A New Application of Discrete Morse Theory to Optimizing Safe Motion Planning Paths

Aakriti Upadhyay[‡]*, Boris Goldfarb[⋄], Weifu Wang[†], and Chinwe Ekenna[‡]

[1] [‡] are with the Department of Computer Science and [⋄] is with the Department of Mathematics and Statistics, University at Albany, SUNY.
{aupadhyay,cekenna,bgoldfarb}@albany.edu
[2] [†] is with Baidu Research; wangweifu@baidu.com

Abstract. We present a novel method for extracting geometric and topological features from a robot's configuration space. To accomplish this, we define a discrete Morse function on the Vietoris-Rips simplicial complex to identify critical points on the surface of obstacles present. These critical points serve as waypoints for determining feasible bounds near an identified obstacle. This work builds on previous work that provides a method to approximate the number of samples required to generate pathways. Our results achieve near-optimal paths with a low computation time and reduced path distance in this work. We conduct experiments in different environments and with various robots, including the Kuka YouBot and PR2 robots in simulation, and demonstrate the performance gains compared to state-of-the-art methods.

Keywords: Discrete Morse theory, Geometric features, Vietoris-Rips complexes.

1 Introduction

Motion planning which involves identifying feasible paths for a robot from a start to a goal configuration is proven to be a PSPACE-hard problem. One of the widely used motion planning methods is a sampling-based motion planning approach [8], which attempts to *approximate* the configuration space ($\mathcal{C}_{space}$) connectivity information with geometric properties embedded in the roadmap. The roadmap generated by these planners does not describe the configuration space information but only a subspace of it. Because the relationship of this subspace to the configuration space is largely unknown, the extracted information of the underlying space becomes of little or no use for future analysis. Thus, it requires investigation of new research directions to overcome these drawbacks.

Interestingly, the concurrent development of topological data analysis (TDA) has yielded some encouraging results [5], [2]. In our preliminary work [30], we used the Vietoris-Rips complex to obtain a homotopy-equivalent abstraction of the configuration space's topology. In this paper, we present a new framework for extracting the geometric information of the *cspace* using discrete Morse theory. The framework built on a pre-processing step that extracts free space ($\mathcal{C}_{free}$) connectivity information using the Vietoris-Rips (VR) complex. It uses simplicial collapse to remove redundant edges and vertices while preserving necessary topological properties developed in [30] and stores them in a roadmap. Then, to

extract the geometric features of the configuration space's obstacles, we create a specific function evaluated on the sample points and extend it to a discrete Morse function on the VR complex that identifies critical points. We use these critical points to locate samples near the obstacles ($\mathcal{C}_{obst}$). The resulting sample set provides a small and efficient roadmap representation of samples surrounding the $\mathcal{C}_{obst}$. Finally, we combine this information with the extracted topological information to generate a space-specific sample roadmap that is memory-efficient. We use this roadmap to successfully construct a path from the start to the robot's goal position with near-optimal properties.

The scientific contributions of our work to the configuration space include:

1. A near-optimal algorithm for planning good quality paths from the extracted geometric representation.
2. An application of density-based discrete Morse function to the extracted topological representation.
3. A re-usable roadmap with preserved topological and geometric information.

2 Related Work

Sampling-Based Motion Planning (SBMP) methods: Sampling-based methods have been categorized into two main classes: graph-based methods such as the Probabilistic Roadmap Method (PRM) [17] and tree-based methods such as Rapidly-exploring Random Tree (RRT) [21]. Uniform sampling method [17] generates nodes uniformly at random in $\mathcal{C}_{space}$ retaining valid ones but remains inefficient in the presence of narrow passages. Obstacle-Based PRM (OBPRM) [33] samples configurations near $\mathcal{C}_{obst}$ surfaces either by pushing configurations to the $\mathcal{C}_{obst}$ boundary or by finding surface intersections of randomly placed line segments. While OBPRM excels in narrow passages, it can be expensive due to the multiple collision detection checks. Gaussian [6] and Bridge-Test [15] filter samples with inexpensive tests to find samples near $\mathcal{C}_{obst}$ boundaries or directly in narrow passages, respectively. However, both methods perform the same basic sampling as uniform random sampling and suffer from requiring many samples to find a valid path in a narrow passage. Additionally, both methods suffer from parameter tuning, which can greatly affect the performance and quality of the mappings produced. We use these auxiliary samplers to generate the topology maps for our approach except OBPRM because it continuously failed to generate a dense map to meet our stopping criteria from [30].

Topological Approaches to Motion Planning: Finding a safe and optimal path in the presence of obstacles has been an important research topic in robot motion planning. Many research has employed the advantage of mathematical tools to improve the quality of paths by sampling-based planners. Work in [19] and [34] combined the methods from computational geometry with SBMP to find paths closer to the boundaries of the obstacles or at a medial-axis distance. More interestingly, a rising application of Morse theory or its related tool, i.e., Reeb graph, in works [26], [9], [25] and [24] showed promising results in finding safer paths around the obstacles under integral/differential optimization. However, their algorithms perform computationally-intensive calculations when dealing with high-dimensional robots or complex environments. Our approach differs as we apply discrete Morse theory, which covers all the trappings of Morse theory in discrete form. Thus, making our algorithm more efficient in complex scenarios.

Coverage Path Planning: Coverage path planning determines a route that guarantees an agent will pass over every point in a given environment. Research in [23], [4], [2] and [1] proposed algorithms that applied brute-force technique or an exact cellular decomposition to guarantee coverage in an unknown space for different inspection purposes. The work in [3] used Unmanned Aerial Vehicle (UAV) for coverage tasks and applied adaptive viewpoint sampling to construct accurate 3D models of large complex structures. Surveys in [12] and [7] discussed several coverage approaches that produced optimal coverage paths with minimum length and low execution time resulting in low energy consumption for UAVs and considered full or partial information about the area of interest. In this work, we provide a roadmap which covers all sub-regions of the $\mathcal{C}_{free}$, i.e., covers the entire $\mathcal{C}_{free}$ with configurations at a safer distance from $\mathcal{C}_{obst}$.

3 Foundations

The section describes the theoretical foundation of our algorithm and defines the relevant mathematical concepts.

Space approximation using the Vietoris-Rips complex: In our previous work [30], we applied two important mathematical concepts to perform a memory-efficient path planning in a given $\mathcal{C}_{space}$ on generating a homotopy-equivalent topological map of the $\mathcal{C}_{free}$.

Definition 1. *(Abstract Simplicial complex) An abstract simplicial complex K is a collection of subsets of a given set X closed under the subset operation. It is a generalization of a graph representing higher-than-pairwise connectivity relationships.*

The elements of the set X are called vertices of K, and the subsets are called the simplices of K.

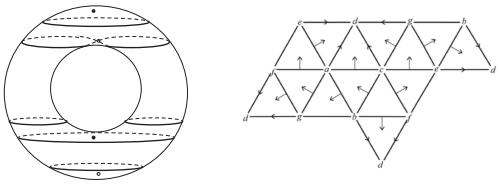
Definition 2. *(Vietoris-Rips complex) Given a set X of points in a Euclidean space E, the Vietoris-Rips complex $R_\varepsilon(X)$ is the abstract simplicial complex whose k-simplices are the subsets of $k+1$ points in X of diameter at most ε.*

In this paper, we apply discrete Morse theory to the same simplicial complex to identify critical points on the boundary of $\mathcal{C}_{obst}$.

Critical simplex in discrete Morse theory: Discrete Morse theory, originally defined by Forman [11], is a discrete analog of the classical smooth Morse theory. It is used in applications like ours to simplify the topological information about the space by decreasing the representation size without affecting the crucial properties such as the homotopy type. The context is either a simplicial or more general cellular complex.

Consider the topologically non-trivial shape called the 2D torus as an obstacle model in the free space of 3D Euclidean space (Fig. 1(a)). It is impossible to embed the torus in the plane, but we can draw its triangulation by indicating parts of the boundary glued together when the labels match.

Once a discrete Morse function is given, the theory provides the construction of a consistent flow indicated with arrows (see Fig. 1(b)). There are several critical cells of various dimensions here, but only one critical point d. As a result, we see that the discrete flow identifies the critical simplices. When a critical simplex is 0-simplex, we refer to it as a critical point.



(a) (b)

Fig. 1: The figure shows the 2D torus model in (a) and an identified critical point d in the planar triangulation of torus in (b).

Constructing a discrete Morse Function in $\mathcal{C}_{space}$: We adopt a discrete Morse theoretic view to enumerate the critical points of $\mathcal{C}_{obst}$. We define a density-based discrete Morse function on the constructed simplicial complex to extract the geometric property.

Definition 3. *Let $D(x)$ denote the distance between the point $x \in \mathcal{C}_{free}$ and the nearest point y on the closest obstacle $O_i \in \mathcal{C}_{obst}$, that is, $D(x) = \min_{y \in O_i} \|x - y\|$.*

Definition 4. *Let $\Gamma(y, \varrho)$ be a density function where $\varrho > 0$ and y is the point on the obstacle surface. Our choice of the function Γ counts all neighbors in $R(X) \subseteq \mathcal{C}_{free}$ close to y within distance ϱ.*

Our function is in fact defined at any point in $\mathcal{C}_{space}$ and is given by

$$f(x) = D(x) \cdot \Gamma(y, \varrho). \tag{1}$$

Theorem 1. *The restriction of f to the vertices of the Vietoris-Rips complex is the restriction of a discrete Morse function defined on all of the complex $R(\mathcal{C}_{space})$. The critical points of this function identify features of the obstacles.*

Proof. We start first arguing the second statement, then show how to obtain the Morse function. Let $\omega \geq 0$. We consider any closest obstacle O from the robot and the subset $X = D^{-1}([0, \omega])$. We denote $\text{Hull}(X, \omega)$ as the convex hull of the set X at scale ω from O, where t and s are the random vertices/points that define the elements in X. In other words,

$$\text{Hull}(X, \omega) = \bigcup_{t,s \in X} [t, s]. \tag{2}$$

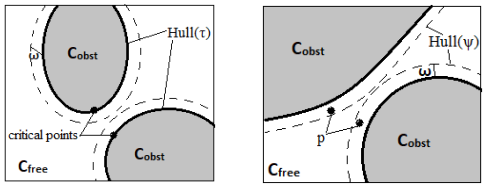
Taking the obstacle O, the local maxima and minima of the function f occurs on the surface of the obstacle O. They will be critical points of f in $\tau \subset O$ (as objects are assumed to be polyhedral) when $f \to 0$. The $\text{Hull}(\tau)$ determines the boundary of the obstacle surface containing these critical points. Let us take point $p \in X$. When $D(p) = 0$, that is $\omega \to 0$, the distance between point p and closest obstacle O becomes negligible. The density of neighboring points in $\mathcal{C}_{free}$ decreases on approaching closer to the obstacle O. However, the value of $\Gamma(y, \varrho)$ has a very meaningful limit. It is easy to see that the features on the surface of the polyhedron O can be detected as critical points of f on the $\text{Hull}(X, \omega)$. Thus, we determine them as critical points which satisfy the following conditions,

1. $\forall$ points $p \in X$; $\exists$ point $y \in O$ such that $f(p) \leq f(y)$,
2. $\forall$ points $y \in O$; $\exists$ point $p \in X$ such that $f(y) \geq f(p)$.

The identified critical point p will satisfy the equation

$$\lim_{\omega \to 0} f(p) = D(p) \cdot \Gamma(y, \omega). \tag{3}$$

Figure 2 shows an example of identified critical point on the obstacle surface.



(a) (b)

Fig. 2: The figure shows the identified critical points on the boundaries of the $\mathcal{C}_{obst}$ in (a) and the feasible critical points (as denoted by p) in $\mathcal{C}_{free}$ closer to the critical points at distance ω in (b).

To apply discrete Morse theory to identify the critical points of f restricted to $\text{Hull}(X, \omega)$, we represent f given on the vertex set of the Vietoris-Rips complex as the restriction of a discrete Morse function f defined on all of the complexes. It is a well-studied problem in discrete Morse theory with many solutions. A very well-known work [18] contains an algorithm called EXTRACT (see Algorithm 1 in loc.cit.). A more efficient version can be found in a recent improvement and extension [14] (see Algorithm 2 called EXTRACTRIGHTCHILD).

4 Methodology

To extract the geometric information, we perform two associated steps. First, we abstract the topological representation of the $\mathcal{C}_{space}$ using the VR complex. Second, we apply the discrete Morse function to this representation. Here, we describe implementation details to identify critical points and feasible critical points information for our topology map.

4.1 Feasible Critical Points in $\mathcal{C}_{space}$

Let $\Phi = D^{-1}([0, \varrho])$ be the compact set of points in $\mathcal{C}_{free}$ at most distance ϱ from the obstacle boundary such that the computed ϱ value can be given as

$$\varrho = \frac{1}{n} \sum D(s); \forall s \in \text{Hull}(R(X)), \tag{4}$$

where C denotes the set of identified critical points, $R(X)$ denotes simplicial complex vertices set, n is the cardinality of set C and $D(s) = \min_{y \in C} \|s - y\|$ from Def.3. The term clearance defines the offset distance of point p from the obstacle boundary, thus shifting p away from $\mathcal{C}_{obst}$ to $\mathcal{C}_{free}$. We say p becomes a feasible critical point with a maximum ϱ-clearance from the obstacle boundary if $p \in \mathcal{C}_{free}$. An example is shown in Figure 2(b).

For comparison, the Generalized Voronoi Diagram (GVD) provides a roadmap to extract high-clearance paths. The GVD defines the maximum clearance for a pathway from the $\mathcal{C}_{obst}$, as utilized in the Medial-Axis PRM [32]. An exact computation of the medial-axis distance is not practical for problems involving many DOFs (degrees of freedom) and a cluttered environment with many obstacles, as this requires an expensive and intricate calculation of the $\mathcal{C}_{obst}$. We have proven in previous work that on reaching a needed sampling condition, the Vietoris-Rips complex provides a topologically equivalent map of the space as the $\check{C}$ech complex. After the simplicial collapse, the resulting simplicial complex is a sparse sub-sampled graph that reconstructs the surfaces equivalent to the Delaunay complex, similar to explained in [10]. Instead of computing the medial-axis distance from $\mathcal{C}_{obst}$ to the boundary of the Voronoi cell, we considered the closest distance from each critical point to the convex hull of the simplicial complex and took the average of all these distance values. In this work, we used the computed mean value as the clearance distance from the $\mathcal{C}_{obst}$, i.e., ϱ.

4.2 Generating ϱ-clearance samples in the $\mathcal{C}_{space}$

Algorithm 1 provides a roadmap with configurations at a distance ϱ from the obstacle in the $\mathcal{C}_{free}$, i.e., feasible critical points, by applying the discrete Morse function f on the constructed VR complex S. The algorithm considers the convex hull of invalid nodes set as the $\mathcal{C}_{obst}$ boundary, returned by function *GetObject-ConvexHull()*, and computes the Morse values of these nodes to identify the potential critical points on the $\mathcal{C}_{obst}$ surfaces, from equation (1). When the collapses in VR complex S get close to the $\mathcal{C}_{obst}$ boundary, the distance becomes an equalizer making density the most important contributing factor. Thus, the local minima and maxima of f depend on the lowest and highest density value achieved by f at a point $y \in \mathcal{C}_{obst}$, in line 10. The algorithm inspects computed Morse values (in line 11) to identify the critical points for each $\mathcal{C}_{obst}$ in line 12. It captures the set of feasible critical points in $S \subseteq \mathcal{C}_{free}$ at clearance ϱ from the identified critical points, in lines 14-15. The greater the value of ϱ decides the maximum clearance of a configuration from a $\mathcal{C}_{obst}$. Finally, the algorithm outputs a new graph G_{fcp} having configurations in the $\mathcal{C}_{free}$ at ϱ-clearance from the $\mathcal{C}_{obst}$ and the set of identified critical points of $\mathcal{C}_{obst}$ in C.

5 Experimental Setup

We perform experiments in simulation as a proof of concept of our methodology. The experiments are executed on a Dell Optiplex 7040 desktop machine running OpenSUSE operating system, and the code is developed in C++ language. We used the brute force K-closest neighbor finding technique [22], the euclidean distance metric, and a straight line local planner for sampling and connection stages. We used the RAPID [13] collision detection method during the sampling, connection, and query stages. The simulation experiments were performed in 5 different environments with robots ranging from 2 DOF to 14 DOF, as shown in Figures 3 and 4. To ensure the accurate verification of our sampling condition, we integrated the computation of the reachable boundary volume of the robot as its environment boundary to meet the sampling condition. The computation time taken for identifying critical and feasible critical points for these robots is negligible, as empirically observed in section 6.

Algorithm 1 ϱ-clearance algorithm

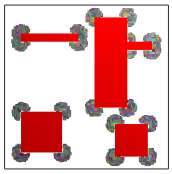
Input: G: complete sampled graph from [30]; S: simplicial complex; O: Obstacle set, D: distance function, Γ: density function, f: Morse values set function, ϱ: value from eq.(4), C: set of identified critical points, N: set of configurations around identified critical points, G_{fcp}: a graph of feasible critical points.

C $\leftarrow$ *null*, N $\leftarrow$ *null*;
G = *GenerateGraph()*; ◁ Refer [30]
O = *GetObjectConvexHull(G)*;
S = *CollapseComplex(G)*; ◁ Refer [30]
if S is not empty **then**
  for each obstacle $o_i \in O$ **do**
    for all sample $x \in S$ **do**
      $D(x) = min_{y \in o_i} ||x - y||$ ◁ Refer Def. 3
    for all node $y \in o_i$ **do**
      $\Gamma(y, \varrho) = \bigcup_{||x-y|| < \varrho} x \in G$ ◁ Refer Def. 4
      $f(y) = D(x) \cdot \Gamma(y, \varrho)$
      if $f'(y) \to 0$ **then**
        C = C $\sqcup$ y
  for all y $\in$ C **do**
    N = N $\sqcup$ $\Gamma(y, \varrho)$
for each neighbor n $\in$ N **do**
  $G_{fcp} \leftarrow$ N[n]
return $\{G_{fcp}, C\}$

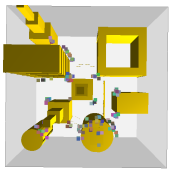
1. **2D environment**: The 2D environment consists of a point robot with random obstacles placed in the space, as shown in Figure 3a (referenced [16]).
2. **Parking Garage**: The 3D environment has a vehicle parking garage structure, as shown in Figure 4b. The robot is a planar car with 3 DOF.
3. **Urban environment**: This environment consists of buildings as obstacles in the city-like structure, as shown in Figure 3b. The robot is a 6 DOF drone.
4. **Kuka YouBot environment**: The environment consists of a tree as an obstacle and a fixed base Kuka YouBot in it, as shown in Figure 3c. This robot is a simulation replica of Kuka YouBot [20] with an extended long arm (10 DOF).
5. **PR2 robot environment**: The environment consists of two pillar blocks placed on the table as obstacles where the robot is required to pass through the blocks to grasp the stick kept on the other side, as shown in Figure 4a. The robot is a simulation replica of the PR2 robot [31] with only the right-hand arm (14 DOF) and has a fixed base.

6 Results

In this section, we discuss the results obtained using different PRM sampling strategies, i.e., Uniform [17], Gaussian [6] and Bridge-Test [15] planners, and RRT methods in 3D environments. We compared the results with two topology baseline RRT methods, Dynamic Domain RRT [34] and Dynamic Region-based RRT [9]. We also show the performance of the RRT, RRT*, PRM, and PRM*



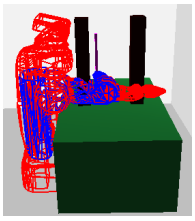
(a) 2D environment (square)



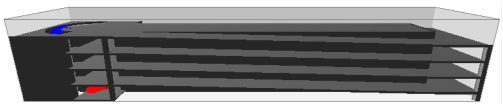
(b) Urban environment (drone)

(c) Kuka YouBot environment

Fig. 3: The figure shows feasible critical points around the identified critical points of the $\mathcal{C}_{obst}$ for three environments. In (b), a point size view of the drone is shown for better visualization of feasible critical points. In (c), we see the feasible critical points of the robot are around the branches and bark of the tree.



(a) PR2 robot environment



(b) Parking Garage (car)

Fig. 4: Environments Studied

methods in the 2D environment (used by Karaman et al., in [16]), and all result values were averaged over 10 experiment runs in each case, i.e., evaluated total 500 trials for the 3D environments and 240 trials for the 2D environment. We used bi-directional RRT methods in all environments.

6.1 Topology Map

Table 1 shows the statistical result values computed for each sampler in each environment over ten runs. The column "Total Nodes" represents the generated density map nodes obtained on satisfying the sampling condition from [30]. In "% Reduction," we report the percentage of nodes removed after a simplicial collapse, and in "% FCP," we report the percentage of feasible critical points present in the topology map. The topology map contains both topological and geometric information about the $\mathcal{C}_{space}$, and its nodes are listed in the column "Extracted Nodes." We highlight the sampling strategy for each environment that preserved the topological and geometric information in their map with low memory overhead and improved the performance of existing RRT and PRM methods. Bridge-Test performed best in the narrow passage regions of the Parking Garage and PR2 robot environments, while Uniform(PRM) performed best in the open space region of the Kuka YouBot environment and Gaussian performed best in the cluttered region of the Urban environment. Our approach, like our previous machine learning method [27], revealed the core functionality of these sampling strategies, resulting in improved performance. So, the extracted properties of $\mathcal{C}_{space}$ aid in guiding the robot for memory-efficient path planning. Figure 3 shows the captured feasible critical points for three of our environments.

Table 1: STATISTICS OF EXTRACTED TOPOLOGICAL AND GEOMETRIC INFORMATION

Environment	Sampler	Total Nodes	% Reduction	% FCP	Extracted Nodes
Parking Garage	Uniform	30000	40	20	18084
	Gaussian	30000	32	15	20616
	Bridge-Test	**25000**	**44**	**12**	**14124**
Urban environment	Uniform	10000	46	9	5455
	Gaussian	**10000**	**41**	**12**	**5872**
	Bridge-Test	5000	34	20	3287
Kuka YouBot environment	Uniform	**15000**	**26**	**46**	**11040**
	Gaussian	20000	27	47	14593
	Bridge-Test	10000	28	42	7156
PR2 robot environment	Uniform	20000	28	56	14466
	Gaussian	5000	36	19	3222
	Bridge-Test	**2000**	**53**	**1**	**940**

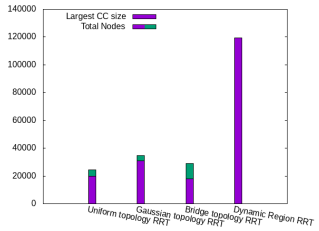
The survey in [12] concluded that when combined with topology methods, SBMP methods provide promising results for optimal coverage path planning. In this paper, we show how our approach combines the topological, i.e., the homotopy equivalent map of the $\mathcal{C}_{free}$ space, and the geometric, i.e., the critical points and configurations near the $\mathcal{C}_{obst}$, properties into a topology map. This topology map contains enough nodes to cover all sub-regions of the $\mathcal{C}_{free}$ in the $\mathcal{C}_{space}$. Our method has usefulness in determining a path that can pass through all nodes in a given area or volume of interest while avoiding obstacles in coverage path planning problems. This paper examines the quality of paths generated by a topology map (a $\mathcal{C}_{space}$ coverage map). Using our topology map, we demonstrate the convergence of the RRT and PRM methods to the near-optimal solution, i.e., produce paths within feasible bound from $\mathcal{C}_{obst}$ using memory-efficient roadmap.

6.2 Comparison to RRT-based algorithms

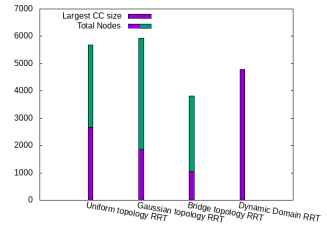
As an initial roadmap, we input the topology map generated by the Uniform, Gaussian, and Bridge-Test sampling strategies into the RRT method. Dynamic Domain RRT was unable to find a path for the Parking Garage environment, while Dynamic Region-based RRT was unable to find a pathway for the Urban and Kuka YouBot environments. Both methods failed to complete in the PR2 robot environment. Our topology methods, on the other hand, finished planning paths in all four environments.

Nodes and largest clique size: Figure 5 depicts the total number of nodes in the roadmap as well as the size of the largest connected component (CC). The number of nodes in the largest CC does not equal the total number of nodes in the topology map, indicating that the number of nodes required to connect the start and goal configuration is less than the total number of nodes in the topology map. Dynamic Region-based RRT and Dynamic Domain RRT acquired all nodes from their roadmap to establish a path. Thus, our methods require fewer nodes than baseline methods to produce a pathway to the destination.

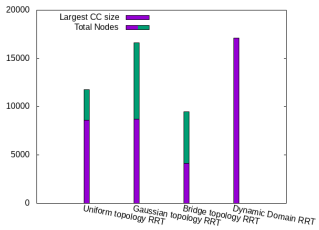
Query time and collision calls: Figure 6 depicts the total time required to solve a query as well as the number of collision check calls made. The query time accounts for topology map generation time, connection time, and path planning time. The collision calls count the number of times nodes' or edges' are validated, and we examined their impact on the topology map's path planning time. Compared to the baseline methods, our planners used fewer collision



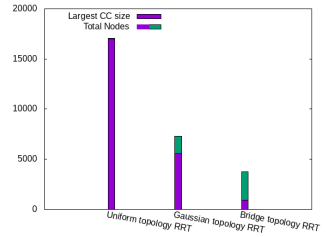
(a) Parking Garage (car)



(b) Urban environment (drone)



(c) Kuka YouBot environment



(d) PR2 robot environment

Fig. 5: The figure shows the total number of nodes present in the roadmap in the purple-green bar and the size of the largest connected component (CC) in the purple bar for all environments. The size of the largest CC indicates the number of nodes connected to establish a path from start to goal position by RRT planners.

checks, resulting in less query time in all three environments. In the PR2 robot environment, BridgeTopologyRRT outperformed other methods.

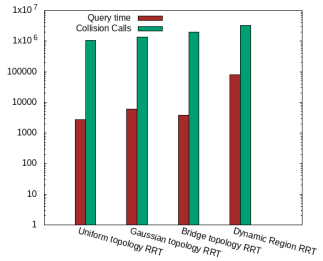
Path Quality: Table 2 depicts an improvement in path quality after employing topology maps. We discovered that BridgeTopologyRRT produced shorter paths in Parking Garage and PR2 robot environments as explained in section 6.1. Similar observed for UniformTopologyRRT in the Kuka YouBot environment and GaussianTopologyRRT in the Urban environment. As a result, we can conclude that our topology methods can produce shorter paths than baseline methods, with less computation time and fewer nodes.

Table 2: Path cost achieved by different RRT planners

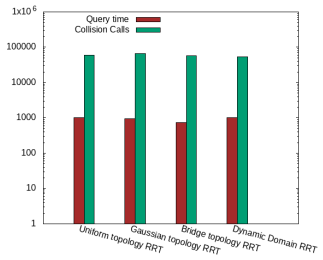
Methods	Parking Garage	Urban	Kuka YouBot	PR2 robot
Uniform Topology RRT	311.10	173.08	**14.03**	2.09
Gaussian Topology RRT	367.78	**164.37**	16.61	2.27
Bridge Topology RRT	**288.3**	184.5	21.56	**1.86**
Dynamic Domain RRT	N/A	167.15	23.39	N/A
Dynamic Region-based RRT	364.7	N/A	N/A	N/A

6.3 Comparison to PRM-based algorithms

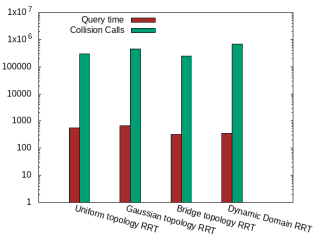
As initial roadmaps, we input the PRM method with two different $\mathcal{C}_{space}$ coverage maps: 1. a density map - nodes generated to provide complete coverage of $\mathcal{C}_{space}$ and 2. a topology map - nodes preserved from the density map after the collapse and geometric feature extraction. We compare the performance of



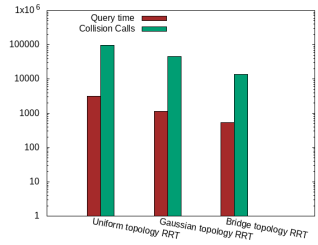
(a) Parking Garage (car)



(b) Urban environment (drone)



(c) Kuka YouBot environment



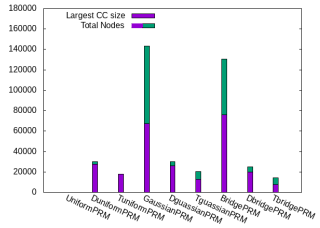
(d) PR2 robot environment

Fig. 6: The figure shows the total query time and the number of collision calls invoked for all RRT methods.

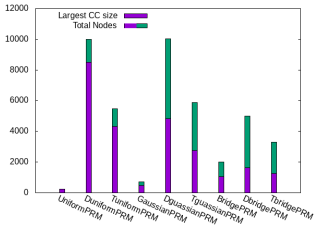
PRM with density map-based PRM and topology map-based PRM for all three samplers in all four environments. In the plot labels, we indicated density map methods with the prefix 'D' and topology map methods with the prefix 'T.'

Nodes and largest clique size: In Figure 7, we noticed that our topology map captured nodes in regions isolated from open free space, i.e., a space enclosed by $\mathcal{C}_{obst}$. Given this, we can conclude that, regardless of the sampler method used, our approach aided in capturing nodes close to $\mathcal{C}_{obst}$, covering enclosed sub-regions of $\mathcal{C}_{free}$, in the same way, that an obstacle-based sampler captures nodes close to $\mathcal{C}_{obst}$. Overall, we conclude that our topology map-based methods required fewer nodes to plan paths in Parking Garage and PR2 environments than other methods. In the Kuka YouBot and PR2 robot environments, however, all nodes were connected to form one largest CC for all planners, as shown in Figures 7c and 7d. Because the absence of enclosed regions in these environments makes the planners use all nodes from their roadmap for the query analysis. After 400 hours of continuous running, the uniform sampling strategy failed to find a path in the Parking Garage environment.

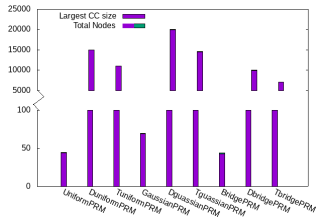
Query time and collision calls: Figure 8 shows that PRM took less time to generate a path than density map-based PRM and topology map-based PRM in Urban and Kuka YouBot environments for all three samplers. This difference is due to the additional pre-processing time needed for topology or density map generation. Topology map-based methods achieved fewer collision calls with the shortest query time in complex environments such as Parking Garage (maze-like structure) and PR2 robot environment (high DOF). As a result, we can conclude that the computation time overhead for generating a topology map becomes negligible for our approach when dealing with complex environments.



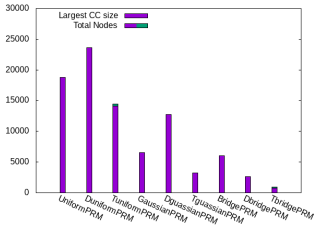
(a) Parking Garage (car)



(b) Urban environment (drone)

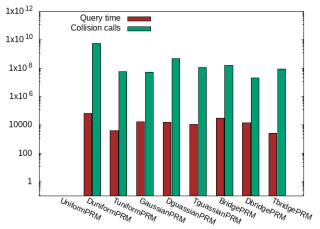


(c) Kuka YouBot environment

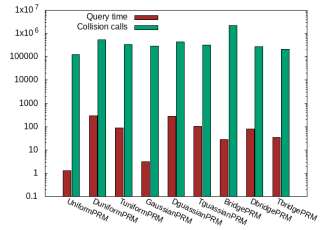


(d) PR2 robot environment

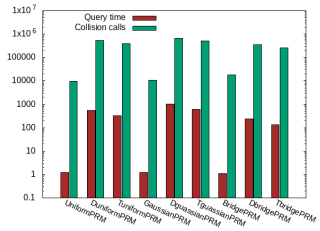
Fig. 7: The figure shows the total number of nodes present in the roadmap and the size of the largest connected component (CC) for all environments. The x-axis labels with the prefix 'D' indicate density map methods, and the prefix 'T' denotes the topology map methods for PRM planners.



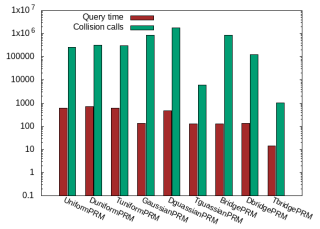
(a) Parking Garage (car)



(b) Urban environment (drone)



(c) Kuka YouBot environment



(d) PR2 robot environment

Fig. 8: The figure shows the total query time and the number of collision calls invoked for all PRM methods.

Path Quality: As shown in Table 3, topology map-based PRM methods produced shorter paths than other methods. It also demonstrates that the preserved important nodes from the density map after the simplicial collapse and

the feasible critical points information in the topology map were crucial in bringing edges closer to $\mathcal{C}_{obst}$. As a result, the topology map's path cost converges to a near-optimal value much faster.

Table 3: Path cost achieved by different PRM planners

PRM Planners	Parking Garage	Urban	Kuka YouBot	PR2 robot
Uniform	DNF	2552.5	105.2	10
Uniform with density map	12027	2495.33	92	12
Uniform with topology map	11450.6	2420.1	**85**	4
Gaussian	14676.7	2493.4	104.6	12
Gaussian with density map	11668.75	2278.78	89	15
Gaussian with topology map	11540	**2139.8**	**85**	5
Bridge-Test	14252	2773.4	116.1	11
Bridge-Test with density map	11812	2377.33	92	16
Bridge-Test with topology map	**11307.5**	2377.4	89	**3**

6.4 Comparison with Existing Work in a 2D Environment

In this environment, we compare the performance of the different planners, i.e., RRT, RRT*, PRM, and PRM*, using our pre-processed maps with the results from [16] to show the improvement in path quality as the sampling density increases. The methods like RRT* and PRM* are proved to converge to an optimal solution as the number of samples increases. We record the behavior of these methods for our topology map to understand the difference in the result from [16]. We performed experiments for sampling densities of 500, 1000, 2500, 5000, 10000, and 15000. We did not sample beyond 15000 nodes because the Hausdorff distance does not change after reaching the lowest constant value, the reason explained in [30]. We analyzed the behavior using the uniform sampling strategy density map and topology map.

Path Cost and Time: In Figure 9a, we observe that as the number of samples increases, the path cost decreases and reaches a minimum cost value for the density map. On the other hand, the methods were able to show a similar pattern with a decrease in path cost using our topology map, and, as the sampling condition gets fulfilled at the last sampling density, the path cost attains a minimum value at an earlier stage. In Figure 9c, we observe that the methods take less time to make connections between the nodes and fewer collision calls using our topology map than the density map. We observe a similar trend for PRM and PRM* methods in Figures 9b and 9d. Thus, the methods showed an early convergence to the optimal solution using our topology map. We conclude that via our topology map, we achieve paths with near-optimal properties at a faster convergence rate compared to results from [16].

7 Discussion

Discrete Morse theory is a recent development in topology that has resulted in an explosion of applications in a wide range of fields. This paper is an example of such a robotics application. Discrete Morse theory should not be confused with discretized smooth Morse theory because it is a genuinely combinatorial subject and thus amenable to efficient computer implementations. As defined, it is a theory that has all of the trappings of smooth Morse theory and, so, can be used in place of the smooth theory. The benefits are enormous. It is purely

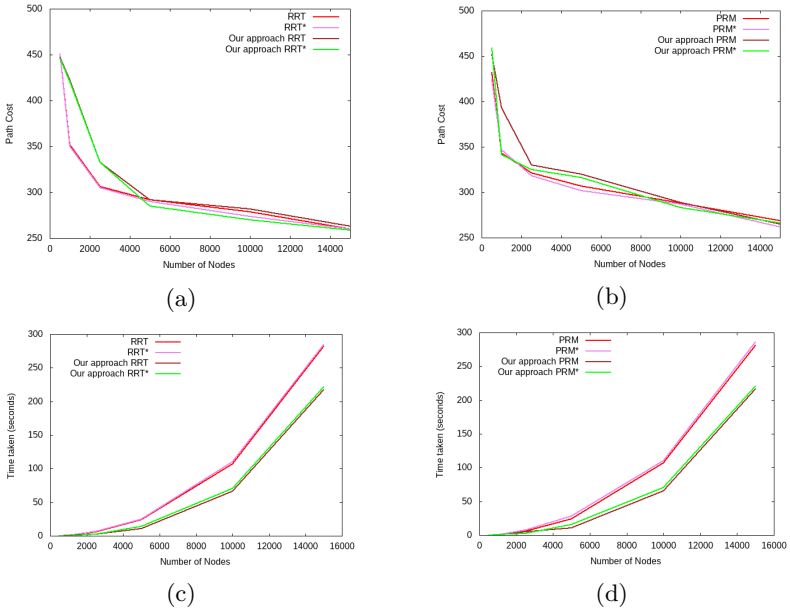


Fig. 9: Plots showing performance of RRT, RRT*, PRM, and PRM* in the 2D environment compared with our approach for Path cost vs. Number of Nodes in (a) and (b), and Time taken vs. Number of Nodes in (c) and (d).

a discrete theory, that is, applied directly to simplicial or more general cellular complexes. The ability to generate discrete functions from samples, such as the density-based function, which is unavailable in a smooth context, is one of the many additional applications that can be particularly useful for our work. We demonstrated two important practical applications of the critical point information retrieved using our density-based discrete Morse function in diverse path planning [28] and computational biology [29]. The former solved a more compelling problem of finding diverse paths in high dimensional space with different homotopy classes. The generated roadmap's coarsely-diverse pathways were defined using critical point information. This roadmap aided in overcoming the computational overhead of recalculating a new path if the previous route failed due to the unexpected placement of a new obstacle. Thus, showing promising applications in self-driving car initiatives. The latter provided an understanding of protein-ligand interactions, which is critical in the study of drug design with potential life-saving biological implications, such as assisting in the development of therapeutic drugs, vaccines, and point-of-care technologies.

8 Conclusion

The work presented an algorithm that implements discrete Morse theory to identify critical points on the boundaries of $\mathcal{C}_{obst}$ using the Vietoris-Rips simplicial complex. We created a topology map of the $\mathcal{C}_{space}$ that approximates $\mathcal{C}_{free}$ with configurations near $\mathcal{C}_{obst}$, i.e., captures topological and geometric information.

Using our topology map, we found that methods such as PRM and RRT improve planning time and path cost. In a 2D environment, the performance of RRT, RRT*, PRM, and PRM* methods demonstrated convergence to near-optimal paths in less time. The future work will transition the pre-processing step of generating the topology map of the entire $\mathcal{C}_{space}$ to incremental path planning steps. The incremental planner will generate local topology maps of sub-space while simultaneously planning local paths to connect to a global pathway. We also intend to put our technique to the test in a dynamic and uncertain environment. Although several research works have investigated the use of Morse theory in path planning problems, little attention is devoted to testing its application in a dynamic environment incorporating uncertainty in path planning methods. Thus, this remains an open question for future research.

References

1. Acar, E.U., Choset, H.: Sensor-based coverage of unknown environments: Incremental construction of morse decompositions. The International Journal of Robotics Research **21**(4), 345–366 (2002)
2. Acar, E.U., Choset, H., Rizzi, A.A., Atkar, P.N., Hull, D.: Morse decompositions for coverage tasks. The international journal of robotics research **21**(4), 331–344 (2002)
3. Almadhoun, R., Taha, T., Gan, D., Dias, J., Zweiri, Y., Seneviratne, L.: Coverage path planning with adaptive viewpoint sampling to construct 3d models of complex structures for the purpose of inspection. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 7047–7054. IEEE (2018)
4. Atkar, P.N., Choset, H., Rizzi, A.A., Acar, E.U.: Exact cellular decomposition of closed orientable surfaces embedded in/spl rfr//sup 3. In: Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164), vol. 1, pp. 699–704. IEEE (2001)
5. Attali, D., Lieutier, A., Salinas, D.: Vietoris–rips complexes also provide topologically correct reconstructions of sampled shapes. Computational Geometry **46**(4), 448–465 (2013)
6. Boor, V., Overmars, M.H., van der Stappen, A.F.: The Gaussian sampling strategy for probabilistic roadmap planners. In: Proc. IEEE Int. Conf. Robot. Autom. (ICRA), vol. 2, pp. 1018–1023 (1999)
7. Cabreira, T.M., Brisolara, L.B., Ferreira Jr, P.R.: Survey on coverage path planning with unmanned aerial vehicles. Drones **3**(1), 4 (2019)
8. Choset, H., Lynch, K.M., Hutchinson, S., Kantor, G.A., Burgard, W., Kavraki, L.E., Thrun, S.: Principles of Robot Motion: Theory, Algorithms, and Implementations. MIT Press, Cambridge, MA (2005)
9. Denny, J., Sandström, R., Bregger, A., Amato, N.M.: Dynamic region-biased rapidly-exploring random trees. In: Twelfth International Workshop on the Algorithmic Foundations of Robotics (WAFR) (2016)
10. Dey, T.K., Fan, F., Wang, Y.: Graph induced complex on point data. In: Proceedings of the twenty-ninth annual symposium on Computational geometry, pp. 107–116 (2013)
11. Forman, R.: A user's guide to discrete morse theory. Séminaire Lotharingien de Combinatoire [electronic only] **48**, B48c–35 (2002)
12. Galceran, E., Carreras, M.: A survey on coverage path planning for robotics. Robotics and Autonomous systems **61**(12), 1258–1276 (2013)
13. Gottschalk, S., Lin, M., Manocha, D.: Obb-tree: A hierarchical structure for rapid interference detection. In: Proc. ACM SIGGRAPH, pp. 171–180 (1996)

14. Holmgren, B., McCoy, B., Fasy, B., Millman, D.: If you must choose among your children, pick the right one. arXiv preprint arXiv:2103.13882 (2021)
15. Hsu, D., Jiang, T., Reif, J., Sun, Z.: Bridge test for sampling narrow passages with probabilistic roadmap planners. In: Proc. IEEE Int. Conf. Robot. Autom. (ICRA), pp. 4420–4426. IEEE (2003)
16. Karaman, S., Frazzoli, E.: Incremental sampling-based algorithms for optimal motion planning. In: Proceedings of Robotics: Science and Systems. Zaragoza, Spain (2010)
17. Kavraki, L.E., Švestka, P., Latombe, J.C., Overmars, M.H.: Probabilistic roadmaps for path planning in high-dimensional configuration spaces. IEEE Trans. Robot. Automat. **12**(4), 566–580 (1996)
18. King, H., Knudson, K., Mramor, N.: Generating discrete morse functions from point data. Experimental Mathematics **14**(4), 435–444 (2005)
19. Krishnaswamy, V., Newman, W.S.: Online motion planning using critical point graphs in two-dimensional configuration space. In: Proceedings 1992 IEEE International Conference on Robotics and Automation, pp. 2334–2339. IEEE (1992)
20. KUKA Robotics Corporation: Kukayoubot. `https://cyberbotics.com/doc/guide/youbot`. Accessed: May 27, 2022
21. LaValle, S.M., Kuffner, J.J.: Randomized kinodynamic planning. In: Proc. IEEE Int. Conf. Robot. Autom. (ICRA), pp. 473–479 (1999)
22. McMahon, T., Jacobs, S., Boyd, B., Tapia, L., Amato, N.M.: Evaluation of the k-closest neighbor selection strategy for prm construction. Technical Report TR12-002, Texas A&M, College Station Tx. (2011)
23. Oksanen, T., Visala, A.: Coverage path planning algorithms for agricultural field machines. Journal of field robotics **26**(8), 651–668 (2009)
24. Orthey, A., Akbar, S., Toussaint, M.: Multilevel motion planning: A fiber bundle formulation. The international journal of robotics research (2020)
25. Orthey, A., Frész, B., Toussaint, M.: Motion planning explorer: Visualizing local minima using a local-minima tree. IEEE Robotics and Automation Letters **5**(2), 346–353 (2019)
26. Shvalb, N., Moshe, B.B., Medina, O.: A real-time motion planning algorithm for a hyper-redundant set of mechanisms. Robotica **31**(8), 1327–1335 (2013)
27. Upadhyay, A., Ekenna, C.: Investigating heterogeneous planning spaces. In: Simulation, Modeling, and Programming for Autonomous Robots (SIMPAR), 2018 IEEE International Conference on, pp. 108–115. IEEE (2018)
28. Upadhyay, A., Goldfarb, B., Ekenna, C.: A topological approach to finding coarsely diverse paths. In: IROS: IEEE/RSJ International Workshop on Intelligent Robots and Systems. IEEE (2021)
29. Upadhyay, A., Tran, T., Ekenna, C.: A topology approach towards modeling activities and properties on a biomolecular surface. In: BIBM: IEEE International Conference on Bioinformatics and Biomedicine. IEEE (2021)
30. Upadhyay, A., Wang, W., Ekenna, C.: Approximating cfree space topology by constructing vietoris-rips complex. In: IROS: IEEE/RSJ International Workshop on Intelligent Robots and Systems, pp. 2517–2523 (2019)
31. WillowGarage: PR2 Robot. www.willowgarage.com (2016)
32. Wilmarth, S.A., Amato, N.M., Stiller, P.F.: MAPRM: A probabilistic roadmap planner with sampling on the medial axis of the free space. In: Proc. IEEE Int. Conf. Robot. Autom. (ICRA), vol. 2, pp. 1024–1031 (1999)
33. Wu, Y.: An obstacle-based probabilistic roadmap method for path planning. Master's thesis, Department of Computer Science, Texas A&M University (1996)
34. Yershova, A., Jaillet, L., Simeon, T., Lavalle, S.M.: Dynamic-domain RRTs: Efficient exploration by controlling the sampling domain. In: Proc. IEEE Int. Conf. Robot. Autom. (ICRA), pp. 3856–3861 (2005)