# Teaching a Randomized Planner to Plan with Semantic Fields

Ian Baldwin
Department of Engineering Science
Oxford University
Oxford, UK
Email: ib@robots.ox.ac.uk

Paul Newman
Department of Engineering Science
Oxford University
Oxford, UK
Email: pnewman@robots.ox.ac.uk

***Abstract*—This paper presents a novel way to bias the sampling domain of stochastic planners by learning from example plans. We learn a generative model of a planner as a function of proximity to labeled objects in the workspace. Our motivation is that certain objects in the workspace have a local influence on planning strategies, which is dependent not only on where they are but also on *what* they are. We introduce the concept of a Semantic Field — a region of space in which configuration sampling is modelled as a multinomial distribution described by an underlying Dirichlet distribution. We show how the field can be trained using example expert plans, pruned according to information content and inserted into a regular RRT to produce efficient plans. We go on to show that our formulation can be extended to bias the planner into producing sequences of samples which mimic the training data.**

## I. Introduction

Stochastic motion planning is an established technique in robotics that enables motion planning in complex workspaces, or for systems that have high-dimensional configuration spaces. **R**apidly-exploring **R**andom **T**rees (**RRT**'s [1]) and **P**robabilistic **R**oad **M**aps (**PRM**'s [2]) are both instantiations of stochastic planners. Both RRT's and PRM's utilise a graphical representation of the workspace, which captures the relationship between free and occluded space. RRT's are categorised as "one-shot" planners - the planning system has to be re-initalised for each new plan or "query". Conversely PRM's, once instantiated, are persistent throughout the task and can be re-queried.

RRTs have been effective for a number of different motion planning tasks, including manipulation and grasping [3], kinodynamic planning[4] and for other non-holonomic systems [5]. With the introduction of probabilistic planners, a substantial body of research has been directed at manipulating the sampling domain to overcome this issue. Uniform sampling of the configuration space suffers from the so-called "narrow passage problem" - more samples are generated in the free space than in the more complex regions, which is typically where more information is required.

Some of the earliest strategies for biasing the sampling distribution in the workspace include the Narrow Passage Sampling strategy [6] and the Gaussian Strategy for PRM's [7], amongst others. Both methods utilise a pre-determined criteria to determine whether samples generated in the configuration or workspace are considered for planning. $\mathcal{C}$-Space sampling strategies include Approximated Medial Axis sampling [8](which has a workspace counterpart), in addition to strategies which explore the $\mathcal{C}$-space boundaries[9]. Workspace-based methods include Dynamic Domain RRTs which adaptively bias the sampling domain[10], cell-based decomposition methods which determine which regions require more attentive sampling [11] [12] and model-based methods which actively seek to sample in unknown or poorly sampled regions [13]. Recent techniques have utilized features in the environment to better determine how to bias the sampling distribution - Adaptive Workspace Biasing [14] generates an optimized weighting of feature vectors defined over a voxelised workspace. Various hybrid methods incorporating multiple different sampling have also been proposed [15].

The majority of planners use either a pre-defined sampling criterion to evaluate the samples generated, or adapt the sampler to features present in the workspace. For terrestrial robotics, there exists a large amount of logged data on how a user operates a vehicle under remote control. It is possible then to consider the operator as an expert, and try to emulate the behaviour of the vehicle in certain environments, whilst still maintaining the exploratory nature of the probabilistic approach. This paper is primarily concerned with generating a sampling process for RRT's, although the framework should extend naturally to PRM's.

### *A. Semantic Fields*

In any path-planning task, there will be certain regions in the configuration or workspace that are more beneficial to sample from regularly. Typically the goal of the planner is considered to be one of these regions - biasing the sample generator of the planner a certain amount towards the target results in improved performance. We begin with the assumption that a human operator uses semantic knowledge to influence their planning strategies near identifiable objects. A few everyday examples are illustrative: people tend to approach doors head-on, avoid the tops of stairwells, and keep clear of moving objects. This sufficiently motivates us to introduce the notion of "semantic fields" - regions of the workspace in which sampling is influenced both by the geometry of an object and *what* the object is. The task is therefore to learn what form this influence takes from such examples and to incorporate that effect into a probabilistic planner.

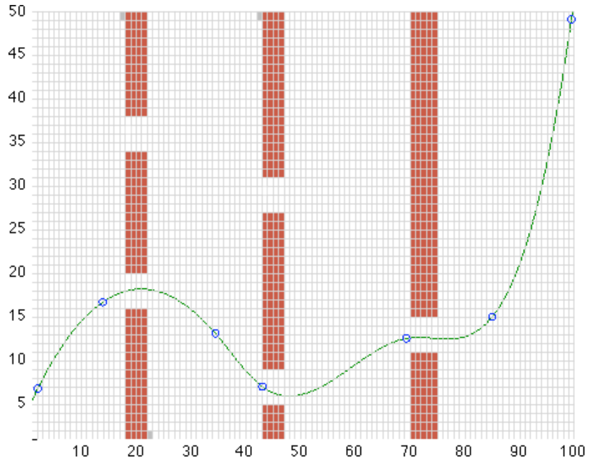


Fig. 1. Illustraton of a $100 \times 50$ cell grid-world. Darker areas correspond to blocked regions. An expert trajectory through the environment is shown in green. The goal is to generate a collision-free path from the lower-left to the upper-right.

Continuing with the simple example of motion through a doorway, consider Figure 1 which shows the test grid-world with an "expert path" generated by a human operator. The goal is to manoeuvre a holonomic point robot from the start position, located at the lower left to the goal position at the top right. Here the environment has been discretised into a 100x50 grid of cells, with darker areas indicating blocked regions. Situated at the centre of each opening in the environment is a semantic field, with an influence over a neighbouring group of cells. The expert is then modeled as a stochastic planner who chooses samples within this region. The semantic field can be considered to be a random variable with $k$ states, where $k$ is the number of cells in the field. A set of N exemplar paths are then observed that pass through the field. Each path is discretized and each element is considered to be an independent observation, corresponding to a specific cell in the grid, and therefore a state of the random variable. The distribution that describes the counts of these observations over the grid is the Multinomial distribution:

$$
\begin{aligned}
Mult(s_1, s_2 \ldots \mid \mu, N) &= \frac{N!}{s_1! \cdots s_k!} \mu_1^{s_1} \cdots \mu_k^{s_k} \\
&= \frac{N}{s_1 \ldots s_k} \prod_{k=1}^{k} \mu_k^{s_k}
\end{aligned} \quad (1)
$$

This Multinomial describes the counts over the semantic field cells $\mathcal{S} = \{s_1 \ldots s_k\}$ for N discrete observations with cell probabilities $\boldsymbol{\mu} = \{\mu_1 \ldots \mu_k\}$ [16]. The goal of the planner during the learning phase is to approximate the multinomial distribution that best describes the collection of exemplar paths shown during the training phase We denote these data as $\mathcal{Z} = \{z_1 \ldots z_k\}$, where $z_k$ corresponds to the number of observations of state $k$. From a Bayesian standpoint, Equation 1 can be considered to be a likelihood term. The conjugate prior of the Multinomial distribution is the Dirichlet, defined as:

$$
Dir(\mu_1, \ldots, \mu_K; \alpha_1, \ldots, \alpha_K) = \frac{1}{B(\boldsymbol{\alpha})} \prod_{k=1}^{K} \mu_k^{\alpha_k - 1} \quad (2)
$$

Where $\boldsymbol{\alpha} = \{\alpha_1 \ldots \alpha_K\}$ are the "shape" parameters, which can be interpreted as "frequency counts" of observed variables, and $B(\boldsymbol{\alpha})$ is a normalisation term. A natural question that arises is the initial choice of alpha, or the "prior" over the Dirichlet parameters. Using equal initial $\boldsymbol{\alpha}$ values of 1 equates to a uniform distribution over parameters. To obtain the posterior distribution:

$$
p(\boldsymbol{\mu} \mid \mathcal{Z}, \boldsymbol{\alpha}) \quad (3)
$$

where $\mathcal{Z}$ is the set of cell counts, the prior distribution (2) is multiplied by our likelihood (1) so that:

$$
\begin{aligned}
p(\boldsymbol{\mu} \mid \mathcal{Z}, \boldsymbol{\alpha}) &\propto p(\mathcal{Z} \mid \boldsymbol{\mu}, N) p(\boldsymbol{\mu} \mid \boldsymbol{\alpha}) \\
&\propto \prod_{k=1}^{K} \mu_k^{\alpha_k + \mathcal{Z}_k - 1}
\end{aligned} \quad (4)
$$

where $\mathcal{Z}_k$ corresponds to the number of observations of state $k$ in the training data. The Bayesian update corresponds to:

$$
p(\boldsymbol{\mu} \mid \mathcal{Z}, \boldsymbol{\alpha}) = Dir(\boldsymbol{\mu} \mid \boldsymbol{\alpha} + \mathcal{Z}) \quad (5)
$$

As such, the multinomial distribution describing the underlying semantic field can be updated in a Bayesian fashion by observing and incrementing the visitations of the expert path to each underlying cell. In order to generate samples from this model, it is necessary to parameterize the Multinomial by a set of $\boldsymbol{\mu}$ values drawn from this posterior. Sampling from the Dirichlet posterior (as parameterised by the $\boldsymbol{\alpha}$ values), involves collating $K$ independant random samples are drawn from unit-scale Gamma distributions:

$$
x_k \sim \Gamma(\alpha_k, 1) \quad (6)
$$

The resultant sample $\boldsymbol{x} = (x_1 \ldots x_i)$ is defined as:

$$
x_i = \frac{x_k}{\sum_{k=1}^{K} x_k} \quad (7)
$$

which is one sample from the posterior distribution. Utilising the Dirichlet/Multinomial distributions in this manner is a technique well known in text classification[17].

Figure 2 shows two exemplar paths through different "doorway" configurations. Figure 3 shows the evolution of samples from the posterior distribution during the learning process for each of the openings shown in Figure 2. Each sample from the posterior $p(\boldsymbol{\mu} \mid \mathcal{Z}, \boldsymbol{\alpha})$ is a distribution that parameterises the Multinomial over the semantic field, and can be considered as one member of a family of such distributions.

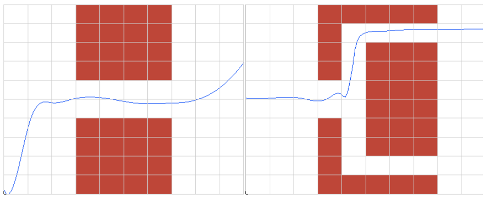

Fig. 2. Training routes through different obstacle types: a) Uniform opening (left), and (b) a more complex obstacle type (right). Training data was obtained by manually generating sets of such paths.

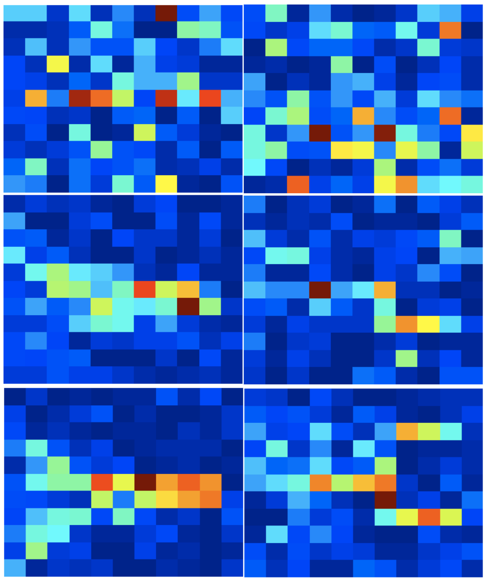

Fig. 3. These are samples drawn from a Dirichlet distribution $p(\boldsymbol{\mu} \mid \mathcal{Z}, \boldsymbol{\alpha})$ as $N$ increases (i.e. more training examples are observed). Blue areas correspond to low probability, red areas to high probability. (Best viewed in color.)

Samples in the workspace are generated by parameterising a Multinomial distribution with the probabilities from the Dirichlet sample, and then sampling uniformly from the probability mass function of the Multinomial. Figure 4 depicts a set of 1000 samples generated from the multinomial distribution described by the sampled posterior. This technique is somewhat similar to the technique used in [18], where the authors use a non-homogeneous Poisson process to perform tracking of people in a lab. The resulting *spatial affordance map* bears some resemblance to the semantic field, although in the planner case we do not require the underlying distribution to be time-varying.

### B. Sequence Generation

Sampling from a static distribution over the semantic fields will lead to better planner performance through complex regions, although it does not replicate the natural paths that were observed during the learning phase. In order to construct a sampling strategy that would generate paths similar to those of an expert, a spatial consistency over paths must be enforced. To model the state-to-state transitions of the learned paths, the cells in the semantic fields were viewed as a 4-connected grid. From the learned data, the transition probability describing the

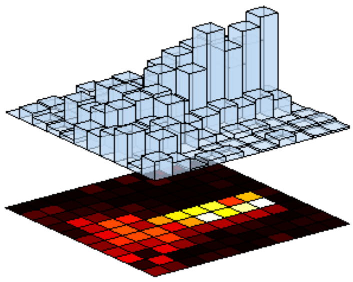

Fig. 4. 1000 samples from a Multinomial distribution (top) as parameterised by a sample from the posterior distribution(below). Regions of high probability in the Dirichlet sample (white) correspond to increased cell counts.

transition from state $s$ to state $s'$ is:

$$P(x_{s'} \mid x_s) = \frac{T(x_s, x_{s'})}{T(x_s, x^*)} \quad \textbf{iff} \quad x_{s'} \in \mathcal{N}(x_s) \qquad (8)$$

Where $T(x, y)$ defines the number of times that a state transition was observed from state $x$ to $y$, $x^*$ denotes every neighboring state and $\mathcal{N}(.)$ defined the set of neighbours of a state. In this way, we learn to bias the sampling towards sequential paths - however, this can be problematic; if the training data are insufficient, the resultant transition matrix would have zero entries corresponding to state transitions that were never observed. To obtain a less extreme prediction of state transition probabilities, we assume a Dirichlet prior over each transition probability:

$$Dir(\mu_1, \ldots, \mu_K; \alpha_1, \ldots, \alpha_K) = \frac{1}{B(\boldsymbol{\alpha})} \prod_{k=1}^{K} \mu_k^{\alpha_k - 1} \qquad (9)$$

Where $\boldsymbol{\alpha}$ in this definition corresponds to the observation of state transitions. Figure 5 depicts the state-neighbour transition probabilities for each of the 121 states in the grid to each of its 4 neighbours:

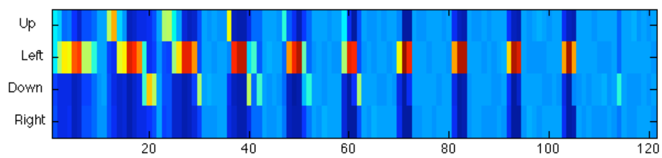


Fig. 5. Probability matrix for all 121 states in the semantic field. As the field is 4-connected, there are 4 possible transitions for each state, actions are (from left to right): Up, down, left and right. Dark blue areas correspond to low probability, red areas correspond to high probability.

A high probability for a state-neighbour pair suggests that the posterior distribution is expressing a preference for transiting to a specific neighbour from a specific state. It can be see from Figure 5 that the lower indexed states express high preference for their neighbours, while those in later states do not. This is a result of the training data collected over the field illustrated in Figure 2(a). There tends to be more heterogeneity in the initial states, due to the fact that there are numerous directions from which one can approach a doorway, transitioning from the left to the right. As the tracks progress through the field, they all tend to behave in a similar way. This can be seen in Figure 3(lower left) where there tends to be a low probability for states that do not occur on the main thoroughfare, and therefore the state transition matrix in this region tends to be more homogeneous as no (or very few) state-transitions were observed. To pose this in a sampling framework, if a node of the RRT is under the influence of a semantic field and is selected by the planning algorithm to undergo the *extend* step, then another sample is generated *within* the field and this is done in proportion to the probabilities of the neighbours. (See Algorithm 1)

### C. Field pruning

During the learning phase, the semantic fields were bounded in a predetermined way (for the example grid-world, this was an $11 \times 11$ grid). This may not be optimal during planning, as the planner may sample in a region of the semantic field which is uninformative, meaning the field may express a uniform action for each possible state, in which case there is little to be gained from utilising the bias of the field. Therefore there is a need to prune the semantic field, in order to definitively specify the region in which there is expected to be a measurable gain from utilising the learned bias. One way of doing this is to analyse the information content of the field, specifically by evaluating the **M**utual **I**nformation (**MI**) between adjacent states in the grid:

$$MI(X;Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) log \frac{p(x,y)}{p(x)p(y)} \quad (10)$$

Using the MI as a guiding criterion, it is possible to parse the grid structure and remove states whose information relative to its neighbors is below a certain threshold. Shown in Figure 6 is the resultant map obtained from evaluating the pairwise MI for states on a semantic field similar to that of the right-most illustration in Figure 2. For this particular region, the field was shown more paths transiting through the upper route than the lower - hence the posterior's preference for the top route.

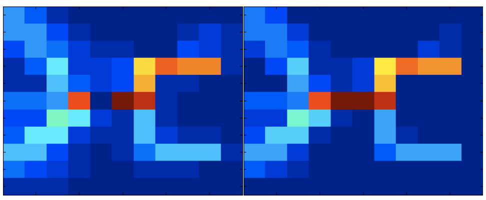

Fig. 6. Posterior distribution as observed through a doorway with two exits. On the left is the generated distribution, on the right is the distribution after it has been pruned by using the Mutual Information content between cells as a criterion.

Figure 6 also shows the semantic field before trimming (darker blue areas corresponding to low probability, colours tending to the red spectrum denote high probability). Note that low-probability areas around the peripheral of the field have been discarded, corresponding to the low information content of these states relative to their neighbours.

### D. Integration of Semantic Fields and RRTs

Because the semantic fields are defined over specific regions in the state space, a uniform distribution over the workspace was incorporated to maintain the "exploratory" nature of the planner. As such, there is a weighting between the uniform distribution, and the *pdf* resulting from the semantic fields. This parameter, termed $\beta$, was required to be estimated (see Results section) and constitutes the only free parameter of the model. Once the optimal weighting has been determined, the overall planning algorithm for the static environment can be summarized as follows:

**Algorithm 1** RRT with Semantic Field bias

**procedure** SEMANTICFIELDRRT($\beta$)
  $\mathcal{T}.initialiseRRT()$
  **while** $counter < maxSamples$ **do**
    $s \leftarrow GenerateRandomSample(\beta)$
    $n \leftarrow \mathcal{T}.RetrieveClosestNode()$
    **if** $NodeIsInSemanticField(n)$ **then**
      $s \leftarrow GenerateFieldSample$ ▷ (s replaced)
    **end if**
    $p \leftarrow \mathcal{T}.Extend(s)$ ▷ (new state)
    **if** $NoCollision(p)$ **then** ▷ (collision checker)
      $\mathcal{T}.addNewNode(p)$
    **end if**
    **if** $GoalReached(p)$ **then**
      $status \leftarrow success$
      $return(status, \mathcal{T})$
    **end if**
  **end while**
  $status \leftarrow failure$
  $return(status)$
**end procedure**

This represents a modified version of the traditional RRT algorithm, incorporating a two-stage sampling strategy: A sample is generated in the workspace according to the weighting $\beta$, and the nearest neighbour (measured in terms of the Euclidean distance) is selected - if that node is under the influence of a semantic field, the sample is over-ridden with a sample generated from the field. This strategy ensures the frontier nodes are still selected in proportion to the Voronoi region they occupy, while simultaneously generating local samples in the semantic field that are more informative for the planner.

## II. RESULTS

Shown in Figure 7 is the result of a static planning query with a standard RRT in the grid-world, alongside a semantic-field based query.

The starting state for this system is at the lower left, with the target located at the upper right. Each planner query was allowed a fixed number of planning cycles before it was

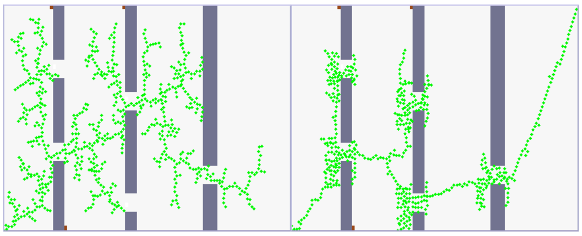

Fig. 7. Behaviour of an RRT in a test grid-world under uniform sampling (left) and with semantic field-based sampling (right)

terminated. Invoking the planner with a uniform sampling set, the RRT behaves as expected - it is able to transition through most of the grid world, although ultimately fails to reach the goal in the allotted time. This is contrasted against the semantic field planner (Figure 7, right) which is able to solve the planning task in the allocated time. In both cases, the RRT had a built-in bias towards the goal - this technique is common practice in order to ensure some measure of convergence towards the target. This sample result shows that the biased planner is able to transition through the environment and reach the goal in less planning cycles than the uniform planner. This is expected as the training data provided prior information that "doorways" were more appealing features in the environment to sample near.

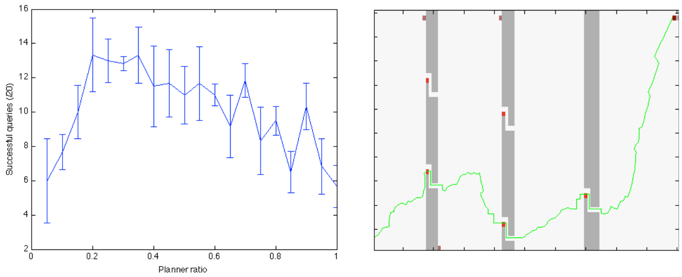


Fig. 8. Results of varying the ratio between uniform and "semantic field" strategies for the grid-world shown. The ratio parameter is defined as $\beta$ where the probabilities of sampling from each scheme are: $p(uniform) = \beta$ and $p(semanticfield) = (1 - \beta)$. The result of varying $\beta$ is plotted against the number of successful planning queries, out of 20 instantiations.

Figure 8 shows the effect of varying the ratio between the uniform *pdf* and the semantic field *pdf*. The plotted data depicts 2400 planning queries over the adjacent grid-world. Purely uniform sampling with a standard RRT was typically unable to solve any planning queries for this scenario, and as such was left out. The relationship between the two components of the sampling scheme (uniform and semantic-field based) shows that extreme weightings of either strategy leads to a low proportion of successful queries, and that a preferable weighting occurs in the $\beta = 0.2 \rightarrow 0.4$ range. To show that paths generated by a planner under the influence of a semantic field are more like the training data than the conventional RRT, a similarity measure was calculated. The nodes generated by the planner within the field are considered to be observations. For a given path-set of observations $N = \{n_1 \dots n_T\}$, and their corresponding probabilities $\nu = \{\nu_1 \dots \nu_T\}$ under the posterior as defined by Equation 5, the similarity measure $\mathbb{S}$ is given by:

$$\mathbb{S}_N = \sum_{t=1}^{T} log(\nu_t) \tag{11}$$

Figure 9 (a) shows the posterior distribution under a set of fabricated paths. The intent of this set is to generate planner behaviour that traverses through a semantic field in a very specific way - in this case following a bend. This particular field was defined over the goal region, and as such all the paths terminate at the centre of the field, which corresponded to the goal state.

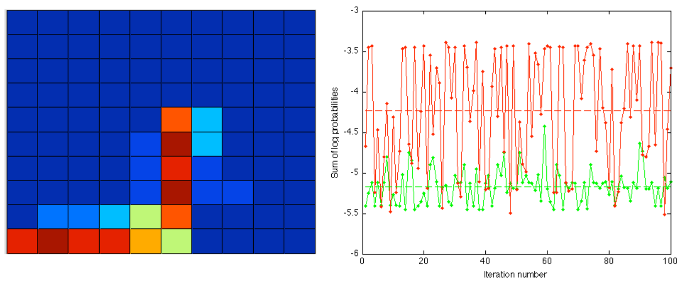


Fig. 9. Data showing the similarity measure between the training data and paths generated using both the semantic-field based approach, and a conventional RRT. (a) Shown is the posterior distribution over the training data (left), as well as (b) the similarity measures of 100 paths generated by both random and semantic-field based RRTs as compared to the training data (right).

Figure 9(b) illustrates the similarity of the training data to paths generated by a standard RRT (highlighted in green) to the training data, as compared to those generated by the semantic-field based planner (highlighted in red). As can be seen in the figure the average similarity of the semantic field RRT (plotted in dashed-red) is higher than that of the random RRT (dashed-green).

## III. Conclusion

The introduction of semantic fields provides a way to learn from expert demonstration as to how features in the environment affect planning strategies. By defining a semantic field over a region in space and observing the behavior of a system under human control, it is possible to update a sampling distribution over the field in a Bayesian fashion. Utilizing this biased sampling strategy during planning shows that the planner is capable of solving queries that a conventional RRT is unable to. This technique can be generalized to more complicated environments, or incorporated into other planner types (for example $\mathcal{C}$-space planners). This method will generalize to handling dynamic obstacles in the planner environment if there are sufficient data to form a suitably confident distribution from which to sample. Further work includes incorporating non-parametric methods into the sampler (for example, Gaussian Processes) to better approximate the distribution over continuous workspaces.

## IV. Acknowledgements

This work was funded by the Office of Naval Research (ONR) Grant N00140810337.

## References

[1] S. LaValle, "Rapidly-exploring random trees: A new tool for path planning," TR 98-11, Computer Science Dept., Iowa State University, Tech. Rep., 1998.

[2] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," in *IEEE International Conference on Methods and Models in Automation and Robotics, 2005*. MorganKaufmann, 1997, pp. 566–580.

[3] M. Kallmann, A. Aubel, T. Abaci, and D. Thalmann, "Planning collision-free reaching motions for interactive object manipulation and grasping," *Eurographics*, vol. 22, pp. 313–322, 2003.

[4] J. Cortes and T. Simeon, "Sampling-based motion planning under kinematic loop-closure constraints," in *In 6th International Workshop on Algorithmic Foundations of Robotics*. Springer-Verlag, 2004, pp. 59–74.

[5] J. Kim and J. Ostrowski, "Motion planning a aerial robot using rapidly-exploring random trees with dynamic constraints," in *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, vol. 2, Sept. 2003, pp. 2200–2205 vol.2.

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

[7] V. Boor, M. H. Overmars, and A. F. van der Stappen, "The gaussian sampling strategy for probabilistic roadmap planners," in *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, vol. 2, 1999, pp. 1018–1023 vol.2. [Online]. Available: http://dx.doi.org/10.1109/ROBOT.1999.772447

[8] Y. Yang and O. Brock, "Adapting the sampling distribution in PRM planners based on an approximated medial axis," in *IEEE International Conference on Robotics and Automation*, vol. 5. Citeseer, 2004, pp. 4405–4410.

[9] N. Amato, O. Bayazit, L. Dale, C. Jones, and D. Vallejo, "OBPRM: An obstacle-based PRM for 3D workspaces," in *Robotics: The Algorithmic Perspective: 1998 Workshop on the Algorithmic Foundations of Robotics*, 1998, pp. 155–168.

[10] A. Yershova, L. Jaillet, T. Simon, and S. M. LaValle, "Dynamic-domain rrts: Efficient exploration by controlling the sampling domain," in *Proceedings IEEE International Conference On Robotics and Automation*, 2005, pp. 3867–3872.

[11] J. van den Berg and M. Overmars, "Using workspace information as a guide to non-uniform sampling in probabilistic roadmap planners," *The International Journal of Robotics Research*, vol. 24, no. 12, p. 1055, 2005.

[12] T. Simeon, J. Laumond, and C. Nissoux, "Visibility-based probabilistic roadmaps for motion planning," *Advanced Robotics*, vol. 14, no. 6, pp. 477–493, 2000.

[13] B. Burns and O. Brock, "Sampling-based motion planning using predictive models," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation, 2005. ICRA 2005*, 2005, pp. 3120–3125.

[14] M. Zucker, J. Kuffner, and J. Bagnell, "Adaptive workspace biasing for sampling-based planners," in *IEEE International Conference on Robotics and Automation. New York: IEEE Press*. Citeseer, 2008, pp. 3757–3762.

[15] S. L. Thomas, M. Morales, X. Tang, and N. M. Amato, "Biasing samplers to improve motion planning performance," in *ICRA*, 2007, pp. 1625–1630.

[16] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, August 2006.

[17] R. E. Madsen, D. Kauchak, and C. Elkan, "Modeling word burstiness using the dirichlet distribution," in *In Proceedings of the 22nd International Conference on Machine Learning*, 2005, pp. 545–552.

[18] M. Luber, G. Tipaldi, and K. Arras, "Spatially grounded multi-hypothesis tracking of people," in *Workshop on People Detection and Tracking, 2009 IEEE International Conference on Robotics and Automation (ICRA), Kobe, Japan*, 2009.