Topology-Driven Recovery Path Planning in Dynamic Obstacle Environments

Mukulika Ghosh[1], Aakriti Upadhyay[2], and Chinwe Ekenna[3]

[1] Missouri State University, MO, USA.mghosh@missouristate.edu
[2] Colorado School of Mines, CO, USA. aakriti.upadhyay@mines.edu
[3] University at Albany NY, USA. cekenna@albany.edu

Abstract. This paper presents a novel topology-driven method for planning recovery paths amidst dynamic obstacles. Our framework integrates Discrete Morse Theory principles from prior research and expands to provide congestion-aware solutions for mobile robot and manipulator alike. To enhance efficiency and reliability, our algorithm balances local segment deflection and topologically diverse paths, aiming to reduce dynamic obstacle interference and optimize path length. We introduce a new approach to path deflection that efficiently navigates around dynamic obstacles while avoiding congested areas. Our methodology is validated through comprehensive simulations, demonstrating its improvement over existing models in terms of efficiency, adaptability, and scalability. We perform experiments in 2D and 3D environments and with 3-14 degrees of freedom robots. Results show comparable path length and improved computation time for a varying number of dynamic obstacles even when compared with existing method.

Keywords: topological planning · dynamic obstacles · local deflection.

1 Introduction

Robust and efficient failure recovery in robotic motion planning remains a pivotal challenge, especially in dynamic and congested environments. Traditional motion planning algorithms, such as Probabilistic Roadmaps (PRM) and Rapidly-exploring Random Trees (RRT), often struggle to adapt to unexpected obstacles and dynamic changes in the environment, leading to inefficiencies and increased computational overhead [2, 28]. These methods typically focus on static environments and lack mechanisms for real-time adaptation to changing conditions.

To address these limitations, recent research has explored various strategies, including the use of topological reasoning and machine learning techniques. For example, Solovey et al. [19] introduced critical radius concepts for sampling-based planners, improving path planning optimality. Bhattacharya and Ghrist [4] applied topological methods to classify paths, enhancing the robustness of trajectory planning. However, these approaches often do not fully leverage the underlying topological structure of the configuration space in dynamic settings.

In this paper, we propose a novel approach that integrates Discrete Morse Theory with congestion-aware recovery path planning to enhance the robustness and adaptability of robotic systems in presence of dynamic obstacles. Discrete Morse Theory provides a powerful framework for understanding the topology of complex spaces by identifying critical points and essential features such as connectivity and voids [7]. By combining this with Vietoris-Rips complexes, our method ensures that the critical topological properties of the configuration space are preserved, even in the presence of dynamic obstacles.

Our approach dynamically adjusts paths in response to moving obstacles, balancing path length optimization with congestion mitigation. This is achieved through local path deflection, which allows for quick adjustments without the need for complete re-computation of paths. By incorporating congestion awareness, we ensure smoother and more efficient navigation in environments with varying levels of dynamic obstacle traffic density.

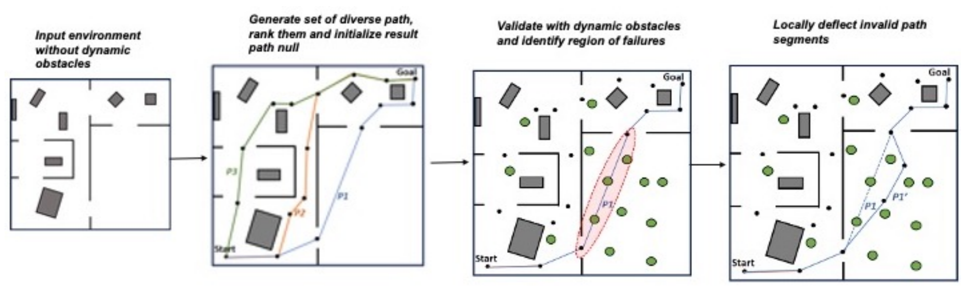


Fig. 1: Concept figure showing diverse paths and path deflection action due to the presence of dynamic obstacles.

This work makes the following key contributions as illustrated in Figure 1:

- Our algorithm incorporates a novel method for local path deflection in the presence of dynamic obstacles. By identifying invalid path segments and adjusting them locally while preserving the global path structure, our approach avoids the need for complete re-computation of paths, thus significantly improving computational efficiency and adaptability in dynamic environments.
- We develop a congestion-aware path planning framework that balances between topologically diverse paths and local re-direction of path segments. This dual approach minimizes congestion due to dynamic obstacle and optimizes path length, ensuring smoother and more efficient navigation in environments with varying levels of obstacle traffic density.
- The use of Vietoris-Rips complexes and Discrete Morse Theory in our path planning framework provides strong topological guarantees. This method captures essential topological properties of the configuration space, ensuring that paths remain connected and valid even in the presence of dynamic changes, thus enhancing the reliability and robustness of the planner.

We validate our proposed methodology through comprehensive simulations in various 2D and 3D environments with robots of different degrees of freedom. Our experiments demonstrate significant improvements in path length, computation time to solve a query and scalability compared to existing methods.

2 Related Work

Decomposing complex environments into simpler subspaces significantly improves path planning efficiency [10, 29]. Results in [14] highlight how Voronoi diagrams simplify space, aiding obstacle avoidance for autonomous vehicles and reducing computational complexity. Similarly, Fabian in [6] demonstrate how combining topological and geometric data through a novel SLAM (simultaneous localization and mapping) technique improves robot navigation and accuracy in dynamic environments. Several survey have looked at existing work in this domain [18, 30]. Work in [13] underscore the importance of generating diverse paths for robust navigation, introducing a probabilistic algorithm that adapts to various constraints. Recent studies, such as those by Jia et al. [12], have focused on decentralized approaches that allow individual robots to make real-time decisions based on local observations, significantly reducing the likelihood of collisions and traffic congestion in crowded areas. These algorithms leverage predictive models to anticipate the movements of dynamic obstacles, enabling smoother navigation.

Phillips and Likhachev proposed a dynamic obstacle avoidance approach by augmenting the roadmap with safe intervals [15]. This approach assumes the robot can wait and finds the optimal path with respect to travel time in the entire roadmap whereas our approach assumes continuous motion for the robot. Additionally scalability to high-dimensional robot planning is dependent on the size of the roadmap.

A most recent work considered the use of Z2 homology to compute path diversity [5]. The computational cost of Z2 homology versus Discrete Morse Theory approaches in path planning is influenced by specific implementation details, the complexity of the environment, and the desired path resolution. While Z2 homology demands extensive algebraic computations to examine topological spaces, making it computationally intensive, Discrete Morse Theory which we use in this work simplifies this by concentrating on a space's critical points, thus minimizing the data required to depict topological characteristics.

In [27] Wang et al. propose a novel approach to multi-robot path planning that aims to reduce congestion without requiring inter-robot coordination. The method leverages topological reasoning to generate topologically distinct paths, ensuring robots are evenly distributed across the environment. This is achieved through a combination of stochastic path assignment and the use of traffic density maps to predict and avoid high-congestion areas. Our approach in this paper combines local and global path adjustments to ensure optimal path selection. It evaluates and adjusts paths locally in response to dynamic obstacles and globally through topologically diverse path selection. This dual approach ensures that valid paths are robust and available.

Research by Tedrake et al. [1, 9] proposes a decomposition of the free space into convex sets. The work provides a computationally efficient safety certification of configurations and trajectories. Our proposed work has a looser objective of assembling free space from more general regions. We will achieve this by identifying convenient patches bounded by levels of a Discrete Morse function and assembling motion planning solutions within patches into global solutions [23]. Our approach will have unique utility in problems where a geometric representation of obstacles is not provided (non-holonomic constraints, loop-closure constraints). It will use exclusively free space samples without previous knowledge of the underlying geometry.

Another closely related work on topological path planning is done by Bhattacharya and Ghrist [4]. Their contribution is in the classification of paths or regions into topologically and geometrically diverse classes [17, 4]. The topological and geometric diversity is then applied in complex path planning problems such as tethered, cabled robots and multi-robot applications. The topological diversity is encoded by augmented graph representing homotopy and homology classes in the configuration space. The augmentation is often computationally expensive and intractable operation in high dimensional planning problems with complex configuration space obstacles [16, 20, 26]. Whereas the topological diversity proposed by our work is scalable to high dimensional planning problems and invariant to the shape or geometry of obstacles. This allows for the generation of topo-geometrically diverse paths in applications where the shape and geometry of obstacles are intractable such as high-genus 3D objects and high curvature regions. As our approach builds a simplified complex from our previous work [25], the global optimal planning algorithms on simplicial-complex proposed in [3, 27] are applicable to our work.

In our previous work [21], we introduced the concept of *minimum path violation (MPV)* as a pre-cursor to failure recovery in motion planning applications. We define MPV as finding a collision free path with minimum difference in number of configurations from possibly invalid straight line path connecting start and goal configurations. In order to realize MPV, we introduced a novel ranking function to sequence a set of homological diverse paths. The algorithm identifies these homologically diverse paths [22]. The critical points are representatives near important features of obstacle space geometry (regions of curvature for density based Morse function). The novel ranking system realizes MPV by ranking the diverse paths based on low path length, low node visibility (configurations in narrow passages) and high edge expansiveness (long collision-free segments) in the stated order.

3 Preliminaries

3.1 Configuration Space and Free Space

The configuration space C of the robot, represents all possible positions and orientations the robot can attain. The free space F is the subset of C where the robot does not collide with any obstacles. Mathematically, F can be defined as:

$$F = \{q \in C \mid \text{robot at } q \text{ is collision-free}\}$$

We assume that C is a subset of Euclidean space, typically $C = [0, 1]^d \subset \mathbb{R}^d$ for some fixed dimension d. For any two configurations $x, x' \in C$, the robot can follow a straight-line path $\pi_{x,x'}(p) = (1 - p)x + px'$ for $p \in [0, 1]$.

3.2 Motion Planning Problem

Given start and goal configurations $s, t \in F$, the motion planning problem involves finding a continuous path $\pi : [0, 1] \to F$ such that $\pi(0) = s$ and $\pi(1) = t$.

The path π must remain entirely within F. Formally, the problem is to find π such that:

$$\pi(0) = s, \quad \pi(1) = t, \quad \pi(q) \in F \quad \forall q \in [0, 1]$$

A path $\pi \in \Pi_{F,s,t}$ (the set of all paths from s to t in F) is considered robust if there exists $\delta > 0$ such that the δ-neighborhood of π lies entirely within F.

In this paper, we represent the roadmap (or graph) of the generated C space through simplicial complex constructed using Vietoris-Rips complex. The Vietoris-Rips complex creates a simplicial complexes from a set of points by forming simplices from points that are pairwise close.

Simplicial Complex A simplicial complex is a set of simplices (points, edges, triangles, etc.) that satisfies certain intersection properties. Formally, a simplicial complex K is a set of simplices such that:

$$\forall \sigma \in K, \text{every face of } \sigma \in K \tag{1}$$

and $\forall \sigma, \tau \in K,\ \sigma \cap \tau$ is a face of both σ and τ.

In our prior work [25], we constructed simplicial complex on C space, to remove redundant information e.g., edges that don't improve coverage or connectivity and provide a space approximation of F as a pre-processing step.

Definition 1. *(Vietoris-Rips complex) Given a set X of points in a Euclidean space $\mathbb{R}^d$, the Vietoris-Rips complex $VR_\epsilon(X)$ is the abstract simplicial complex whose k-simplices are the subsets of $k+1$ points in X of diameter at most ϵ. Formally,*

$$VR_\epsilon(X) = \{\sigma \subseteq X \mid diam(\sigma) \leq \epsilon\},$$

where $diam(\sigma) = \max_{x,y \in \sigma} \|x - y\|$ is the diameter of σ.

3.3 Discrete Morse Theory

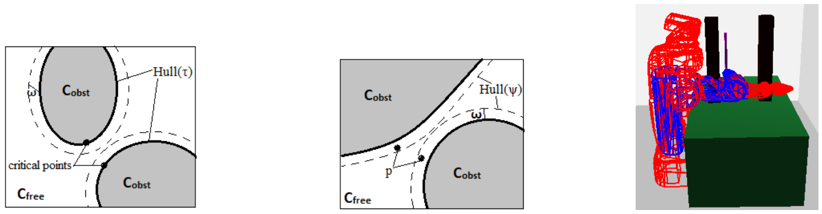
Discrete Morse Theory provides a combinatorial adaptation of classical Morse theory, used to study the topology of cell complexes. It focuses on the critical points of a discrete Morse function, which help in understanding the topology of the configuration space.

Critical Points. A discrete Morse function f on a simplicial complex K assigns real values to the simplices of K such that for any simplex σ:

$$\begin{aligned} \{\tau^{(p+1)} \supset \sigma^{(p)} \mid f(\tau) \leq f(\sigma)\} \leq 1 \\ \{\nu^{(p-1)} \subset \sigma^{(p)} \mid f(\nu) \geq f(\sigma)\} \leq 1 \end{aligned} \tag{2}$$

where $\sigma^{(p)}$ is a p-dimensional simplex, $\tau^{(p+1)}$ is a $(p+1)$-dimensional simplex, and $\nu^{(p-1)}$ is a $(p-1)$-dimensional simplex. Critical points are simplices where these inequalities are strict. We define critical points as the set of simplex cells at which the discrete Morse function f reaches its extremum.

An extension in our work [24], applied discrete Morse theory to the same simplicial complex to identify critical points on the boundary of obstacles and



(a) (b) (c)

Fig. 2: The figure shows the identified critical points on the boundaries of the obstacle space in (a), the feasible critical points (as denoted by p) in free space closer to the critical points at distance ω in (b), and an example environment/robot(c).

form clusters of feasible configurations to deflate paths around different critical points (see Figure 2 for an example).

Let D be the Euclidean distance function that measures the distance between the point $x \in F$ and the nearest point y on the closest obstacle $O_i \in C \backslash F$, that is, $D(x) = \min_{y \in O_i} \|x - y\|$. Let $\Gamma(y, \omega)$ be a density function where $\omega > 0$ and y is the point on the obstacle. The function Γ counts all neighbors close to y in S within distance ω.

Definition 2. *Let f be a discrete Morse function on $VR(S)$ restricted to the vertices of the Vietoris-Rips complex. In our case, this is also the restriction of a Morse function formally defined at any point in C by*

$$f(x) = D(x) \times \Gamma(y, \omega). \tag{3}$$

4 Methodology

4.1 Recovery Path Planner Algorithm

An overview of our recovery path planner algorithm is provided in Algorithm 1. We use our previous work [25, 21] to build initial simplical complex, generate a set of topologically diverse paths and sort the paths based on a ranking function (Lines 1-3). We then post validate the diverse paths against dynamic obstacles in the ordered sequence (Line 6). If a path is found to be invalid in presence of dynamic obstacles (Line 7), a local deflection of the path is attempted to avoid them (Line 8-9). Upon successful deflection, rather than reporting the deflected path, the algorithm continues to iterate over next diverse paths in the sequence. This ensures the best solution path in terms of path cost is returned. However, to improve the efficiency only the diverse paths following in the sequence whose cost is less than that of the deflected path are considered (Line 12). The information for the best cost solution path is updated with each iteration (Lines 13-15). When all the diverse paths within the best solution path's cost are validated and deflected against dynamic obstacles, the stored best solution is returned (Lines 16-17). If no solution path is found, the algorithm defaults to a user-provided planner (DP) to provide a solution (Line 18).

Algorithm 1 Recovery Path Planner

Require: C: Planning space without dynamic obstacles, O: Dynamic obstacles, Q: Query, DP: an optional default planner
Ensure: p: Path to goal

$G \leftarrow$ Build-Initial-Complex(C) [25]
$P \leftarrow$ Get-Diverse-Set-Paths(G, Q) [21]
Sort P by ranking function ρ [21]
$C' = C \cup O$
$p_o = \emptyset$, $w = 0$
for $i = 1$ `to` $|P|$ **do**
    if $p_i \in P$ is invalid in C' **then**
        R= Get-Invalid-Regions(G, p_i)
        p'_i = Local-Deflect(p_i, R)
    else
        $p'_i = p_i$
    $k = \max \{j | j \geq i$ and $\text{cost}(p_j) < \text{cost}(p'_i)\}$
    if p_o is $\emptyset$ or $\text{cost}(p'_i) \leq \text{cost}(p_o)$ **then**
        $p_o = p'_i$
        $w = k$
    if w is equal to i and p_o is not $\emptyset$ **then**
        return p_o
return Call $DP(C'$,$Q)$

4.2 Algorithm Description

Obstacle configuration sampling. Obstacle space is often intractable for high-dimensional planning problem. In order to construct a point cloud representation of the C-obstacle surface, we use Uniform Obstacle based sampling method (UOBPRM) [8], which generates a uniform distribution of configurations on C-obstacle surfaces. UOBPRM uses intersections between C-obstacles and uniformly distributed fixed-length segments to sample configurations in the configuration space. We modify the algorithm to retain the samples in C-obstacle space instead of retaining the free space sample from the line segments intersecting obstacle/free space boundaries. As a result, it forms the obstacle point set, O stated in Section 3.3. Although the distribution of the intersection points provided by UOBPRM is uniform, the required sampling density to approximate the C-obstacle surface is parameter dependent. This is empirically determined depending on the size of the planning environment.

Generation of diverse paths and ranking. Using our previous work, we construct a simplical complex, G in the free region of the planning environment while considering static or pseudo-static obstacles i.e., obstacles that have no change or in-frequent changes in their location. We use the conditions from [25] to approximate the topological coverage of the free space. We apply our discrete Morse function [24] on the samples generated using UOBPRM [8] to distinguish important critical points, CP from all possibly generated obstacle points on C-obstacle surface, O. These critical points are then used to define different clusters of feasible critical points closer to the obstacle surface. Given a query, our algorithm uses the cluster information to direct paths through them to generate a set of diverse paths [22].

Definition 3. *(Path Diversity) Suppose the distance between any two critical points is at least 3ω. Then, two paths p_a and p_b in $\mathbb{P}$ are diverse if $F^{-1}(p_a) \neq F^{-1}(p_b)$ where F defines a relation between the feasible points of a path and the set of critical points that is at ω-clearance, shown in Figure 2.*

F is a function from the set of critical points C to the power set of S, i.e., the set of its subsets. If all $F(c_i)$ are disjoint, such that the critical points are at least 2ϱ apart, then there is a well-defined function F^{-1} from feasible critical points $F(C)$ back to C. Alternatively, $F^{-1}(v)$ can be defined as the pre-image of v under the multi-valued function F. For each path p, we then have $F^{-1}(p)$ which is the union $\bigcup_{v \in p} F^{-1}(v)$ but can also be thought of as a sequence in C because it inherits the order from p.

We also rank these paths based on the metrics given in [21] to extract a pre-informed knowledge of possible feasible paths in the configuration space. Our path ranking properties are defined as below. Details regarding premises and proofs can be found in [21].

Definition 4. *(Path Cost) Taking $\mathbb{P}$ as the set of diverse paths, a path p is given priority if it has the shortest length compared to the remaining paths in P.*

Definition 5. *(Node visibility) Let m define the limit of maximum visibility. If a vertex $v \in V$ has j connected neighbors where $j \leq m$ and $j \neq 0$. Then, we decide the priority of v with the rank score achieved by $m - j$, i.e., the higher the rank score means the highest priority.*

Definition 6. *(Edge expansiveness) Let V_a and V_b be vertices in a roadmap R of the free space topology-approximation graph G. Suppose the shortest unfeasible straight-line connection between V_a and V_b has length D. Considering there exists an edge β connecting two vertices between V_a and V_b and has length $h \leq D$, then we get the priority of β by computing the rank score as $D - h$. Specifically, the smaller the rank score, the higher the priority of the edge β.*

Region generation: The simplicial complex provides a sub-division of the planning environment based on the proximity to the critical points. Each critical point, $cp \in CP$ maps to a set of feasible points, $FC(cp)$. These feasible critical points are the seed vertices used in a flood-fill approach to classify every vertex in the underlying complex into sub-divided regions i.e., the critical point to which the vertex is nearest to.

Given a path generated by the diverse path generation method as stated above, we segment the path based on the association of its' constituent vertices with the sub-divided regions and the feasible critical points the path is passing through. Each path segment is validated with the changed planning environment which includes the dynamic obstacles. The segments and their corresponding regions are evaluated to choose between local recovery in the region or global solution through a topologically diverse path selection.

Algorithm 2 Local Deflect

Require: p: Path, O: Obstacles, G: Complex
Ensure: p': Deflected path
1: Decompose p into segments (p_i, p_{i+1}), initialize $s' = p_0$, $p' = \emptyset$
2: **for** each segment (p_i, p_{i+1}) **do**
3: Plan paths P_i' from s' to $FC(p_{i+1})$ in $G' = \text{SubComplex}(G, p_i, p_{i+1})$
4: Sort P_i' and validate against O
5: **if** $\exists$ Valid $p_i' \in P_i'$ **then**
6: Update s' and append p_i' to p'
7: **else**
8: Remove invalid elements from G' and **goto** Step 3
9: **return** p' or $\emptyset$

Local Deflection: Local deflection is defined as the re-routing of path segments invalidated by dynamic obstacles while preserving the homotopy class of the original path. This is done by limiting the re-routing to the same topological regions that the path originally passed through, with feasible critical points in these regions serving as intermediate waypoints. The cost of the rerouted path is then compared with other topologically distinct paths to determine the best solution. For example, in Fig. 3, three topologically distinct paths—Paths 2, 4, and 1—are computed in the presence of static obstacles and ranked by a cost function. When Path 2 is invalidated by a dynamic obstacle, local deflection identifies an alternative route, Path 3, which follows the same topological class as Path 2. The algorithm selects Path 3 as the solution if its cost is lower than that of Path 4; otherwise, it continues to evaluate Paths 4 and 1 to find the path with the lowest cost.

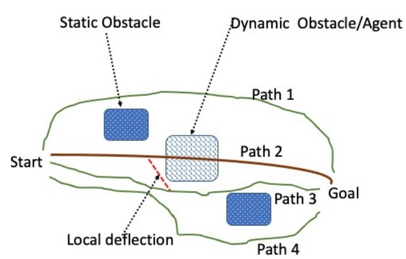


Fig. 3: The figure shows the local deflection of topological diverse path in presence of dynamic obstacle. Path 1, 2, 4 are topologically diverse paths computed considering static obstacles only. Path 3 is locally deflected path from Path 2 when dynamic obstacle is considered.

Algorithm 2 details the local deflection method given a path p from start to goal configuration. The path p is decomposed into a sequence of path segments $[p_i, p_{i+1}]$ where $p_i, i \in (0, |p|)$ is a feasible critical point (Line 1). Each segment is validated with dynamic obstacles in order, and a substitute is computed for any segment invalidated by dynamic obstacles (Lines 2-8). To replace an invalid path segment, the shortest path from p_i to each point in $FC(p_{i+1})$—the set of feasible critical points near p_{i+1}—is computed on the relevant sub-complex of graph G maintaining the homotopy class of the segments (Line 3). These paths are sorted by cost and validated against dynamic obstacles (Lines 4). Invalid edges are removed from the sub-complex, and recalculation continues until no valid paths remain indicating congestion caused by dynamic obstacles (Line 8). The shortest valid path replaces the original segment, and the process repeats with end-point of the last segment as start point (Line 6). If no valid replacement is found, the next topologically distinct path is considered.

4.3 Topological Guarantees

Solovey et al. in [19] explored the critical radius necessary for sampling-based motion planning procedures. The critical radius was defined as $r_n = \gamma n^{-1/d}$, where γ is a constant and d is the dimension of the space. The paper focused on ϵ-nets and their efficacy in covering the configuration space, which is primarily a metric-based approach. While the paper addresses high dimensions through ϵ-nets, it might still struggle with the exponential growth in sample complexity as dimensions increase. This is because traditional metric-based methods do not sufficiently leverage the underlying topological structure of the space, hence there is no topological guarantee.

Theorem 1. *Let C be a compact, connected configuration space, $F \subseteq C$ the free space, and $O \subset C$ the dynamic obstacle space. Consider the initial path $\gamma_0 : [0,1] \to F$ as a continuous function from the interval $[0,1]$ to the free space F. The Recovery Path Planner algorithm guarantees that if γ_0 is feasible (i.e., does not intersect O), then for any continuous deformation of the obstacles, the adjusted path γ_t, obtained via the algorithm, remains in the same homotopy class as γ_0.*

Proof. Consider C, a compact, connected topological space representing the configuration space. The free space $F \subset C$ is the complement of the obstacle space O, i.e., $F = C \setminus O$. The free space F is also assumed to be connected and open.

The initial path $\gamma_0 : [0,1] \to F$ is a continuous function, such that $\gamma_0(0)$ is the start configuration and $\gamma_0(1)$ is the goal configuration. Since F is connected, such a path exists and lies entirely within F.

We construct a simplicial complex G over F using the Vietoris-Rips construction with a filtration parameter ϵ. This complex $G_\epsilon(F)$ serves as a topological approximation of F and retains the homotopy type of F for sufficiently small ϵ. Specifically, $G_\epsilon(F)$ captures the essential topological features (e.g., connected components, holes) of F. Discrete Morse Theory is applied to $G_\epsilon(F)$, identifying critical simplices that correspond to critical topological features (e.g., passageways in F). The path γ_0 is represented within this simplicial complex, ensuring that it follows a route that avoids collapsing through critical points, preserving homotopy classes.

Path Deformation and Local Deflection: As the dynamic obstacles O_t deform over time t, the free space $F_t = C \setminus O_t$ evolves. The Recovery Path Planner adjusts γ_0 to γ_t, ensuring that γ_t remains in F_t. This adjustment is done locally using a simplicial homotopy, which preserves the homotopy class of the path within the evolving free space.

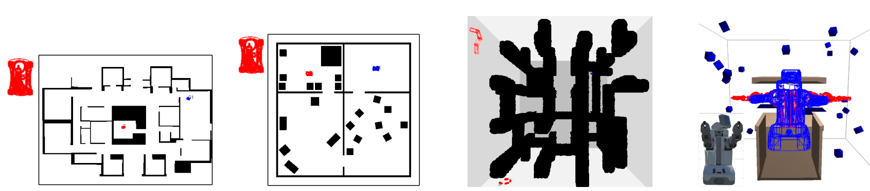
Homotopy Preservation: Since the Vietoris-Rips complex $G_\epsilon(F_t)$ at each time step retains the same homotopy type as the original free space F, and the Recovery Path Planner modifies the path γ_t within this homotopy-preserving structure, the final path γ_t remains in the same homotopy class as γ_0.

Therefore, the Recovery Path Planner guarantees that the adjusted path γ_t maintains the same topological properties (specifically, the homotopy class) as the original path γ_0, ensuring that the path remains feasible and topologically valid despite the presence and evolution of dynamic obstacles.

5 Experiments and Discussion

We conducted experiments on 2D and 3D environments with robots varying from 3 to 14 degrees of freedom (DOF) and varied number of dynamic obstacles. The start and goal positions are shown in red and blue colors respectively.

1. **Office:** A 2D office layout with multiple entry and exit for each room and the robot has 3 DOFs, as shown in Figure 4a. The number of dynamic obstacles ranges from 5 to 25.
2. **House:** It is a 2D house layout with one door opening from one room to other and the robot has 3 DOFs, as shown in Figure 4b. Number of dynamic obstacles ranges from 2 to 10.
3. **Pipes (homogeneous):** It is a 3D environment with cubic pipes having a free-flying 4 linkage robot of 9 DOFs, as shown in Figure 4c. Number of dynamic obstacles ranges from 5 to 25.
4. **Pipes (heterogeneous):** Similar environment from Figure 4c having a 9 DOFs free-flying 4 linkage robot and 6 DOFs dynamic obstacles. Number of dynamic obstacles ranges from 5 to 70.
5. **Table-top:** 3D table top environment with a 14 DOFs (2 spherical and 10 revolute) fixed base PR2 robot. Figure 4d shows the robot with dynamic agents (cubes) in it. Number of dynamic free flying obstacles with 6 DOF ranges from 3 to 18.



(a) (b) (c) (d)

Fig. 4: Environments Studied: (a) Office (KukaYouBot base) (b) House (KukaYouBot base) (c) Pipes tunnel (4-linkage robot) (d) Tabletop (PR2 robot)

We use uniform random sampling and radius-based straight line connection with Euclidean distance metric to generate the initial complex. RAPID collision detection library [11] is used to check the validity.

In each environment, the degrees of freedom (DOFs) and geometries of all dynamic obstacles are the same, but they differ from those of the robot in the Pipes (heterogeneous) and Tabletop environments to demonstrate the general applicability of our method with varying obstacles. To simulate the path for dynamic obstacles, a graph using PRM and uniform sampling is constructed for each environment, considering only static obstacles and the static parts of the robot (such as the base, torso, and head of PR2). Random walks are generated for each dynamic obstacle from the graph, and the obstacles move back and forth along these paths at a uniform velocity, similar to the robot. All the experiments are conducted on a Ubuntu 20.04.6 machine with 4.2 GHz processor, 16 GB memory. The algorithm is implemented in C++. For brevity, we use DMT for Discrete Morse Theory in the coming discussions.

5.1 Computation Time

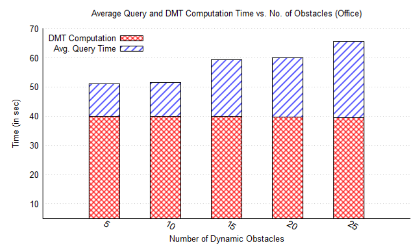
Fig. 5 shows the computation time in the experimental environments with varied number of dynamic obstacles. The average query time includes the time to compute diverse number of paths and finding a solution in the presence of dynamic obstacles. The DMT computation time includes time to construct the obstacle surface representation (without considering the dynamic obstacle), building of the complex, finding the critical points using discrete Morse function and subdivision of the free planning space. The time reported is averaged over 10 runs with varying seeds for the uniform distribution sampling.

The average query time increases with the increase in number of obstacles in all of the environments as demonstrated in the plots. This can be attributed to the time spent in local recovery for the diverse paths. The increase in time is smaller with a low number of dynamic obstacles compared to a high number of obstacles. This is caused by the number of iterations spent by the local recovery for each diverse path to avoid the large number of dynamic obstacles. The increase in the time is less substantial in Pipes environment than other environments, considering number of obstacles as high as 70. The robot in the Pipes environments has high DOF without any joint restriction (as compared to PR2 in Tabletop) except self collision checking which is done during DMT computation. This increases the maneuverability of the robot and thereby connections in the underlying complex where the robot can locally deflect to.

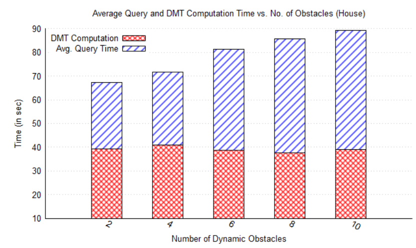
The DMT computation time remains almost similar in most of the environments and is not affected by the increase in the number of dynamic obstacles. This is expected as the number of dynamic obstacles does not contribute to the computations involved. The slight variations can be attributed to the UOBPRM sampling for obstacle points. The DMT computation is the high in Tabletop and Pipes environment due to time spent in self-collision check. However, the total computation does not exceed 200 secs indicating an application to robotic system with high DOF and is scalable to high number of dynamic obstacles.

5.2 Path Quality

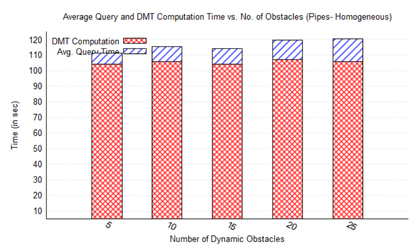
Path Length Fig. 6 shows the path length information of the solution path before and after the local recovery deflection in the experimental environments with varying number of dynamic obstacles. The deflection path length is the



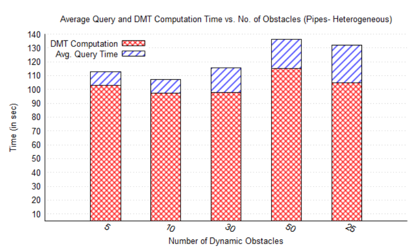
(a) Office



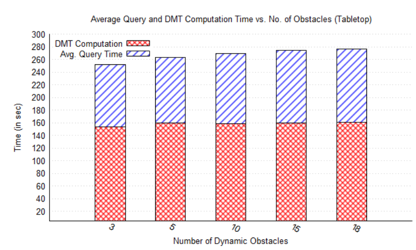
(b) House



(c) Pipes (Homogeneous)



(d) Pipes (Heterogeneous)



(e) Tabletop

Fig. 5: Average Query Time and DMT Computation Time vs Number of Dynamic Obstacles.

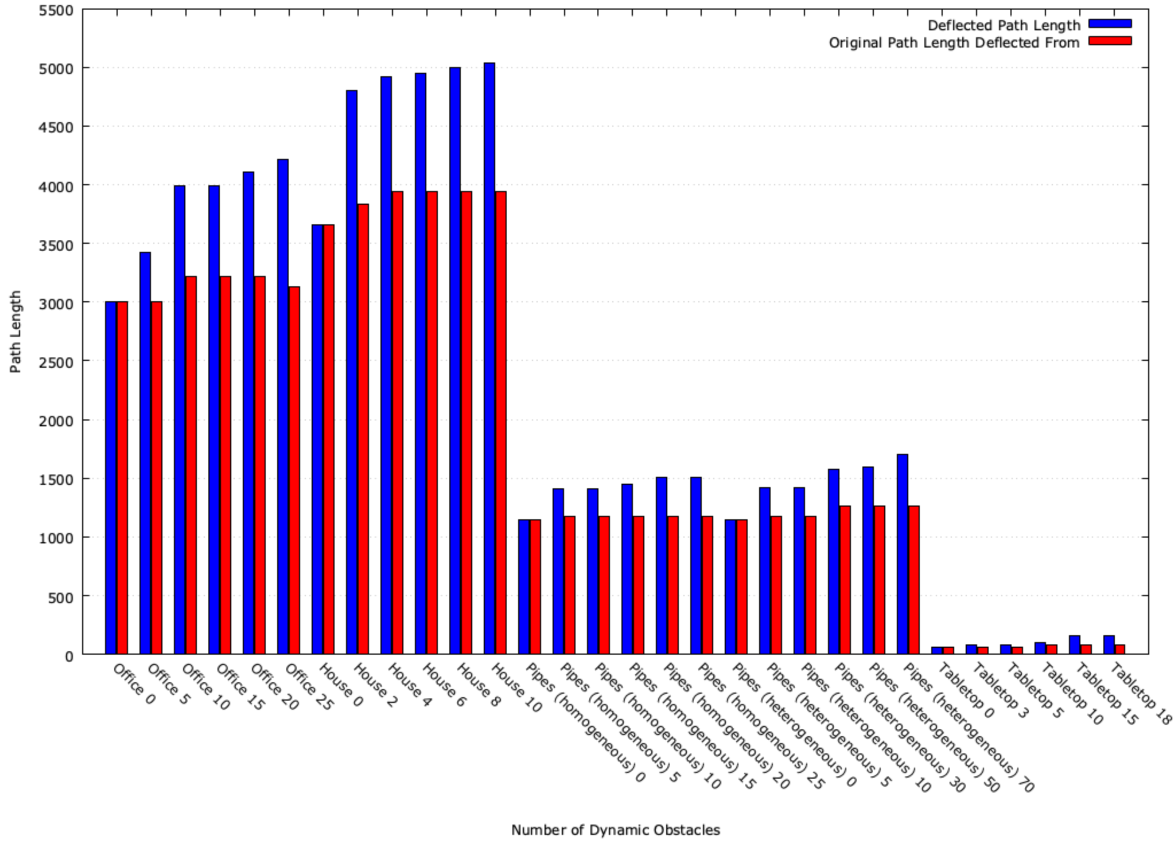
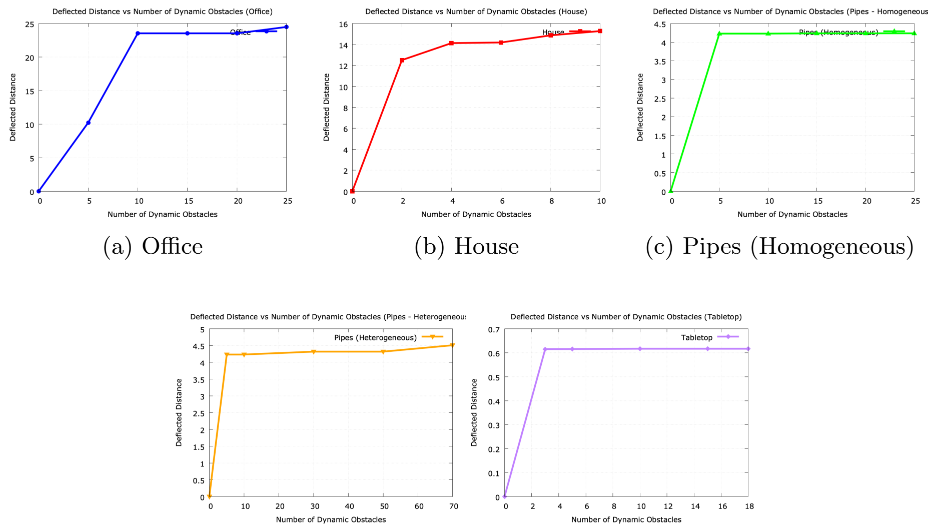


Fig. 6: Path Length information for all Environments showing the original path length and the deflection path length

length of the solution path provided by the algorithm. The original path length is the length of the diverse path prior to local recovery deflection chosen by the algorithm as the solution path.

As observed in the figure, the deflected path length is more than the original path length it deflected from in all the environments. The increase in deflected path length from the original path length is more with high number of dynamic obstacles than with less number of dynamic obstacles. This is expected as increase in dynamic obstacles will cause increase in deflection to avoid them. The increase in path length from the original is more in 2D environments (Office and House) than the other 3D environments (Pipes and Tabletop). This can be attributed to the increased dimension of the robot in 3D environments causing increase in connectivity in the underlying complex.

Local Deflection Fig. 7 shows the maximum deflected distance of the solution path from the original diverse path it is deflecting from in the experimental en-



(a) Office (b) House (c) Pipes (Homogeneous)

(d) Pipes (Heterogeneous) (e) Tabletop

Fig. 7: Maximum Deflected Distance vs Number of Dynamic Obstacles for Different Environments

vironments with varying numbers of dynamic obstacles. The deflected distance d is calculated as the maximum of the minimum Euclidean distance of the deflected path, p_{def}'s intermediate configurations, $b \in p_{def}$, from the original path p's intermediate configurations , $a \in p$ as given in equation below.

$$d = \max(\min ||a - b||), \forall a \in p, b \in p_{def}. \quad (4)$$

As observed from the figure, the maximum deflected distance remains nearly same with the increase in number of dynamic obstacles, even though there is increase in path length, as observed in Fig. 6. This can be attributed to the design of the local recovery planner redirecting the invalid path segments to other feasible critical points in the same topological classified region. These feasible critical points are guaranteed by construction to be within 2ω-distance of each other, where ω is the radius used for density computation. Moreover, a sharp increase in deflected distance will attribute to increase in the path length causing the local recovery planner to choose the next diverse path. However, conversely, increase in the path length does not necessarily imply an increase in the maximum deflected distances.

Comparison with Wang et. al. [27]. We do a comparative analysis of our algorithm with an adapted version from the work presented in [27] for single robot scenario. We used the traffic density estimation along with the path length in ordering of the diverse paths for the single robot that serve as the reference paths. The first valid locally deflected path is returned as the solution. The path length of solution path and the original reference path along with query time

Table 1: Comparison of Path Length and Average Query Time

Environment	No. of Obstacles	Path Length		Avg. Query Time		Orig. Path Length	
		T1	T2	T1	T2	T1	T2
Office	5	3420	3420	62.1	11.2	3005	3005
	10	4092	3992	111	11.6	3121	3215
	15	4092	3992	172	19.5	3121	3215
	20	4220	4110	248	20.3	3121	3215
	25	4220	4220	315	26.0	3121	3129
House	2	4806	4806	25.6	28.0	3827	3832
	4	4973	4921	97.0	30.9	3861	3944
	6	4991	4954	103	42.6	3861	3944
	8	5003	5003	131	48.2	3944	3944
	10	5033	5033	155	50.3	3944	3944

is provided in Table 1. The adapted version of [27] is referred as T1 and our approach as T2. The table demonstrates the length of the solution path returned by our approach is comparable and sometimes better than that returned by the adapted version. However, the average query time in the adapted version is significantly higher than our approach. This is due to the pre-validation of all the diverse path with every dynamic obstacle path to generate the traffic density estimate. Using historic data of traffic estimate as mentioned in [27] might reduce this computation overload. However, for high dimensional robots it is not apparent how that can be incorporated.

6 Conclusion

In this paper, we present a novel recovery path planning method in environments containing dynamic obstacles. Our method integrates Discrete Morse Theory with local deformation of paths to efficiently find near optimal solution for the robot to follow while avoiding dynamic obstacles. Our evaluations show applicability of our approach to robots of varying degrees of freedom and scalability with respect to varying number of dynamic obstacles.

In future work, we would explore other reactive behavioral policies for the robot such as waiting, in conjunction to local deflection method. Our local deflection method and congestion determination is greedy in nature, which causes the solution path returned to be near optimal. The combination of other behavioral policies will facilitate finding optimal solution in environments with obstacles of non-uniform velocity and varied morphology.

References

1. Amice, A., Dai, H., Werner, P., Zhang, A., Tedrake, R.: Finding and optimizing certified, collision-free regions in configuration space for robot manipulators. In:

International Workshop on the Algorithmic Foundations of Robotics. pp. 328–348. Springer (2022)
2. Barraquand, J., Kavraki, L., Latombe, J.C., Li, T.Y., Motwani, R., Raghavan, P.: A random sampling scheme for path planning. In: Robotics Research: The Seventh International Symposium. pp. 249–264. Springer (1996)
3. Bhattacharya, S.: Towards optimal path computation in a simplicial complex. International Journal of Robotics Research (IJRR) **38**(8), 981–1009 (June 2019). https://doi.org/10.1177/0278364919855422, dOI: 10.1177/0278364919855422
4. Bhattacharya, S., Ghrist, R.: Path homotopy invariants and their application to optimal trajectory planning. In: Proceedings of IMA Conference on Mathematics of Robotics (IMAMR). St Anne's College, University of Oxford (September 9-11 2015)
5. Bhattacharya, S., Ghrist, R., Kumar, V.: Persistent homology for path planning in uncertain environments. IEEE Transactions on Robotics **31**(3), 578–590 (2015). https://doi.org/10.1109/TRO.2015.2412051
6. Blochliger, F., Fehr, M., Dymczyk, M., Schneider, T., Siegwart, R.: Topomap: Topological mapping and navigation based on visual slam maps. In: 2018 IEEE International Conference on Robotics and Automation (ICRA). pp. 3818–3825. IEEE (2018)
7. Burton, B.A., Lewiner, T., Paixão, J., Spreer, J.: Parameterized complexity of discrete morse theory. ACM Transactions on Mathematical Software (TOMS) **42**(1), 1–24 (2016)
8. (Cindy), H.Y., Thomas, S.L., Eppstein, D., Amato, N.M.: UOBPRM: A uniformly distributed obstacle-based PRM. In: Proc. IEEE Int. Conf. Intel. Rob. Syst. (IROS). pp. 2655–2662 (2012). https://doi.org/10.1109/IROS.2012.6385875, http://dx.doi.org/10.1109/IROS.2012.6385875
9. Dai, H., Amice, A., Werner, P., Zhang, A., Tedrake, R.: Certified polyhedral decompositions of collision-free configuration space (2023)
10. González, D., Pérez, J., Milanés, V., Nashashibi, F.: A review of motion planning techniques for automated vehicles. IEEE Transactions on intelligent transportation systems **17**(4), 1135–1145 (2015)
11. Gottschalk, S., Lin, M., Manocha, D.: Obb-tree: A hierarchical structure for rapid interference detection. Technical Report TR96-013, University of N. Carolina, Chapel Hill, CA (1996)
12. Jia, Y., Song, Y., Xiong, B., Cheng, J., Zhang, W., Yang, S.X., Kwong, S.: Hierarchical perception-improving for decentralized multi-robot motion planning in complex scenarios. IEEE Transactions on Intelligent Transportation Systems (2024)
13. Kuderer, M., Sprunk, C., Kretzschmar, H., Burgard, W.: Online generation of homotopically distinct navigation paths. In: 2014 IEEE International Conference on Robotics and Automation (ICRA). pp. 6462–6467. IEEE (2014)
14. Li, L., Zuo, X., Peng, H., Yang, F., Zhu, H., Li, D., Liu, J., Su, F., Liang, Y., Zhou, G.: Improving autonomous exploration using reduced approximated generalized voronoi graphs. Journal of Intelligent & Robotic Systems **99**, 91–113 (2020)
15. Phillips, M., Likhachev, M.: Sipp: Safe interval path planning for dynamic environments. 2011 IEEE International Conference on Robotics and Automation pp. 5628–5635 (2011), https://api.semanticscholar.org/CorpusID:16210040
16. Ramaithitima, R., Whitzer, M., Bhattacharya, S., Kumar, V.: Automated creation of topological maps in unknown environments using a swarm of resource-constrained robots. IEEE Robotics and Automation Letters (RA-L) **1**(2), 746–753 (January 2016), dOI: 10.1109/LRA.2016.2523600
17. Sahin, A., Bhattacharya, S.: Optimal path planning in distinct topo-geometric classes using neighborhood-augmented graph and its application to path planning for a tethered robot in 3d (2023), arXiv:2306.01203 [cs.RO]

18. Saputra, M.R.U., Markham, A., Trigoni, N.: Visual slam and structure from motion in dynamic environments: A survey. ACM Computing Surveys (CSUR) **51**(2), 1–36 (2018)
19. Solovey, K., Kleinbort, M.: The critical radius in sampling-based motion planning. The International Journal of Robotics Research **39**(2-3), 266–285 (2020)
20. 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)
21. Upadhyay, A., Ghosh, M., Ekenna, C.: Minimal path violation problem with application to fault tolerant motion planning of manipulators. In: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2023)
22. Upadhyay, A., Goldfarb, B., Ekenna, C.: A topological approach to finding coarsely diverse paths. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 6552–6557. IEEE (2021)
23. Upadhyay, A., Goldfarb, B., Ekenna, C.: Incremental path planning algorithm via topological mapping with metric gluing. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 1290–1296 (2022). https://doi.org/10.1109/IROS47612.2022.9981379
24. Upadhyay, A., Goldfarb, B., Wang, W., Ekenna, C.: A new application of discrete Morse theory to optimizing safe motion planning paths. In: International Workshop on the Algorithmic Foundations of Robotics. pp. 18–35. Springer (2022)
25. Upadhyay, A., Wang, W., Ekenna, C.: Approximating cfree space topology by constructing vietoris-rips complex. In: Proceedings 2019 IROS: IEEE/RSJ International Workshop on Intelligent Robots and Systems. IEEE (2019)
26. Wang, X., Bhattacharya, S.: A topological approach to workspace and motion planning for a cable-controlled robot in cluttered environments. In: International Conference on Intelligent Robots (IROS) (1-5, October 2018)
27. Xiaolong Wang, A.S., Bhattacharya, S.: Coordination-free multi-robot path planning for congestion reduction using topological reasoning. Journal of Intelligent and Robotic Systems (JINT) (2023), dOI:10.1007/s10846-023-01878-3
28. Xie, D., Morales, M., Pearce, R., Thomas, S., Lien, J.M., Amato, N.M.: Incremental map generation (img). In: Algorithmic Foundation of Robotics VII: Selected Contributions of the Seventh International Workshop on the Algorithmic Foundations of Robotics. pp. 53–68. Springer (2008)
29. Xiong, C., Chen, D., Lu, D., Zeng, Z., Lian, L.: Path planning of multiple autonomous marine vehicles for adaptive sampling using voronoi-based ant colony optimization. Robotics and Autonomous Systems **115**, 90–103 (2019)
30. Yu, S., Fu, C., Gostar, A.K., Hu, M.: A review on map-merging methods for typical map types in multiple-ground-robot slam solutions. Sensors **20**(23), 6988 (2020)