Toggle PRM: Simultaneous Mapping of C-free and C-obstacle - A Study in 2D -

Jory Denny and Nancy M. Amato

***Abstract*— Motion planning is known to be difficult. Probabilistic planners have made great advances, but still have difficulty for problems that require planning in narrow passages or on surfaces in C_{space}. This work proposes Toggle PRM, a new methodology for PRMs that simultaneously maps both free and obstacle space. In this paper, we focus on 2 DOF problems and show that mapping both spaces leads to increased sampling density in narrow passages and to improved overall efficiency as compared to previous sampling based approaches.**

I. Introduction

In robotics, one challenging problem is planning a valid (e.g., collision-free) path for a movable object (robot) in an environment. This has been extensively studied and is commonly called *Motion Planning* [20]. Motion planning is not only used in robotic applications but even such domains as gaming/virtual reality [13] and bioinformatics [22].

Sampling-based planners [10] were a major breakthrough in motion planning. These algorithms were able to solve many previously unsolved problems, especially for high-dimensional configuration space (C_{space}). While these methods have been shown to be *probabilistically complete*, narrow passages, or small volume regions of C_{free}, remain difficult for them to map. In particular, it has been shown that the volume of such passages impacts the efficiency of a sampling-based planner [8]. The intuition is that the smaller the relative volume of the corridor, the more difficult it is to generate samples in it. As discussed in Section II, there have been many variants proposed which aim to address this weakness of sampling-based planners [1], [3], [7], [23].

This paper introduces a new approach to sampling-based planning. In the new method, called Toggle PRM, both free space, C_{free}, and collision space, C_{obst}, are mapped in an integrated, coordinated fashion. The intuition behind this strategy is that each connection attempt (local planner result), whether successful or not, provides important information about the connectivity of both spaces, and moreover, witnesses to unsuccessful connections between configurations in one space provide useful configurations in the other space. For example, a failed connection between two collision configurations on either side of a narrow passage would lead to the discovery of a configuration in the narrow passage. Moreover, the probability of finding such a narrow passage configuration would not depend on the volume of the passage, but only on the fact that a connection was attempted between configurations in the bounding obstacles.

This research supported in part by NSF Grants EIA-0103742, ACR-0081510, ACR-0113971, CCR-0113974, ACI-0326350, CRI-0551685, CCF-0833199, CCF-0830753, by the DOE, Chevron, IBM, Intel, HP, and by King Abdullah University of Science and Technology (KAUST) Award KUS-C1-016-04.

J. Denny and N. M. Amato are with the Parasol Lab., Dept. of Computer Science and Engineering, Texas A&M Univ., College Station, Texas, 77843-3112, USA {jdenny,amato}@cse.tamu.edu

In this paper, we focus on 2 DOF problems and show that this paradigm shift from only mapping the free space to the coordinated mapping of both free and collision spaces results in significant improvements to the effectiveness and efficiency of sampling-based planners. Particular contributions of this work include:

- Introducing a method, Toggle PRM, that maps C_{free} and C_{obst}.
- Sketching theoretical arguments that explain increased sampling of configurations inside narrow passages for arbitrary two-dimensional problems.
- Providing experimental results that confirm that Toggle PRM sampling in narrow passages is less dependent on the volume of the narrow passage and of the volume of the obstacles surrounding the narrow passage.

The above establishes that Toggle PRM addresses one of the major challenges of previous sampling-based methods for 2 DOF problems. We also provide experimental results that indicate Toggle PRM has similar benefits for higher DOF problems.

II. Preliminaries and Related Work

In this section, preliminaries of motion planning and related work are discussed. After providing preliminary definitions, an overview of the various known PRM variants is given. Secondly, work that utilizes collision information to guide sampling is surveyed. These are usually more complex algorithms utilizing machine learning methods. Lastly, methods which model C_{space} to predict collisions are described.

A. Preliminaries

A robot is a movable object whose position and orientation can be described by n parameters, or *degrees of freedom* (DOFs), each corresponding to an object component (e.g., object positions, object orientations, link angles, link displacements). Hence, a robot's placement, or configuration, can be uniquely described by a point $(x_1, x_2, ..., x_n)$ in an n dimensional space (x_i being the ith DOF). This space, consisting of all possible robot configurations (feasible or not) is called *configuration space* (C_{space}) [15]. The subset of all feasible configurations is the *free* C_{space} (C_{free}), while the union of the unfeasible configurations is the *blocked* or *obstacle* C_{space} (C_{obst}). Thus, the motion planning problem becomes that of finding a continuous trajectory in C_{free}

connecting the points in C_{free} corresponding to the start and the goal configurations. In general, it is intractable to compute explicit C_{obst} boundaries [20], but we can often determine whether a configuration is feasible or not quite efficiently, e.g., by performing a collision detection (CD) test in the *workspace*, the robot's natural space.

Randomized sampling-based motion planners explore C_{space} and produce a data structure containing representative feasible configurations and some information about the connectivity of C_{free}. One such planner, the *Probabilistic Roadmap Method* (PRM) [10], builds a roadmap (graph) in the free C_{space}. The first phase in this process, *node generation*, is where collision-free configurations are sampled and added as nodes to the roadmap. In the second phase, *node connection*, neighboring nodes are selected by a *distance metric* as potential candidates for connection. Then, simple *local planners* attempt connections between the selected nodes; successful connections are represented as roadmap edges.

B. PRM variants

Although the initial PRMs, described in Section II-A, were successful in solving many problems previously thought unsolvable, they were not successful in problems which must sample small volumes of C_{space} (e.g., surfaces or corridors). To address this deficit, many variants of the method have been proposed, e.g., [2], [12], [19]. The performance of these methods is dependent on the problem instance. Below is a description of the PRM variants most related to this work.

Obstacle-Based PRM (OBPRM) [1] finds configurations near the surfaces of obstacles in the environment. Firstly, a configuration is sampled uniformly at random. Then a random direction is selected, and the configuration is moved along this direction until the configuration changes validity. The valid configuration from this change (before or after depending on initial validity value) is kept. This algorithm successfully generates configurations near obstacles. However, it does not sample with a known density on surfaces.

Gaussian PRM [3] attempts to find configurations at a Gaussian distance d away from obstacles. First a configuration is generated uniformly. A second configuration is generated at a Gaussian distance d away along a random directional vector. If and only if one of the configurations is valid and the other configuration is invalid, then the valid configuration is added to the roadmap.

Bridge Test PRM [7] is similar to Gaussian PRM. It generates a node in collision. Then generates another sample configuration at a small distance λ away from the first sample on a random directional vector. If the configuration is also in collision, the midpoint between these configurations is evaluated. The midpoint is kept if and only if it is valid. Hence, this test attempts to bridge the gap and generate configurations in a narrow passage. However, this test may fail many times before successfully bridging a gap, depending on the narrow passage.

Medial Axis PRM (MAPRM) [14], [23] samples on the medial axis of the free space. A configuration is generated at random, valid or invalid. Then the configuration is pushed towards the medial axis of the free space based upon its C_{space} clearance. MAPRM provably improves the probability of sampling in narrow passages over uniform random sampling, but can be computationally intensive even if approximations are taken.

All the variants are useful in certain cases, each providing their own benefits and weaknesses. Only one of the methods, MAPRM, is proven to generate a higher density of configurations within narrow passages as compared to uniform random sampling. However, in practice, all of the variants improve over uniform random sampling. Most of these methods utilize validity information to aid generating samples within narrow passages. This work presents another method, Toggle PRM, which both aides in generating samples in narrow passages and reduces computational cost as compared to other methods.

C. Variants utilizing collision information

There have been many improvements upon PRM by using information gain or learning techniques to guide sampling in specific regions of the environment. Many of these methods utilize information about nodes in collision, however none of them create a map of C_{obst}. This section briefly describes some of these methods.

Feature Sensitive Motion Planning [16], [18] applies a supervised learning technique to aid PRMs in solving problems in heterogeneous environments. This work first creates a small roadmap. Regions are recursively identified, broken up, and merged to form homogeneous spaces of the environment. Regions are classified as free, cluttered, narrow passage or blocked based upon an off-line decision tree. Finally, a specific sampler is applied to the region based on this classification. This method utilizes collision information in an entropy based region partitioning scheme [18].

Region-Sensitive Adaptive Motion Planner (RESAMPL) [21] is a feature sensitive motion planning aid designed to help both multi-query and single-query planning methods. In this work, regions are defined by spheres centered at a node surrounding its nearest neighbors. Both valid and invalid nodes are kept from sampling, and regions are classified by entropy. Regions are labeled as free, surface, narrow, or blocked. All regions are represented in a *region graph* whose nodes are the regions, edges are defined by overlap between regions, and weights are based upon the classification transfer. The information gain and *region graph* bias sampling to improve upon Feature Sensitive Motion Planning [16].

Wong and Jenkin [24] propose a new planning method, which samples uniformly, and classifies each sample as either valid or invalid. If a pair of invalid configurations in C_{obst} are less than a predefined $2r$ distance apart and not from the same obstacle, then the gap between the samples is considered a narrow passage. A region is defined as the centroid between these two points with radius $2r$. Sampling is biased in these centroids.

These methods utilize information from C_{obst}, but none utilize information of the connectivity of the obstacle space, as is proposed in this work.

D. Modeling C_{space}

Other efforts, such as [4], [5], [17], seek to model C_{space}. The goal in these is reduction in computation time rather than strictly guiding sampling. In [4], [5] a machine learning technique, locally weighted regression, is used to create an approximate model of C_{space} avoiding unnecessary collision checks. They use their approximate model to bias sampling toward areas which improve the model, called active sampling. They create a roadmap, called a predictive roadmap, which uses this model and active sampling for multiple query scenarios. These methods require an accurate and complex model for predicting collisions. These methods improve over uniform random sampling, and significantly reduce the computation time.

III. Toggle PRM

An overview of *Toggle PRM* is shown in Algorithm 1. In this algorithm, the major difference from traditional PRM methods is roadmaps of both C_{free} and C_{obst} are constructed. Additionally, the local planning method returns a witness configuration if the connection attempt fails. Both of these allow important information gain to occur that Toggle PRM utilizes to map both spaces in a coordinated manner. The algorithm takes in an environment describing a motion planning problem, a sampler to generate nodes (e.g., uniform random sampling), and a connector, which is a combination of a local planner (e.g., straight line) and a nearest neighbor finder (e.g., k-closest). We start by initializing two empty roadmaps. While the map construction is not done (e.g., until the map is a certain size, a query is solved, maximum attempts made, etc.), then the following happens: samples are generated in both C_{free} and C_{obst}, and connections are attempted in both maps. Witnesses returned by failed connections are added to the opposite space's roadmap. When connections are attempted in C_{obst}, toggling the meaning of validity is necessary for the local planner to be accurate for C_{obst}. The algorithm repeats until the stopping criterion is reached, or it is shown to be unreachable.

The intuition of Toggle PRM is that using the information gained regarding the connectivity of the spaces, i.e., from nodes witnessing failed connections by local planners, the roadmaps of both spaces grow in important regions. Thus, successive iterations of Toggle PRM will expand the roadmap. For example in Figure 1, let sample s be a node within a narrow passage and n_1 and n_2 be samples outside the narrow passage. When connections are attempted, the failure witnesses, x_1 and x_2, are stored in the collision roadmap G_{obst}. These can be connected to yield yet another sample in the narrow passage spatially within the triangle created by s, n_1, and n_2. Hence, mapping both spaces through iterative applications of Toggle PRM allows expansion of the connected components within narrow passages.

Algorithm 1 *Toggle PRM*. Local planners and connectors are modified to return configurations from failed connection attempts.

Input: Environment e, Sampler s, Connector c
Initialize Roadmap Graph G_{free} and G_{obst} to $\emptyset$
while $!done$ **do**
  Queue q
  $nodes \leftarrow s.Sample(e)$
  $q.enqueue(nodes)$
  while $!done$ && $!q.isEmpty()$ **do**
    Node $n \leftarrow q.dequeue()$
    if n is valid **then**
      Add n to G_{free}
      $collisionNodes \leftarrow c.Connect(G_{free})$
      $q.enqueue(collisionNodes)$
    else $\{n$ is invalid$\}$
      Toggle Validity
      Add n to G_{obst}
      $validNodes \leftarrow c.Connect(G_{obst})$
      $q.enqueue(validNodes)$
      Toggle Validity
    end if
  end while
end while

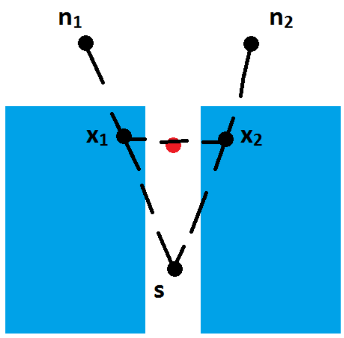


Fig. 1. Toggle PRM example.

A specific advantage of Toggle PRM is increased probability of sampling within narrow passages. This is argued theoretically in Section III-A and shown experimentally in Section IV.

We note that once a successful sample is obtained in a narrow passage, the passage itself may be efficiently explored by a planner such as a rapidly exploring random tree (RRT) [11]. Indeed, many efforts have established that the best planner is in fact an adaptive strategy that selects the best planner for the current situation (e.g., Hybrid PRM [9]). This is not the focus of this work, however. Here, we focus on understanding this new approach to sampling-based planning.

A. Narrow passages

Narrow passages are notoriously problematic for probabilistic planners. In this section, we argue that Toggle PRM can effectively find samples in narrow passages in 2 DOF

C_{spaces} regardless of the volume of the passage or the volume of the obstacles surrounding the passage.

Proposition 1: In Toggle PRM, witnesses to failed connection attempts increase the probability of sampling in a narrow passage in 2 DOF C_{space} compared to uniform random sampling.

Proof: Let A be the set of configurations in C_{free} composing the narrow passage, and let O denote the set of configurations in C_{obst}.

The probability of sampling in A given two samples taken uniformly at random is one minus the probability that both samples fall outside A.

$$P_{\text{uniform}}(2) = 1 - \left(1 - \frac{Area(A)}{Area(C_{space})}\right)^2$$

In Toggle PRM, samples can also be generated in A as witnesses to failed connection attempts between configurations in O. To compute the probability of this occurring, we first determine the probability for a particular point $p \in O$. Let $r_{p,\theta}$ denote the ray originating at p in direction θ, $\theta \in [0, 360)$. Let $q_{p,\theta,1}$, $q_{p,\theta,2}$, and $q_{p,\theta,3}$ denote the points, if they exist, where $r_{p,\theta}$ exits O for the first time, next enters O, and next leaves O, respectively. If the segment $(q_{p,\theta,1}, q_{p,\theta,2}) \subseteq A$, then a connection attempt between p and any point on the segment $s_{p,\theta} = (q_{p,\theta,2}, q_{p,\theta,3})$ will fail and yield a witness in A. To determine all points $q \in O$ that would yield a witness configuration in A on a connection attempt to $p \in O$, we can integrate over all θ:

$$f_A(p) = \int_\theta s_{p,\theta} \mathrm{d}\theta$$

Then, we can compute the probability of sampling two points in O such that a connection attempt between them will fail and yield a witness in A as follows:

$$P_{obst}(2) = \frac{\iint\limits_{p \in O} f_A(p) \mathrm{d}x \mathrm{d}y}{Area(C_{space})^2}$$

The above integral represents the total area of O for which $f_A(p)$ is nonzero multiplied by the average usable area from each point in O. We divide by $Area(C_{space})^2$ to find the combined probability over two samples. Thus, the overall probability for Toggle PRM given two samples is as follows:

$$P(2) = P_{uniform}(2) + P_{obst}(2)$$

. We can see that this probability will be greater than $P_{uniform}(2)$ if $f_A(p)$ is not empty for some $p \in O$, which is true if there is indeed a narrow passage in C_{space}. ■

Discussion: From Proposition 1, we can see that the basic idea of simultaneously mapping both C_{free} and C_{obst} has higher probability than uniform random sampling for successfully sampling in a narrow passage. For example, we can compare the probabilities between Toggle PRM and uniform random sampling in a simple 2D case (Figure 2). Say the total C_{space} volume is $100\ units^2$ (10 x 10 units), the narrow passage, A, is $4\ units^2$ (0.4 x 10 units), the surrounding obstacles, B and D, are $48\ units^2$ each (4.8 x 10 units). Given two samples the probability of uniformly sampling within A is 0.078 ($p = 1 - (1 - \frac{Area(A)}{Area(C_{space})})^2$). For Toggle PRM, the sum of the probabilities of sampling two configurations on opposite sides of the obstacles ($\frac{Area(C_{obst}) * AverageUsableArea(C_{obst})}{Area(C_{space})^2}$) is equivalent to $\frac{Area(B+D) * \frac{1}{2} * Area(B+D)}{Area(C_{space})^2} = 0.461$. In total over two samples, Toggle PRM has a probability of 0.54. This corresponds to the formulas described above.

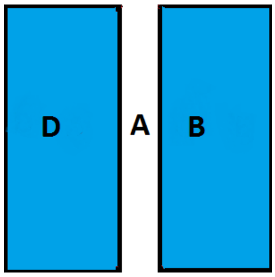


Fig. 2. An environment in which a narrow passage A is bounded by obstacles B and D.

Intuitively, Toggle PRM is less dependent on both the volume of a narrow passage and the volume of obstacles surrounding a narrow passage. From Proposition 1, we know samples in a narrow passage can be obtained by either directly sampling the passage, or by attemping, but failing, to connect samples in obstacles bounding the passage. Similarly, if a narrow volume exists in C_{obst}, then we can use failed connection attempts in C_{free} to obtain a sample in the region. So, Toggle PRM has a compounding effect from sampling and connecting such that if a narrow passage is surrounded by narrow obstacle volume, samples will still be generated within the narrow passage.

IV. Experimental Analysis

In this section we describe experiments conducted to test the effectiveness of Toggle PRM. Section IV-A describes the experimental setup used in the various experiments. Sections IV-B, IV-C, and IV-D compare Toggle PRM to other common sampling-based methods in a variety of narrow passages varying both the narrow passage widths and surrounding obstacle widths for 2 DOF examples. Section IV-E compares the various planners on two 2 DOF motion planning problems solving queries. Lastly, Section IV-F shows the extendability of the method to a 6 DOF motion planning problem.

A. Experimental setup

Toggle PRM, uniform random sampling, Gaussian sampling, bridge-test sampling, and obstacle-based sampling (OBPRM) were implemented using the C++ motion planning library developed by the Parasol Lab at Texas A&M University. RAPID [6] is used for collision detection computations. In these planners, connections are attempted between each node and its k-nearest neighbors according to a distance metric; here we use $k = 5$, C-space Euclidean distance,

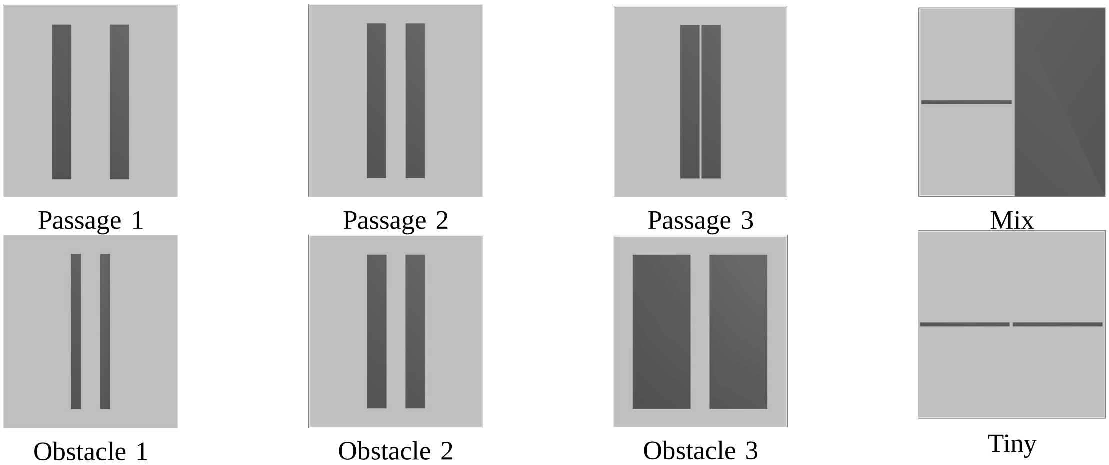


Fig. 3. Simple environments with varying narrow passage and obstacle volumes.

Environment	C_{space}	Narrow Passage	C_{obst}
Passage 1	40000	6400	3200/obst
Passage 2	40000	3200	3200/obst
Passage 3	40000	320	3200/obst
Obstacle 1	40000	3200	1600/obst
Obstacle 2	40000	3200	3200/obst
Obstacle 3	40000	3200	9600/obst
Mix	10000	4	96, 4800
Tiny	10000	4	96/obst

TABLE I
IMPORTANT VOLUMES FOR VARIOUS NARROW PASSAGE EXAMPLES. AMOUNTS GIVEN IN $units^2$.

and a simple straight-line local planner using a bisection evaluation strategy. The bisection evaluation strategy is commonly used because it usually identifies failures faster and hence reduces the amortized cost of failed connection attempts. We chose $k = 5$ to keep the computation cost of connections relatively low. Toggle PRM is implemented as described in Algorithm 1 and connections are attempted for up to k-nearest neighbors, but stop as soon as a witness is returned from one of the attempted connections. Gaussian sampling is implemented as described in [3], with a Gaussian $d_{gauss} = min(NP_{width}, Obst_{width})$, where NP_{width} is the narrow passage width and $Obst_{width}$ is the narrowest width of all surrounding obstacles. d_{gauss} is chosen to find samples which straddle the edges of C_{obst} surrounding the narrow passage. Bridge-test sampling is implemented as described in [7], with $d_{bridge} = NP_{width} + d_{gauss}$ chosen so that samples can span the narrow passage. OBPRM is based on C_{space} validity changes and implemented as in [1].

The methods were evaluated in environments with varying narrow passage types (see Figure 3). All cases are 2 DOF using a near point size robot. The various important C_{space} volumes associated with these environments are shown in Table I. For evaluation and comparison of the various planners, a few metrics are used. Firstly, the number of free nodes sampled in the narrow passage is used to compare how well samplers generate nodes in various types of narrow passages. Secondly, for query evaluations we compare the total number of free nodes needed to solve the query, the number of CD calls (used as a standard metric of efficiency for experiments solving queries), and the percentage of nodes within the narrow passage. A good planner for sampling and solving problems with a narrow passage will ideally minimize the number of free nodes in the roadmap and CD calls while maximizing the percentage of nodes within the narrow passage. All metrics are averaged over 10 runs for all experiments.

B. Sensitivity to narrow passage volume

In this experiment, 1000 samples are attempted with each method (valid or invalid) in three simple environments (Passage 1, Passage 2, and Passage 3, shown in Figure 3) of varying narrow passage volumes surrounded by obstacles of constant volume. Various important C_{space} volumes are shown in Table I. From these areas and their respective proportions of C_{space}, we expect that over the three environments uniform sampling will yield a decreasing percentage of nodes in the narrow passages as the passage volume decreases. Toggle PRM is less dependent on the volume of narrow passages so the number of free nodes generated in the narrow passage should not vary as much as the narrow passage area gets smaller and smaller.

The total number of free nodes generated in the narrow passages is shown in Figure 4. Although the number of free nodes generated by all samplers decreases (as expected) as the narrow passage width decreases, the number of free nodes in the narrow passage for Toggle PRM decreases less dramatically. Bridge-test results are varying because the performance depends on the d_{bridge} chosen. Our selection of d_{bridge} was described in Section IV-A. The percentage of free nodes compared to obstacle nodes for Toggle PRM is shown in Figure 5. From this graph we can see that the percentages in each space are fairly close to each other. Ultimately more nodes remain in the free graph. From these results, Toggle PRM clearly samples more densely within a narrow passage as the volume of that passage decreases than do the other methods.

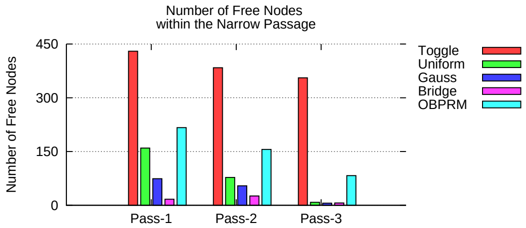


Fig. 4. The number of free nodes sampled in the narrow passages from 1000 attempts in the three environments for Toggle PRM(red), uniform sampling (green), Gaussian sampling (blue), bridge-test sampling (magenta), and obstacle-based sampling (cyan).

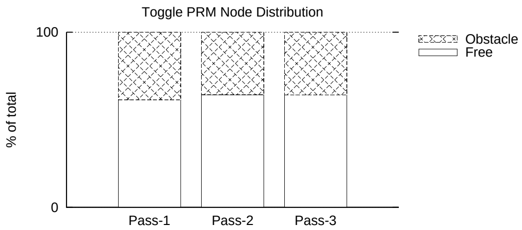


Fig. 5. The percentage of free nodes compared to the percentage of nodes in C_{obst} created by Toggle PRM.

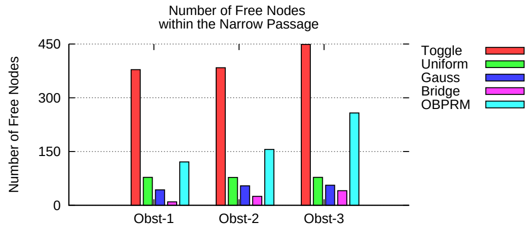


Fig. 6. The number of free nodes sampled in the narrow passages from 1000 attempts in the three environments for Toggle PRM(red), uniform sampling (green), Gaussian sampling (blue), bridge-test sampling (magenta), and obstacle-based sampling (cyan).

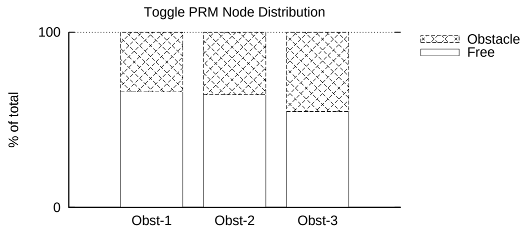


Fig. 7. The percentage of free nodes compared to the percentage of nodes in C_{obst} created by Toggle PRM.

C. Sensitivity to surrounding obstacle volumes

This section shows the sensitivity of the planners to the volume of the obstacles surrounding the narrow passage. In this experiment, 1000 samples are attempted with each method in three simple environments of varying obstacle volumes surrounding a narrow passage of constant volume. The three environments, Obstacle 1, Obstacle 2, and Obstacle 3, are shown in Figure 3. Various important C_{space} volumes are shown in Table I. From these areas and their respective proportions with all of C_{space}, we expect that over the three environments uniform sampling will yield the same percentage of free nodes in the narrow passages, while the other methods should increase the number of free nodes generated in the passage. Since Toggle PRM is less dependent than the other methods on the C_{obst} volume surrounding the narrow passage, the number of free nodes in the narrow passage should not depend as much on it.

The number of free nodes is shown in Figure 6. Uniform random sampling remains constant in all three cases. All other methods increase the number of free nodes generated as the proportion of obstacle space increases. Toggle PRM does not vary as much as the other methods. Additionally we see that Toggle PRM generates the highest number of free nodes given 1000 attempts. Toggle PRM generates a higher number of free nodes on the last case, because most of the uniform nodes sampled are in the obstacles, so connections are attempted between obstacles more often, thus increasing the number of nodes in the narrow passage.

The percentage of nodes in the free map compared to the obstacle map in Toggle PRM is shown in Figure 7. Again, we see the various proportions of nodes in the two maps are similar in all cases. From these results, Toggle PRM again samples more densely within the narrow passage.

D. Other narrow passage types

With the trends seen so far, Toggle PRM is expected to be able to sample in narrow passages where other samplers have trouble, such as a small narrow passage and small bounding obstacle volumes. In this experiment, 1000 samples are attempted with each method in two simple environments with tiny narrow passage volume and varying obstacle volumes. The two environments, Mix and Tiny, are shown in Figure 3. Various important C_{space} volumes are shown in Table I. Since the narrow passage and one or both surrounding obstacles have small area, we expect all the methods to have trouble generating samples in the narrow passage. Since Toggle PRM is less dependent on the volumes of C_{obst} and narrow passages, the number of free nodes in the narrow passage should be significantly higher than the other methods.

The number of free nodes is shown in Figure 8. All but OBPRM and Toggle PRM have difficulty generating a sample within this small narrow passage (0 nodes in most cases). Toggle PRM, however, generates a significantly higher number of nodes in the passage. With a different connection strategy, such as allowing connection attempts to configurations in a node's own connected component of the graph, this number would be even higher. The percentage of nodes in the free map compared to the block map in Toggle PRM is shown in Figure 9. Again, because of the connection strategy given, the connector does not allow connections to a node's own connected component, once a path through the narrow passage is found nodes are generated in the narrow passage at a smaller rate. From these results, Toggle PRM again samples more densely within the narrow passage.

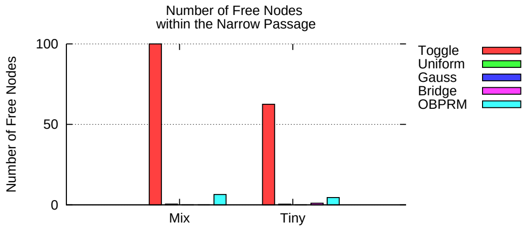


Fig. 8. The number of free nodes sampled in the narrow passages from 1000 attempts in the three environments for Toggle PRM(red), uniform sampling (green), Gaussian sampling (blue), bridge-test sampling (magenta), and obstacle-based sampling (cyan).

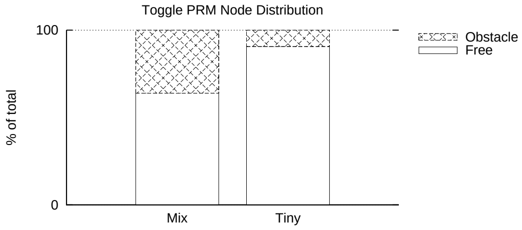
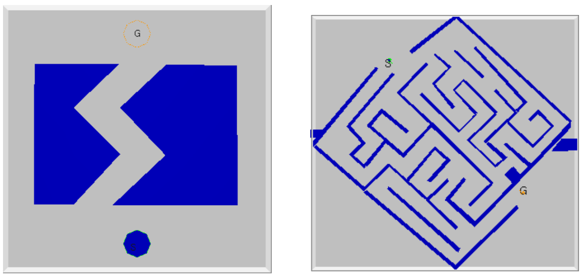


Fig. 9. The percentage of free nodes compared to the percentage of nodes in C_{obst} created by Toggle PRM.



Zig2D Maze2D

Fig. 10. Two simple 2D environments. Query solutions must traverse the narrow passage. Robots are shown in the bottom of each figure, they are approximated circles.

E. Improved problem solving

When solving queries, Toggle PRM is expected to do well on problems in which a query must traverse a narrow passage. For this experiment, the methods are compared against each other in terms of efficient query solving. Toggle PRM is adapted to use a priority queue where valid nodes have priority over invalid nodes to be removed from the queue in order to quickly arrive at a small set of nodes to solve the query. Here, two 2 DOF problems are analyzed. The environments are shown in Figure 10. Planners are run in an incremental manner until the query is solved. In each iteration, 10 nodes are generated and added to the map and then the query is checked. Solutions must traverse the narrow passages. Three metrics are compared: number of free nodes sampled, CD calls as a measure of time, and the approximate percent of free nodes within the narrow passage. Only free nodes are reported for Toggle PRM to compare to the other methods. Data is shown in Table II.

In these experiments, the query was solved in all cases. In both environments, we see that Toggle PRM more efficiently samples the space and thus can reduce the number of CD calls, keeping a high percentage of nodes within the narrow passages. Thus, Toggle PRM is able to solve queries in both scenarios with a significant reduction in CD calls, while requiring a comparably low number of nodes to solve the query.

Planner	Free Nodes	CD	% Free in Passage
Zig2D			
Toggle PRM	**83.4**	**2000**	83
Uniform	435	4026	10
Gaussian	117.5	3017	33
Bridge	93.1	8458	100
OBPRM	297	4918	26
Maze2D			
Toggle PRM	1412	**14360**	98
Uniform	1630	49230	75
Gaussian	1111	41312	94
Bridge	1468	163880	100
OBPRM	**888**	31395	89

TABLE II

RESULTS FOR ZIG2D AND MAZE2D. TOGGLE PRM IS MOST EFFICIENT COMPARED TO THE OTHERS.

From these experiments, the utility of Toggle PRM for difficult narrow passages is seen. The Toggle method, i.e., mapping both C_{free} and C_{obst}, is more efficient in generating nodes in narrow passages. Another benefit of the method is that after nodes are generated in the narrow passage the roadmap gets expanded from there. This expansion as seen by the results occurs quickly.

F. Higher dimensions

In this section, an experiment using the same general Toggle PRM method is applied to a rigid body robot with 6 DOF to show the general extendability of the algorithm to a higher DOF problem. As with the 2 DOF queries, Toggle PRM is adapted to use a priority queue where valid nodes have priority over invalid nodes to be removed from the queue. This experiment is evaluated by efficiency in solving a query. The environment is shown in Figure 11. The query must traverse the maze to solve the query. The percentage of nodes within the narrow passage is approximated by using bounds of the workspace obstacle, as the 6-dimensional C_{space} volume would be difficult to compute. Again, only free nodes are reported for Toggle PRM to compare to the other methods. Results are shown in Table III.

As with the 2D experiments, Toggle PRM outperforms the other methods in efficiency. Toggle PRM had significantly fewer CD calls than all other methods and kept the approximate percent of nodes in the narrow passage fairly high (49.3%). We see that other methods are less successful in sampling (causing a higher number of CD calls) even though the actual samples they create might be placed appropriately to solve the problem.

V. CONCLUSION

In this paper, we present a new PRM approach which simultaneously maps C_{free} and C_{obst}. The new method,

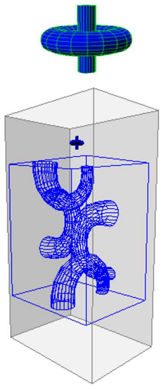


Fig. 11. Maze environment. Query solution must traverse the narrow passage from top to bottom.

Maze			
Planner	Free Nodes	CD	% Free in Passage
Toggle PRM	**456**	**7.34e4**	49.3
Uniform	2562	94.9e4	20.8
Gaussian	474	1.75e5	63.8
Bridge	620	1.20e6	64.3
OBPRM	749	5.49e5	63.4

TABLE III

RESULTS FROM THE MAZE PROBLEM. PERCENTAGE OF FREE NODES IN NARROW PASSAGE IS APPROXIMATED BY WORKSPACE BOUNDS OF THE NARROW PASSAGE.

Toggle PRM, increases the efficiency and density of sampling within narrow passages over previous PRM methods. Toggle PRM does this by retaining witness nodes from failed local planning attempts. The main advantage in doing this is that when a connection fails in one space, it provides important information about the other space. Thus, the witness configuration can be saved and placed in the opposite space's map. Through both theoretical analysis and experimental analysis in 2 DOF cases, the benefit of Toggle PRM is shown to increase the sampling density in a narrow passage. Additionally, it was shown that Toggle PRM is less dependent on the volumes of the narrow passage and obstacles surrounding the passage, unlike uniform sampling.

In the future, we plan to explore the applicability of Toggle PRM in higher dimensions. Additionally, optimizations can be made for both the connections attempted and nodes saved from failed connection attempts.

REFERENCES

[1] N. M. Amato, O. B. Bayazit, L. K. Dale, C. V. Jones, and D. Vallejo. OBPRM: An obstacle-based PRM for 3D workspaces. In *Robotics: The Algorithmic Perspective*, pages 155–168, Natick, MA, 1998. A.K. Peters. Proc. Third Workshop on Algorithmic Foundations of Robotics (WAFR), Houston, TX, 1998.

[2] R. Bohlin and L. E. Kavraki. Path planning using Lazy PRM. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pages 521–528, 2000.

[3] V. Boor, M. H. Overmars, and A. F. van der Stappen. The Gaussian sampling strategy for probabilistic roadmap planners. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, volume 2, pages 1018–1023, May 1999.

[4] B. Burns and O. Brock. Sampling-based motion planning using predictive models. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pages 3313–3318, 2005.

[5] B. Burns and O. Brock. Toward optimal configuration space sampling. In *Proc. Robotics: Sci. Sys. (RSS)*, pages 105–112, 2005.

[6] S. Gottschalk, M. C. Lin, and D. Manocha. OBB-tree: A hierarchical structure for rapid interference detection. *Comput. Graph.*, 30:171–180, 1996. Proc. SIGGRAPH '96.

[7] D. Hsu, T. Jiang, J. Reif, and Z. Sun. Bridge test for sampling narrow passages with probabilistic roadmap planners. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pages 4420–4426, 2003.

[8] D. Hsu, L. E. Kavraki, J.-C. Latombe, R. Motwani, and S. Sorkin. On finding narrow passages with probabilistic roadmap planners. In *Proc. Int. Workshop on Algorithmic Foundations of Robotics (WAFR)*, pages 141–153, 1998.

[9] D. Hsu, G. Sánchez-Ante, and Z. Sun. Hybrid PRM sampling with a cost-sensitive adaptive strategy. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pages 3885–3891, 2005.

[10] L. E. Kavraki, P. Švestka, J. C. Latombe, and M. H. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Trans. Robot. Automat.*, 12(4):566–580, August 1996.

[11] J. J. Kuffner and S. M. LaValle. RRT-Connect: An Efficient Approach to Single-Query Path Planning. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pages 995–1001, 2000.

[12] S. M. LaValle and J. J. Kuffner. Randomized kinodynamic planning. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pages 473–479, 1999.

[13] J.-M. Lien, O. B. Bayazit, R.-T. Sowell, S. Rodriguez, and N. M. Amato. Shepherding behaviors. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pages 4159–4164, April 2004.

[14] J.-M. Lien, S. L. Thomas, and N. M. Amato. A general framework for sampling on the medial axis of the free space. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pages 4439–4444, September 2003.

[15] T. Lozano-Pérez and M. A. Wesley. An algorithm for planning collision-free paths among polyhedral obstacles. *Communications of the ACM*, 22(10):560–570, October 1979.

[16] M. Morales, L. Tapia, R. Pearce, S. Rodriguez, and N. M. Amato. A machine learning approach for feature-sensitive motion planning. In *Algorithmic Foundations of Robotics VI*, pages 361–376. Springer, Berlin/Heidelberg, 2005. book contains the proceedings of the International Workshop on the Algorithmic Foundations of Robotics (WAFR), Utrecht/Zeist, The Netherlands, 2004.

[17] M. A. Morales A., R. Pearce, and N. M. Amato. Metrics for analyzing the evolution of C-Space models. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pages 1268–1273, May 2006.

[18] M. A. Morales A., L. Tapia, R. Pearce, S. Rodriguez, and N. M. Amato. C-space subdivision and integration in feature-sensitive motion planning. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pages 3114–3119, April 2005.

[19] C. L. Nielsen and L. E. Kavraki. A two level fuzzy PRM for manipulation planning. Technical Report TR2000-365, Computer Science, Rice University, Houston, TX, 2000.

[20] J. H. Reif. Complexity of the mover's problem and generalizations. In *Proc. IEEE Symp. Foundations of Computer Science (FOCS)*, pages 421–427, San Juan, Puerto Rico, October 1979.

[21] S. Rodriguez, S. Thomas, R. Pearce, and N. M. Amato. (RESAMPL): A region-sensitive adaptive motion planner. In *Algorithmic Foundation of Robotics VII*, pages 285–300. Springer, Berlin/Heidelberg, 2008. book contains the proceedings of the International Workshop on the Algorithmic Foundations of Robotics (WAFR), New York City, 2006.

[22] A. P. Singh, J.-C. Latombe, and D. L. Brutlag. A motion planning approach to flexible ligand binding. In *Int. Conf. on Intelligent Systems for Molecular Biology (ISMB)*, pages 252–261, 1999.

[23] S. A. Wilmarth, N. M. Amato, and P. F. Stiller. MAPRM: A probabilistic roadmap planner with sampling on the medial axis of the free space. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, volume 2, pages 1024–1031, 1999.

[24] S. W. H. Wong and M. Jenkin. Exploiting collision information in probabilistic roadmap planning. In *Int. Conf. on Mechatronics*, 2009.