological localization [37]. On the other hand, it is usually hard to directly get a good topological map. Bosse *et al.* introduced the idea to divide the free space into small regions and combine them together into a high level topological map [1]. In Section 3 we discuss the property of topological map and define good topological map that is convenient for motion planning, and strategies of partitioning the configuration spaces into simple regions for motion planning [6].

Finally, many applications of motion planning could benefit from taking multiple homotopically distinct paths into consideration. But determining if two paths belongs to exact the same homotopy class could be very hard. In Section 4, we introduced some approximate metrics of measuring the homotopic similarity between two classes proposed by Denny *et al.* [9].

2 Topological Complexity of a Topological Space

In this section, we first introduce a formal definition of the motion planning problem. Then we will give the topological representation of the problem. After that, we are ready to discuss about a homotopy invariant of a path-connected topological space, which is proposed by Farber in 2003 [11].

2.1 Configuration Spaces of Motion Planning

The word *robot* was first invented by the Czech writer Karelǎpek in his play R.U.R. (Rossum's Universal Robots) [41]. But the concept of automata, can be found in many ancient mythologies. Autonomous robots, which should be able to complete many tasks without any human's control or assistance, is an essential objective of modern robotics research.

In the motion planning problem, we use *configuration* to describe the object's position and possible direction(all geometric characteristics of the object). The *configuration space* is often referred to all possible configurations(states). Typically, every configuration A consists of a finite set of real numbers, and we can see the configuration space X as a subset of the Euclidean space of which the dimension is the number of values we used to encode one state of the object. In this setting, every point in this subset represents a state of the object. Then we can naturally relate Topology and motion planning through the configuration space X. Given the configuration space X of a motion planning system, we may solve the problem by only looking at X's topology properties, possibly with some necessary geometric features.

Example 2.1 (Piano movers' problem [11, 31]). *We look at a simple version of the piano movers' problem here. As Figure 1 shows, the black piano shape can only move horizontally and the rectangles are obstacles. Let the four piano shapes be four different states of the piano. The problem is how to move the piano from one state to another one without colliding any obstacle.*
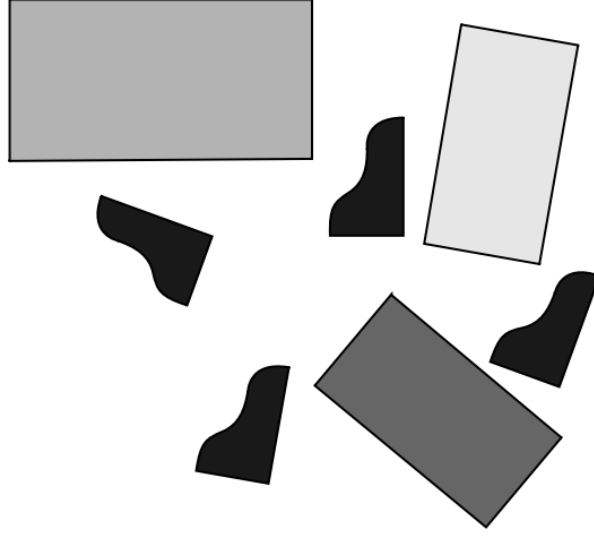
Figure 1. An example of piano mover's problem

In this case, every unique state of the piano can be described by three variables, two for the coordinates of the center of the piano and another one for denoting the direction. Therefore, the configuration space of it is a subset set of $\mathbb{R}^3$.

Schwartz and Sharir also give an algebraic formulation for the general version of "Piano" mover's problem. In this setting, we are considering collision-free states of some hinged object B, where B can be decomposed to a finite number of rigid compact subparts $B_1, B_2, \ldots,$. Every rigid subpart is bounded by a number of algebraic surfaces. Any two subparts B_1, B_2 can be connected together by any one of following ways:

1) B_1 and B_2 are partially overlapping such that B_1 and B_2 are combined by a unique point in the overlapping part.

2) B_1 and B_2 are connected by a hinge between two points a_1, a_2 where $a_1 \in B_1, a_2 \in B_2$. In this case, B_2 can revolve around some axis X fixed in the frame of B_1.

3) B_2 can slide and rotate along some axis X fixed in the frame of B_1.

To formulate the problem algebraically, they first describe a superspace of the set of all valid states of the hinged object B. The rotation group is a smooth 3-dimensional algebraic submanifold of the 9-dimensional Euclidean space of 3×3 real matrices. Recall that an algebraic manifold is a smooth algebraic variety. If B itself is minimal, i.e., B is a rigid compact subpart. Then its state can be represented by an Euclidean motion T from some base state to the given state. The transformation $Ts = Rs + s_0$ is defined by a pair (s_0, R) where s_0 is a point in $\mathbb{R}^3$ and R is a rotation matrix described above. So the transformation can be seen as a point in a smooth 6-dimensional algebraic submanifold of $\mathbb{R}^{12}$.

Otherwise, B consists of multiple parts through different ways listed earlier. One could find that in all cases, the overall states of all parts can be represented as a point in a smooth algebraic manifold in a Euclidean space of some dimension. For more details including a
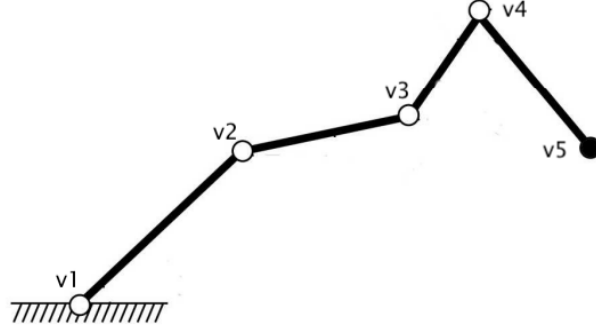
Figure 2. An example of the robot arm problem

general path-finding algorithm runs in time exponential in the number of degrees of freedom of the object, we refer to [31].

Example 2.2 (The robot arm [11, 24]). *In 1991, Latombe described a general robot arm problem. An arm is constituted by connecting a couple of bars through revolving joins as Figure 2 shows.*

Assume bars of the arm can have self-intersections. If the arm is in 2-dimensional plane, fix any v_i, the configuration space of v_{i+1} is S^1. Therefore, the whole configuration space X is:

$$\underbrace{S^1 \times S^1 \times \cdots \times S^1}_{\text{\# of bars}}$$

Similarly, in 3-dimensional space, the configuration space X is:

$$\underbrace{S^2 \times S^2 \times \cdots \times S^2}_{\text{\# of bars}}$$

2.1.1 General Configuration Spaces

Let A be a topological space and $Q_m = \{q_i | 1 \leq i \leq m, q_i \in A\}$ be a subset of points in A representing obstacles. Let $B = A - Q_m$ and let $X = F(A, m, n)$ denote some subset of $\underbrace{B \times B \times \cdots \times B}_{n}$ such that for every element $E = (e_1, e_2, \ldots, e_n)$ in X, every e_i of E has a unique value in B. Then $F(A, m, n)$ is the collision-free configuration space of a system of n points moving in space A. When G is a connected graph, the configuration spaces $F(G, m, n)$ are studied usually in robotics.

2.1.2 Topology of Configuration Spaces of Polygonal Linkages

In general, a linkage is a graph with fixed edge lengths. Figure 3 shows some linkages of various types. Let L be any graph with given edge lengths. A *configuration* of L is a geometric realization of L in Euclidean space. A *reconfiguration* is a continuous sequence of such
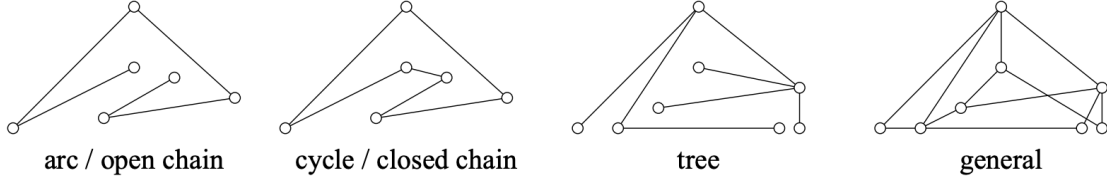
4

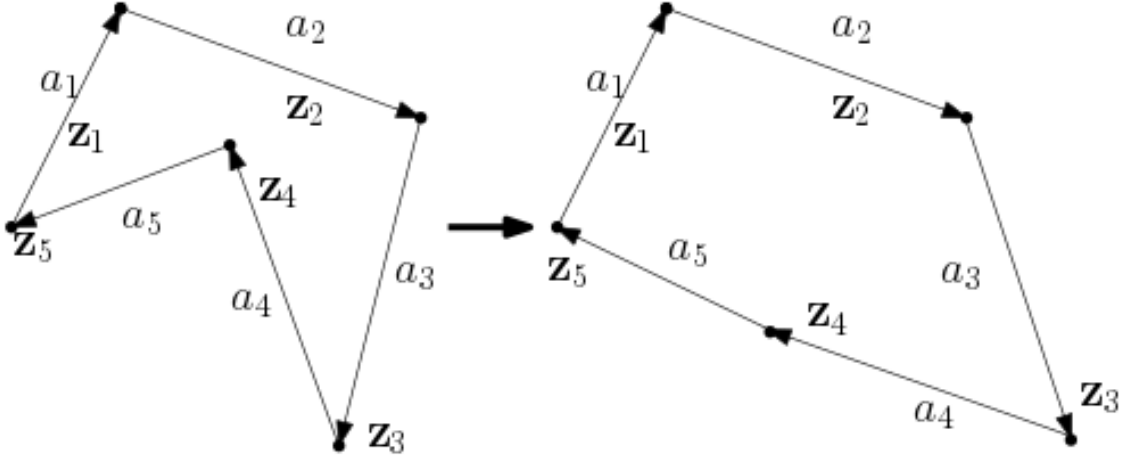**Figure 3.** Examples of different type linkages [7]

Figure 4. An example of reconfiguration of a linkage

configurations (see 4 for example). The configuration space X consists of all configurations and paths corresponding to reconfigurations [7].

There are a lot of results in studies on algebraic topological invariants of this configuration space. Here we study the topology of the configuration space of linkage of cycle, which are important manifolds representing shapes of closed polygonal chains.

Let $\mathbf{a}$ be any vector in $\mathbb{R}_+^m$, where $\mathbf{a} = (a_1, a_2, \ldots, a_m)$ consists of m positive real numbers. Then we define the variety $M(a)$ as:

$$M(\mathbf{a}) = \{(z_1, z_2, \ldots, z_m); z_i \in S^2, \sum_{i=1}^{m} a_i z_i = 0\}/SO^3$$

Here SO^3 performs as diagonal of the product $\underbrace{S^2 \times \cdots \times S^2}_{n}$, and $M(\mathbf{a})$ describes the variety of all closed polygonal shapes in 3-dimensional space where the side lengths are defined by $\mathbf{a}$. As an example, Figure 4 shows two different shapes with different coefficients $(z_1, z_2, \ldots, z_m)$ as the orientations of edges with same edge lengths.

Given the above definition of the variety of polygonal linkage, intuitively, we have the following lemma from the triangle inequality.

Lemma 2.3. *Let* $||\mathbf{a}|| = \sum_{i=1}^{m} a_i$. *$M(\mathbf{a})$ is nonempty if and only if $a_i < ||\mathbf{a}||$ for $1 \leq i \leq m$.*

We say a vector $\mathbf{a} \in \mathbb{R}_+^m$ is *generic* if $M(\mathbf{a})$ does not include an element $\mathbf{z} = (z_1, z_2, \ldots, z_m)$ such that $z_i = \pm 1$ for all i.

Lemma 2.4. *If* **a** *is generic, then $M(a)$ is a closed smooth manifold of dimension $2(m - 3)$ [11].*

To learn the relation between the variety of **a** and **a**, we need first define the *short* subset of $\{1, 2, \ldots, m\}$. We say a subset $V \subseteq \{1, 2, \ldots, m\}$ if such that $\sum_{i \in V} a_i \leq \sum_{j \notin V} a_j$. From Lemma 2.3 we know every subset V contains only a singleton is short subset. Let $S(\mathbf{a})$ denote all short subsets corresponding to **a**. Note that $S(\mathbf{a})$ is partially ordered set which is determined by its maximal elements.

Hausmann and Knutson proved the following lemma of a property of the variety and short subsets in 1998.

Lemma 2.5. *If two vectors $\mathbf{a}, \mathbf{a}' \in \mathbb{R}_+^m$ are generic, and there is an isomorphism between $S(\mathbf{a})$ and $S(\mathbf{a}')$, then $M(\mathbf{a}$ and $M(\mathbf{a}'$ are diffeomorphic.*

Recall that two smooth manifolds X, Y are diffeomorphic if there is a bijective and differentiable map $f : X \to Y$ and f^{-1} is differentiable.

There are also some universality theorems focusing on the topological properties of the configuration spaces of linkages.

Theorem 2.6 (Jordan and Steiner '98 [18]). *Any compact real algebraic variety $V \subset \mathbb{R}^n$ is homeomorphic to the union of components of the configuration space of a mechanical linkage.*

Theorem 2.7 (Kapovich and Millson '02 [20]). *For any smooth compact manifold X, there is a linkage L of which the moduli space is diffeomorphic to a disjoint union of a number of copies of X.*

Recall that the moduli space is the solutions of a geometric problem.

2.1.3 Perceptual Control Manifold of Configuration Spaces

Perceptual Control Manifold (PCM) [40] [34] is defined on the product of robot configuration space and sensor space. It offers a flexible way of generating motion plans to exploit sensor space effectively. PCM can be approximate by Topology representing network (TRN) [26] and then combine with diffusion-based path planning strategy. Recall 3-dimensional space is described as:

$$S_1^2 \times S_2^2 \times \cdots \times S_n^2$$

in vision-based control, the configuration is:

$$C_1^2 \times C_2^2 \times \cdots \times C_m^2$$

where C_i is related to a set of images from one or more video cameras. In Cartesian position of robot control, the image feature can also be defined as a function S_i which maps robot configurations to image feature values. $C_i : S_i \to C_i$. The PCM is a n-dimensional manifold defined on the product space $C \times S$, as it derived from n independent joint parameters over m non-independent parameters.

PCM is defined by varying all the joints S_i and represent the hypersurface in $C \times S$ space for higher degree of freedom. Given one robot configuration, it maps to exactly on point on the PCM. That ensure the uniqueness for motion planning. Obstacles and other constrains can also presented in PCM. With a prior knowledge of the robot dynamics and camera parameters, we can model the PCM and conduct motion planning on the space with strategies such as TRN. In TRN, initial and target configuration is mapping to PCM, then to find the best matching to update network and recursively eliminate obstacles and diffusion. By learning the PCM, the diffusion-based strategy is able to yield motion planning with obstacle avoidance.

2.2 Navigational Complexity of the Motion Planning Problem

Farber explored the topological properties of the robot motion planning problem and showed that there is a homotopy invariant quantity of configuration spaces X called the *navigational complexity $TC(X)$* [12, 14]. This invariant explains how knowing the cohomology algebra of configuration space of a robot one may predict instabilities of its behavior.

Let X denote the configuration space of a mechanical system S Then we can use a continuous path $\gamma : [0,1] \to X$ to represent a sequence of continuous motions of S where $\gamma(0), \gamma(1)$ are the initial and final states of the system, respectively.

If X is path-connected, it is easy to see that one may move the system between any two states. Let PX denote the space of all these continuous paths. Let $\pi : PX \to X \times X$ be the map from path to initial-final states configuration. Then π is a Serre fibration.

On the other hand, a motion planning algorithm is a function $s : X \times X \to PX$, and s is a section of π.

Now we are going to learn the continuity of motion planning algorithms. We say a motion planning algorithm s is continuous if solutions $s(A, B)$ for all $A, B \in X \times X$ continuously depend on initial-final states A and B.

Lemma 2.8 (Farber '03 [12]). *There exists a continuous motion planning algorithm for a configuration space X if and only if X is contractible.*

In practice, most of planning problems involves noncontractible configuration spaces. So we want to learn more about the *discontinuities*. Farber proposed an invariant $\mathbf{TC}(X)$ to measure the complexity the planning problem in space X. Besides this, they also described four different ways in which $\mathbf{TC}(X)$ affects the structure of motion planning algorithms for X. To describe those, we first use four distinct notions of navigational complexity of topological spaces: $\mathbf{TC}_j(X), j = 1, 2, 3, 4$.

A topological space X is called an *Euclidean Neighbourhood Retract* if it can be embedded into an Euclidean space $\mathbb{R}^k$ such that for some open neighbourhood $U, X \subset U \subset \mathbb{R}^k$, there exists a retraction $r : U \to X, r|_X = \mathbf{1}_X$ [14].

Definition 2.9. *A motion planning algorithm $s : X \times X \to PX$ is called* tame *if $X \times X = \bigcup_{1 \le i \le k} P_i$ for some finite number k such that*

1) $s|_{P_i} : P_i \to PX, 1 \le i \le k$ is continuous.

2) All P_i are disjoint with each other.

3) Every P_i is an ENR

In practice, all motion planning algorithms are tame and the function s restrict to any P_i is usually real algebraic and continuous.

Definition 2.10. *The topological complexity of a tame motion planning algorithm s is the minimal number of domains of continuity k in a representation of s in definition 2.9 [15].*

Definition 2.11. *The topological complexity $\mathbf{TC}_1(X)$ of a path-connected topological space X is the minimal topological complexity of motion planning algorithms of X [15].*

Observe that $\mathbf{TC}_1(X) = 1$ if and only if X is a contractible Euclidean Neighbourhood Retract. If there is no tame motion planning algorithm for X, $\mathbf{TC}_1(X) = \infty$.

Before we introduce the second notion of the topological complexity of topological spaces, we need first introduce the Schwarz genus. Let $\pi : X \to B$ be a fibration. Let k be a number such that there is an open cover of the base $B = \bigcup_{1 \leq i \leq k} S_i$ of which for each set S_i, there is a continuous section $s_i : U_i \to X$ of π. The Schwarz genus is defined as the minimum k satisfies this condition by Schwarz in 1958 [32].

Definition 2.12. *Let X be any path-connected topological space. The topological complexity $\mathbf{TC}_2(X)$ is the Schwarz genus of the fibration [15]*

$$\pi : PX \to X \times X$$

Let $\operatorname{cat}(X)$ denote the Lusternik-Schnirelmann category of X [8], then

$$\operatorname{cat}(X) \leq \mathbf{TC}_2(X) \leq \operatorname{cat}(X \times X)$$

The third notion of the topological complexity relates to the *order of instability* of motion planning algorithms introduced by *Farber* in 2004. For more details, see [14]. The order of instability is a fundamental feature of a motion planning algorithm. Formally, it is defined as the maximum value $r \leq k$ such that for any $\epsilon > 0$ we can find r pairs of initial-final configurations $(A_1, B_1), (A_2, B_2), \ldots, (A_r, B_r)$ in where the distance between any two pairs is less than ϵ and every pair belongs to a unique set P_i as defined in definition 2.9.

Definition 2.13. *The topological complexity $\mathbf{TC}_3(X)$ is the minimum order of instabilities of all tame motion planning algorithms in X [15].*

Before describing how $\mathbf{TC}_4(X)$ is defined, we first introduce the complexity of random motion planning algorithms [13].

Let X be any path-connected topological space and PX be the space of all continuous paths.

Definition 2.14. *For any $A, B \in X$, let $\gamma_1, \gamma_2, \ldots, \gamma_n \in PX$ be an order sequence of continuous paths such that $\gamma_i(0) = A, \gamma_i(1) = B$ for any i, and $p_1, p_2, \ldots, p_n \in [0, 1]$ is a sequence of probabilities summing to 1. Then a random n-valued path σ from A to B is defined as:*

$$\sigma = p_1 \gamma_1 + \cdots + p_n \gamma_n$$

Let $P_n X$ denote the set of all n-valued random paths in X. Let π_n be a map from $P_n X$ to the initial-final configuration space $X \times X$, then π_n is continuous [13]. Then an *n-valued random motion planning algorithm* can be defined as a continuous section of π_n, $s : X \times X \to P_n X$.

Definition 2.15. *The topological complexity $\mathbf{TC}_4(X)$ is defined as the minimum value n such that there exists an n-valued random motion planning algorithm for X [15].*

Till now we have listed $\mathbf{TC}_i(X), i = 1, 2, 3, 4$. Although these four different notions of topological complexity are not necessarily same, they have identical values when X is a simplicial polyhedron. Since $\mathbf{TC}_2(X)$ is the most handy invariant comparing to others, we set $\mathbf{TC} := \mathbf{TC}_2$. Now it is natural to look at the some lower bounds of the navigational complexity.

Let $\dim(X)$ denote the covering dimension of X.

Theorem 2.16. *If X is a path-connected paracompact locally contractible topological space, then $\mathbf{TC}(X) \leq \mathrm{cat}(X \times X) \leq 2\dim(X) + 1$.*

Recall that a topological space is paracompact if every open cover of it has an locally finite open refinement.

Theorem 2.17 (Farber '04 [14]). *For any r-connected CW-complex,*

$$\mathbf{TC}(X) < \frac{2 \cdot \dim(X) + 1}{r + 1} + 1$$

2.3 Topological Complexity of $F(\mathbb{R}^m, n)$

Recall $F(\mathbb{R}^m, n)$ represents the configuration space of a system with n distinct points in $\mathbb{R}^m$. Let $P\mathbb{R}^m$ denote the set of all continuous paths in $\mathbb{R}^m$ such that for every path γ, for every element $e \in \gamma$, e has unique coordinate at each dimension in $\mathbb{R}^m$. A valid motion planning algorithm for $F(\mathbb{R}^m, n)$ is a map from $\mathbb{R}^m \times \mathbb{R}^m \to P\mathbb{R}^m$.

Farber and Yuzvinsky discovered the following theorem for $\mathbf{TC}(F(\mathbb{R}^m, n))$ in 2004.

Theorem 2.18.
$$\mathbf{TC}(F(\mathbb{R}^m, n)) = \begin{cases} 2n - 1 & m \text{ is odd,} \\ 2n - 2 & m = 2. \end{cases}$$

Moreover, they mentioned that for the cases of m is even and greater than 2, $\mathbf{TC}(F(\mathbb{R}^m, n))$ is either $2n - 1$ or $2n - 2$. But they conjectured that this value is $2n - 2$. For the proof of the theorem, see [17].

In this section, we have discussed about the configuration spaces of motion planning problem under several different settings such as the piano mover's problem, the robot arm problem, and the general configuration spaces for polygonal linkages. After that, we introduced and analyzed some topological complexity of the configuration spaces. There are some algorithms designed with these properties exist [15].
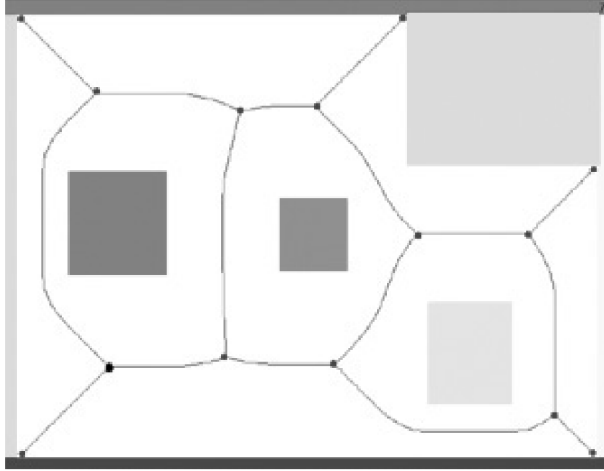
Figure 5. An example of good topological map, the rectangles represent the obstacles

3 Topological Maps and Decomposition

Many motion planning algorithms exploit topology in the free space to help design efficient motion planners or produce concise maps. Most of those works have utilized the generalized Voronoi diagram (GVD), which is a topological map that can be embedded into the free space. A *topological map* usually is used to describe a graph in where the nodes correspond to distinct position and edges represent paths between different positions in robotics. In 2003, Choset and Rizzi explored that a topological map is not only a graph but carries more information and may actually be used to perform task decompositions [6].

3.1 Good Topological Maps

The topological map M for a free space X is a map that every homotopy class of X has a corresponding homotopy class in M. Moreover, if the fundamental group of M is isomorphic to the fundamental group of the free space, we say M is a *good topological map.* For example, for 2-dimensional problems the generalized Voronoi diagram is a good map. See Figure 5. Analogously, a topological map contains redundant loops which do not have corresponding loops in the free space is *bad*.

Some topological map is the abstraction of a 1-dimensional roadmap, which is geometric structures embedded in the free space with three properties: accessibility (it is possible to plan a path between two points onto the roadmap), connectivity (move towards the vicinity of the target point along the roadmap), and departability (reach the target). To see why the roadmap needs to be only 1-dimensional, consider a point-GVD G in $\mathbb{R}^3$, G consists of 2-dimensional manifolds. The overlap of these 2-dimensional sheets is a 1-dimensional structure called the generalized Voronoi graph (GVG) that the distance from any point in it to adjacent three obstacles are same [5]. Although the GVG satisfies accessibility and departability, it is not guaranteed to be connected in general. In 1995, Choset and Burdick introduced an extension of GVG called the hierarchical generalized Voronoi graph (HGVG), which is also retract-like. But the later is connected under certain conditions in [5]. Therefore, it could be used to produce the roadmap for performing a motion planning algorithm. For the detail

of how to construct such a HGVG incrementally, see [4].

However, generally speaking, the roadmaps mentioned above hardly be good topological maps, and there is no 1-dimensional retract for space of dimensions greater than 2. In other words, it is hard to generate good topological maps for $\mathbb{R}^m$ with $m > 2$.

3.2 Task Decomposition

According to the definition of good topological map above, restracts work as good maps because the retraction naturally offers the accessibility, departability, and the retract is connected in every connected components of the free space by the continuity of the retraction function. Moreover, it preserves the cardinality of the first fundamental group as well.

As pointed earlier, we cannot define a 1-dimensional retract of a space with dimension at least 3. Nevertheless, a topological map of a 2-dimensional work space can be used to decompose a multi-dimensional configuration space into a set of contractible regions. Therefore, we can find good maps through retraction in each region and put them back together through the topological adjacency relationship from the original work space. The result is called *piecewise retract* by Choset and Rizzi. Notice that although the piecewise retract may be a bad topological map most times, it consists of smaller good maps for subsets of the configuration space and thus works well for a motion planner. For examples of applications of piecewise retract, see [6].

4 Homotopic Path Similarity in Motion Planning

As mentioned in the introduction, many applications of robotic motion planning benefit from considering multiple homotopically distinct paths rather than a single one. For example when the planning involves multi-agent system or under uncertainty. However, it could be difficult to efficiently check if two paths belong to different homotopy classes. Denny *et al.* proposed metrics to approximate the homotopic similarity between two paths and showed how to compute this approximate similarity efficiently [9].

4.1 Probabilistic Path Similarity Metric

Instead of determine if two paths are in the identical homotopy class, we want to have a metric approximate how likely two paths are in the same homotopy class.

Definition 4.1. *A path $\pi_m = \{q_1, q_2, \ldots, q_m\}$ is a contiguous sequence of configurations, two configurations q_i, q_{i+1} are adjacent if $\delta(q_i, q_{i+1}) \leq r$ where δ is a distance function and r is specified in the problem.*

Definition 4.2. *Let π_m be a path defined above. The t-resolution path τ of π_m is a contiguous sequence of configurations discretized at t_{res} steps, where t_{res} is a resolution for a parametric representation $t \in [0, 1]$ of π_m, where $t = 0$ is q_1 and $t = 1$ is q_m. Let τ^k denote the k_{th} configuration of τ, $k \in [0, 1]$.*

```
LOCAL PLANNER SUCCESS SIMILARITY (τ₁, τ₂, Δ):
    ⟪τ₁, τ₂ are t-resolution paths, Δ is a local planner.⟫
    s ← 0, n ← 0
    for t ∈ [0, 1] at t-resolution increments do
        if Δ(τ₁ᵗ, τ₂ᵗ) then
            s ← s + 1
        n ← n + 1
    return s/n
```

Figure 6. Algorithm for computing Local Planner Success Similarity

Let A be the set of all t-resolution. Let $\mu : A \times A \to [0, 1]$ be a path similarity metric. If μ is of high-quality, then for any two paths $\alpha, \beta \in A$, $\mu(\alpha, \beta) = 0$ means α and β are in different homotopy classes where $\mu(\alpha, \beta) = 1$ means their homotopy classes are identical. Moreover, for all $0 < \mu(\alpha, \beta) < 1$, the value represents the probability that α and β are with the same homotopy class.

4.2 Local Planner Success Similarity

Denny *et al.* introduced a naive similarity metric based a common subroutine of sampling-based motion planners [21], called the Local Planner Success Similarity (LPSS), see Figure 6. The LPSS essentially computes a proportion of successful local planning attempts to total attempts between configurations of the two paths with the same t value. $\mu = 1$ in this metric guarantee the two paths are in the same homotopy class. The value of μ close to 1 implies the homeomorphism between two paths are likely to exist. Conversely, If the value of μ is near 0, they are unlikely in the same class. This algorithm would not yield 0 value since the start and goal configurations of τ_1, τ_2 are the same. This method is not a good approximation of the similarity metric since it only tries one possible transformation. However, if it correctly determines two paths are in the same class, it has a byproduct that tell us how one can deform into another.

4.3 Topology-based Edit Distance Similarity

Based on topological reasoning, Denny *et al.* suggested another similarity metric called Topology-Based Edit Distance Similarity (TBEDS) that compares the portions of the work space each path accesses. As shown in Figure 7, the algorithm converts each path into a string and uses edit distance as the metric. In this algorithm, the method for converting a path into the topology-based string assumes the work space of the robot has been tetrahedralized and converted into a Reeb Graph. For details of this method, see [9].

If a value of one is returned in this metric, then the paths taken by the point of interest of the robot are in the same homotopy classes of the work space. However, this does not mean the homotopy classes of τ_1, τ_2 are same. It still depends on which interest point the algorithm choose and the exact problem. On the other hand, the value close to 0 does imply two paths are in different homotopy classes.

> TOPOLOGY-BASED EDIT DISTANCE SIMILARITY(τ_1, τ_2, Δ):
> $\langle\!\langle \tau_1, \tau_2$ *are t-resolution paths, Δ is a local planner.*$\rangle\!\rangle$
> $s_1 \leftarrow \text{TOPOLOGYBASEDSTRING}(\tau_1)$
> $s_2 \leftarrow \text{TOPOLOGYBASEDSTRING}(\tau_2)$
> $d \leftarrow \text{LEVENSHTEINDISTANCE}(s_1, s_2)$
> return $1 - \frac{d}{\max(|s_1|, |s_2|)}$

Figure 7. Algorithm for computing Topology-based Edit Distance Similarity

Applications of extracting homotopically distinct paths from a roadmap through those metric exist in [9].

5   Discussion

In this survey, we reviewed recent progress on several applications of topology in robot motion planning, namely, the topological complexity of the configuration space in motion planning, topological maps for building motion planners or producing concise route, approximate homotopic path similarity in motion planning. We then went through the basic setting of the problem, introduced configuration spaces of some classic mechanical system, and explored the properties of four topological complexities of the topological space. There are some open problems regarding the homotopy invariant $\mathbf{TC}(X)$ such as the rational version of $\mathbf{TC}(X)$, symmetric motion planning, as well as motion planning in aspherical spaces, see survey [15].

Choset and Rizzi applied their task decomposition strategy to two classes of practical problems to build "good" topological maps. In the future, we can expect more problems can be solved with task decomposition and some more general techniques be designed in order to exploit this tool on motion planning. On the other hand, the practical interest of considering multiple homotopically distinct paths in certain problems motivate us to explore more on finding efficient way to compute a good approximation of the homotopic path similarity and developing algorithms that produce homotopically distinct paths through these metrics.

References

[1] Michael Bosse, Paul Newman, John Leonard, Martin Soika, Wendelin Feiten, and Seth Teller. An atlas framework for scalable mapping. In *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, volume 2, pages 1899–1906. IEEE, 2003.

[2] Frederic Bourgault, Alexei A Makarenko, Stefan B Williams, Ben Grocholsky, and Hugh F Durrant-Whyte. Information based adaptive robotic exploration. In *IEEE/RSJ international conference on intelligent robots and systems*, volume 1, pages 540–545. IEEE, 2002.

[3] John Canny. *The complexity of robot motion planning.* MIT press, 1988.

[4] Howie Choset and Joel Burdick. Sensor based planning. ii. incremental construction of the generalized voronoi graph. In *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, volume 2, pages 1643–1648. IEEE, 1995.

[5] Howie Choset and Joel Burdick. Sensor based motion planning: The hierarchical generalized voronoi graph. *Algorithms for Robot Motion and Manipulation*, pages 47–61, 1996.

[6] Howie Choset and Alfred A. Rizzi. Topology in motion planning. In Paolo Dario and Raja Chatila, editors, *Robotics Research, The Eleventh International Symposium, ISRR, October 19-22, 2003, Siena, Italy*, volume 15 of *Springer Tracts in Advanced Robotics*, pages 90–99. Springer, 2003.

[7] Robert Connelly and Erik D Demaine. Geometry and topology of polygonal linkages. *Handbook of discrete and computational geometry*, pages 197–218, 2004.

[8] Octavian Cornea, Gregory Lupton, John Oprea, Daniel Tanré, et al. *Lusternik-Schnirelmann category.* Number 103. American Mathematical Soc., 2003.

[9] Jory Denny, Kaiwen Chen, and Hanglin Zhou. A topology-based path similarity metric and its application to sampling-based motion planning. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6498–6505. IEEE, 2018.

[10] Alexander Dranishnikov. On topological complexity of twisted products. *Topology and its Applications*, 179:74–80, 2015.

[11] Michael Farber. Topological complexity of motion planning. *Discret. Comput. Geom.*, 29(2):211–221, 2003.

[12] Michael Farber. Topological complexity of motion planning. *Discrete and Computational Geometry*, 29(2):211–221, 2003.

[13] Michael Farber. Collision free motion planning on graphs. In *Algorithmic Foundations of Robotics VI*, pages 123–138. Springer, 2004.

[14] Michael Farber. Instabilities of robot motion. *Topology and its Applications*, 140(2-3):245–266, 2004.

[15] Michael Farber. Topology of robot motion planning. In *Morse theoretic methods in nonlinear analysis and in symplectic topology*, pages 185–230. Springer, 2006.

[16] Michael Farber, Serge Tabachnikov, and Sergey Yuzvinsky. Topological robotics: motion planning in projective spaces. *arXiv preprint math/0210018*, 2002.

[17] Michael Farber and Sergey Yuzvinsky. Topological robotics: subspace arrangements and collision free motion planning. *Translations of the American Mathematical Society-Series 2*, 212:145–156, 2004.

[18] Denis Jordan and Marcel Steiner. Configuration spaces of mechanical linkages. *Discrete & Computational Geometry*, 22(2):297–315, 1999.

[19] Leslie Pack Kaelbling and Tomás Lozano-Pérez. Integrated task and motion planning in belief space. *The International Journal of Robotics Research*, 32(9-10):1194–1227, 2013.

[20] Michael Kapovich and John J Millson. Universality theorems for configuration spaces of planar linkages. *Topology*, 41(6):1051–1107, 2002.

[21] Lydia E Kavraki, Petr Svestka, J-C Latombe, and Mark H Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE transactions on Robotics and Automation*, 12(4):566–580, 1996.

[22] David Kortenkamp and Terry Weymouth. Topological mapping for mobile robots using a combination of sonar and vision sensing. In *AAAI*, volume 94, pages 979–984, 1994.

[23] Benjamin Kuipers and Yung-Tai Byun. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Robotics and autonomous systems*, 8(1-2):47–63, 1991.

[24] Jean-Claude Latombe. *Robot motion planning*, volume 124. Springer Science & Business Media, 2012.

[25] Steven M LaValle. *Planning algorithms*. Cambridge university press, 2006.

[26] Thomas Martinetz and Klaus Schulten. Topology representing networks. *Neural Networks*, 7(3):507–522, 1994.

[27] Petar Pavešić. Complexity of the forward kinematic map. *Mechanism and Machine Theory*, 117:230–243, 2017.

[28] Petar Pavešić. Topological complexity of a map. *arXiv preprint arXiv:1809.09021*, 2018.

[29] Petar Pavesic. A topologist's view of kinematic maps and manipulation complexity'. *Contemp. Math*, 702:61–83, 2018.

[30] Eric Roberts. Motion planning in robotics. 1998.

[31] Jacob T Schwartz and Micha Sharir. On the "piano movers" problem. ii. general techniques for computing topological properties of real algebraic manifolds. *Advances in applied Mathematics*, 4(3):298–351, 1983.

[32] Albert S Schwarz. The genus of a fibre space. *Trudy Moskovskogo Matematicheskogo Obshchestva*, 11:99–126, 1962.

[33] Micha Sharir. Algorithmic motion planning. 1997.

[34] Rajeev Sharma and Herry Sutanto. A framework for robot motion planning with sensor constraints. *IEEE Transactions on Robotics and Automation*, 13(1):61–73, 1997.

[35] Steve Smale. On the topology of algorithms, i. *Collected Papers Of Stephen Smale, The (In 3 Volumes)-Volume 3*, 3:1254, 2000.

[36] Sebastian Thrun and Arno Bücken. Learning maps for indoor mobile robot navigation. Technical report, CARNEGIE-MELLON UNIV PITTSBURGH PA DEPT OF COMPUTER SCIENCE, 1996.

[37] Iwan Ulrich and Illah Nourbakhsh. Appearance-based place recognition for topological localization. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, volume 2, pages 1023–1029. Ieee, 2000.

[38] Jur P Van Den Berg, Dennis Nieuwenhuisen, Léonard Jaillet, and Mark H Overmars. Creating robust roadmaps for motion planning in changing environments. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1053–1059. IEEE, 2005.

[39] VA Vassiliev. Cohomology of braid groups and complexity of algorithms. *Funktsional. Anal. i Prilozhen*, 22:15–24, 1988.

[40] Michael Zeller, Rajeev Sharma, and Klaus Schulten. Motion planning of a pneumatic robot using a neural network. *IEEE Control Systems Magazine*, 17(3):89–98, 1997.

[41] Dominik Zunt. Who did actually invent the word 'robot'and what does it mean? *The Karel Čapek website*, 2002.