Relevant Region Exploration On General Cost-maps For Sampling-Based Motion Planning

Sagar Suhas Joshi$^1$ and Panagiotis Tsiotras$^2$

Abstract
Asymptotically optimal sampling-based planners require an intelligent exploration strategy to accelerate convergence. After an initial solution is found, a necessary condition for improvement is to generate new samples in the so-called “Informed Set”. However, Informed Sampling can be ineffective in focusing search if the chosen heuristic fails to provide a good estimate of the solution cost. This work proposes an algorithm to sample the “Relevant Region” instead, which is a subset of the Informed Set. The Relevant Region utilizes cost-to-come information from the planner’s tree structure, reduces dependence on the heuristic, and further focuses the search. Benchmarking tests in uniform and general cost-space settings demonstrate the efficacy of Relevant Region sampling.

1 INTRODUCTION
In recent years, sampling-based motion planning (SBMP) algorithms have gained popularity due to their ability to handle high dimensional search spaces and kino-dynamic constraints. Deterministic search methods such as A* do not scale well owing to the computational cost associated with the a priori discretization of the search space. Incremental SBMP algorithms such as RRT$^1$, on the other hand, avoid this computational overhead by generating random samples to build a connectivity tree online. However, the RRT algorithm only guarantees probabilistic completeness. Complementing the exploration module of RRTs with an exploitation module results in asymptotic optimality for these randomized methods i.e., the probability of finding the optimal solution approaches unity as the number of samples approach infinity. While the exploration module generates new samples and extends the connectivity tree, the exploitation module processes this tree to improve the current solution. The popular RRT$^*$ algorithm$^2$ implements a “local rewiring” procedure for exploitation, while RRT$^#$$^3$ utilizes dynamic programming to implement a “global rewiring” procedure. RRT$^#$ ensures optimal connection for each vertex in the current graph at the end of every iteration. The DRRT algorithm$^4$ uses gradient descent to optimize the location of samples in the tree structure. The BIT$^*$ algorithm$^5$ creates batches of samples and prioritizes search according to the potential solution
The FMT* algorithm \cite{6} expands the search outwards in the cost-to-arrive space and uses “lazy” dynamic programming to reduce the number of collision checks.

Conventionally, SBMP algorithms have employed a uniform random exploration strategy. This results in an implicit Voronoi bias, leading to a rapid exploration of the search space. However, in order to achieve a more focused exploration, several improvements to the uniform sampling strategy have been suggested. These include local biasing \cite{7}, use of a heuristic-based quality measure to guide exploration \cite{8}, application of an information-theoretic framework to steer exploration \cite{9,10} and translating ideas from A* to the continuous domain \cite{11}.

Gammell et al. \cite{12} used an admissible heuristic and the current (sub-optimal) solution cost to define the Informed Set. This set has 100 \% recall, i.e., it includes all points that can potentially improve the current solution. Exploration outside the Informed Set is thus redundant. For minimum-length planning problems in Euclidean spaces, the authors of \cite{12} proposed an efficient method to generate samples in the $L_2$-Informed Set: a prolate hyperspheroid with focal points at the start and goal locations and transverse diameter equal to the current best solution cost. The Lebesgue measure of the $L_2$-Informed set decreases as the solution improves, leading to a focused search. However, Informed Sampling effectively resorts to uniform random sampling if the Lebesgue measure of the Informed Set is comparable to that of the entire search space. This can happen if the heuristic estimate of the solution cost fails to provide a good enough approximation of the true solution cost.

Arslan and Tsiotras introduced the Relevant Region in \cite{13}. These authors proposed a selective vertex inclusion procedure along with a machine learning approach \cite{13} to generate new samples in the Relevant Region. However, the approaches in \cite{13} and \cite{14} fall into the category of rejection sampling methods, which do not scale well for high dimensional problems. The current work rigorously defines the Relevant Region set, analyzes its theoretical properties.
and presents a generative method to sample it. The proposed algorithm first selects a vertex in the Relevant Region and then generates a new sample in its neighborhood, while ensuring that it still lies in the Relevant Region.

The Expansive space trees (EST) algorithm [15] also proceeds by selecting a vertex (with probability inversely proportional to number of vertices in its neighborhood) and generates a new sample in its vicinity. Guided ESTs [16] add the A* cost and an exploration term to the vertex weights. The SBA* algorithm [17] incorporates a graph density and a constriction measure into the vertex weight. The algorithm balances the exploitative iterations of SBA* with the exploration oriented iterations of RRT* using simulated annealing. While the algorithm proposed in this work falls into the category of EST-like methods, a crucial difference is that it only expands vertices and generates new samples in the Relevant Region. Thus, in contrast to EST and its variants, the proposed algorithm avoids needless exploration.

The SBMP algorithms and the exploration strategies mentioned above are traditionally geared towards finding the (length) optimal path in uniform cost spaces. However, many applications require planning algorithms to find the optimal path with respect to a provided cost function. These include the problem of navigation on a rough terrain for a mobile robot (see Fig. 1), safety critical path planning with clearance cost-map (example in Fig. 2), human aware motion planning [18], and planning on energy landscapes [19]. The Transition-based RRT (T-RRT) algorithm [20] takes a user-defined cost function as an additional input and adds a transition test based on the Metropolis criterion to accept or reject potential new states. The transition test favors exploration of low-cost regions of space and leads to better quality paths. An enhanced, bi-directional version of T-RRT is presented in [21]. Berenson et al [22] combine gradient information within the T-RRT framework to address the issue of navigating cost-space chasms. Finally, Devaurs et al [23] combine the filtering properties of the transition test with the local rewiring procedure of RRT* to obtain the asymptotically optimal T-RRT* algorithm. While the transition test promotes exploration of low-cost regions, unlike the Informed and Relevant Re-
gion sets, it does not focus the search based on the current sub-optimal solution cost. Secondly, the probabilistic rejection strategy of the transition test might not scale well to higher dimensional spaces, as the probability of generating a “good” sample that can pass the transition test may decrease rapidly. The Relevant Region sampler proposed in this work addresses these issues. It utilizes heuristics, the current solution cost and the cost function information to effectively focus the search in general cost-space environments.

In the following sections, the path planning problem on general cost-maps is formally defined, followed by a comparison between the Informed and the Relevant Region sets. A technique to generate samples in the Relevant Region is then proposed, followed by benchmarking results.

2 PROBLEM DEFINITION

2.1 Path Planning Problem

Consider the search space $X$, which is assumed to be a subset of $\mathbb{R}^d$, where $d$ is a positive integer such that $d \geq 2$. Let $X_{\text{obs}}$ denote the obstacle space and $X_{\text{free}} = \text{cl}(X \setminus X_{\text{obs}})$ denote the free space. Here, $\text{cl}(A)$ represents closure of the set $A \subset \mathbb{R}^d$. Let $M(A)$ denote the Lebesgue measure of the set $A \subset \mathbb{R}^d$. Let $x_s \in X_{\text{free}}$ denote the initial state and $X_{\text{goal}} \subset X_{\text{free}}$ represent the goal region. Let $C : X \rightarrow \mathbb{R} \geq 0$ denote a continuous state cost function that imposes a “cost-map” over the search space. The path-cost from $x_1 \in X$ to $x_2 \in X$ along a path $\pi : [0, 1] \rightarrow X$ with $\pi(0) = x_1$, $\pi(1) = x_2$ is given by

$$d_\pi(x_1, x_2) = \int_0^1 C(\pi(s)) \|d\pi(s)/ds\|_2 ds. \quad (1)$$

The above equation represents the integral of cost (IC) criterion as a measure of path quality. If the path is a straight line, i.e., $\pi(s) = x_1 + (x_2 - x_1)s$, $s \in [0, 1]$, then the IC cost is given by

$$d_\ell(x_1, x_2) = \|x_2 - x_1\|_2 \int_0^1 C(x_1 + (x_2 - x_1)s) ds. \quad (2)$$

Note that if $C(x) = 1$ for all $x \in X$, then the IC cost in (2) reduces to the familiar Euclidean distance $\|x_2 - x_1\|_2$. Let $\Pi$ denote the set of paths from $x_s$ to $X_{\text{goal}}$. Thus, given $(X, X_{\text{obs}}, x_s, X_{\text{goal}}, C)$, the optimal path planning problem is one of finding the minimum-cost, feasible path $\pi^* \in \Pi$.

$$\arg \min_{\pi \in \Pi} d_\pi(x_s, x_g)$$

subject to: $\pi(0) = x_s$, $\pi(1) = x_g \in X_{\text{goal}}$, $\pi(s) \in X_{\text{free}}$, $s \in [0, 1]$. \quad (3)

Consider the graph $G = (V, E)$ that encodes the connectivity between a finite set of vertices $V \subset X_{\text{free}}$ with edges $E \subseteq V \times V$. A spanning tree $T = (V_s, E_s)$ is
embedded in $\mathcal{G}$ such that $V_s = V$ and $E_s = \{(u, v) \in E \mid v = \text{parent}(u)\}$. Here the function $\text{parent} : V \rightarrow V$ represents the mapping from a vertex to its unique parent vertex. By definition, $\text{parent}(x_s) = x_s$. SBMP algorithms numerically integrate (2) to calculate the edge-cost $d_{\ell}(v, u)$ for any edge $(u, v) \in E$. This work considers cost functions with $C(x) \geq 1$ for all $x \in \mathcal{X}$, so that the edge-cost between any two vertices is at least the Euclidean distance between these two points. Given the spanning tree $\mathcal{T}$ in $\mathcal{G}$, the function $g_T : V \rightarrow \mathbb{R}_{\geq 0}$ provides the cost-to-come value for any $v \in V$, i.e., it is the sum of the edge-costs along the path from $v$ to $x_s$ in $\mathcal{T}$. The exploitation module in an SBMP algorithm rewrites $\mathcal{T}$ (that is, re-assigned parents to vertices) so as to minimize the cost-to-come values of the vertices. A consistent heuristic function on $\mathcal{X}$ (such as the Euclidean distance or $L_2$-norm) is defined as: $h : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}_{\geq 0}$. The function $h$ always gives an under-estimate of the path-cost between any two points in the search space, and obeys the triangle inequality. The SBMP algorithms solve the planning problem (3) by drawing random samples from $\mathcal{X}$ and by incorporating the collision-free ones in $\mathcal{G}$. An efficient sampling strategy must generate samples so as to find an initial solution or improve the current one. The exploration problem in SBMP is one of finding such a sampling strategy to yield faster convergence.

2.2 The Informed Set

Let $c_i$ be the cost of the best solution found by the planning algorithm after $i$ iterations. The Informed Set is defined as

$$\mathcal{X}_{\text{inf}} = \{x \in \mathcal{X} \mid h(x_s, x) + h(x, x_g) < c_i\}. \quad (4)$$

Note that $\mathcal{X}_{\text{inf}}$ uses a heuristic approximation of both the cost-to-come and the cost-to-go to get an (under)estimate of the solution cost constrained to pass through any $x \in \mathcal{X}$. All the points with a solution cost estimate greater than $c_i$ are excluded. Generating new samples in $\mathcal{X}_{\text{inf}}$ is thus a necessary condition for improving the current solution. An algorithm for direct sampling of the $L_2$-Informed Set is given in [12].

2.3 Relevant Region

Consider the set of relevant vertices defined as

$$V_{\text{rel}} = \{v \in V \mid g_T(v) + h(v, x_g) < c_i\}. \quad (5)$$

Let $\epsilon > 0$ ball around a relevant vertex $v \in V_{\text{rel}}$ be defined as

$$B'(v) = \{x \in \mathcal{X} \mid \|x - v\|_2 < \epsilon, \ v \in V_{\text{rel}}\}. \quad (6)$$

Consider the estimate of the solution cost constrained to pass through $x \in B'(v)$

$$\hat{f}_v(x) = d_{\ell}(x, v) + g_T(v) + h(x, x_g). \quad (7)$$
The Relevant Set around \( v \in V_{rel} \) is defined as
\[
B^\epsilon_{rel}(v) = \{ x \in B^\epsilon(v) \mid \hat{f}_v(x) < c_i \}. \tag{8}
\]
Using (5), (8), the Relevant Region is defined as the union of the relevant sets around all relevant vertices
\[
X_{rel}^\epsilon = \bigcup_{v \in V_{rel}} B_{rel}^\epsilon(v). \tag{9}
\]
The value of \( \epsilon \), which controls the size of the Relevant Set, is taken to be slightly greater than the step-size parameter \( \eta \) (in our benchmarking simulations, we used \( \epsilon = 1.5\eta \)). The step-size parameter \( \eta \) in SBMP controls the maximum edge length in \( G \) \cite{12}. Note that a very small value of \( \epsilon \) would hinder exploration, while a large value of \( \epsilon \) may provide a poor estimate of the cost-to-come in (7), as the edge \((x, v)\) may not be feasible. The following theorem proves that for any \( \epsilon > 0 \), \( B_{rel}^\epsilon(v) \) is not a singleton.

**Theorem 1.** For every \( v \in V_{rel} \), there exists \( \delta > 0 \), such that, for all \( x \in B^\delta(v) \), it follows that \( \hat{f}_v(x) < c_i \).

**Proof.** Note from (7) that for each given \( v \in V_{rel} \) the function \( \hat{f}_v \) is continuous in \( x \) since both \( d_\ell(\cdot, v), h(\cdot, x_\delta) \) are continuous. Also, \( \hat{f}_v(v) = g_\ell(v) + h(v, x_\delta) < c_i \) since \( v \in V_{rel} \). Since \( \hat{f}_v \) is continuous at \( v \), it follows that for any \( \zeta > 0 \) there exists \( \delta > 0 \) such that \( x \in B^\delta(v) \) implies that \( |\hat{f}_v(x) - \hat{f}_v(v)| < \zeta \). Choosing \( \zeta = c_i - \hat{f}_v(v) > 0 \) one then obtains that for all \( x \in B^\delta(v) \) we have that \( |\hat{f}_v(x) - \hat{f}_v(v)| < c_i - \hat{f}_v(v) \) and hence \( \hat{f}_v(x) < c_i \). \( \square \)

**Corollary 2.** Let \( v \in V_{rel} \). For every \( \epsilon > 0 \) there exists \( \delta > 0 \) such \( B^\delta(v) \subset B_{rel}^\epsilon(v) \).

**Theorem 3.** For any \( \epsilon > 0 \), the Relevant Region \( X_{rel}^\epsilon \) is a subset of the Informed Set \( X_{inf} \).

**Proof.** Let \( x \in X_{rel}^\epsilon \). Then there exists \( v \in V_{rel} \) so that \( x \in B^\epsilon_{rel}(v) \), and hence \( d_\ell(x, v) + g_\ell(v) + h(x, x_\delta) < c_i \). Since the heuristic function is consistent, \( h(x, v) < d_\ell(x, v) \) and \( h(v, x_\delta) < g_\ell(v) \). Using the triangle inequality, it follows that \( h(x, x_\delta) < h(x, v) + h(v, x_\delta) \). Combining the above inequalities yields \( h(x, x_\delta) < d_\ell(x, v) + g_\ell(v) + h(x, x_\delta) < c_i \). Hence, \( x \in X_{inf} \). It follows that \( X_{rel}^\epsilon \subset X_{inf} \). \( \square \)

Theorem 3 implies that generating samples in \( X_{rel}^\epsilon \) does not lead to redundant exploration outside \( X_{inf} \). Note that, in contrast to \( X_{inf} \) which uses the heuristic estimate \( h(x_\delta, x) \) of the cost-to-come, \( X_{rel}^\epsilon \) uses \( d_\ell(x, v) + g_\ell(v) \) from (7). This approximation considers the cost-function information (see (2)), the structure of \( \mathcal{T} \), and hence the topology of \( X_{free}^\epsilon \). While the \( L_2 \)-norm is still a consistent heuristic for cost-maps with \( C(x) \geq 1 \) for all \( x \in X \), it does not take into account \( C \) or \( X_{obs} \). It may provide a poor estimate of the solution cost, leading
to $\mathcal{M}(\mathcal{X}_{\text{inf}}) \approx \mathcal{M}(\mathcal{X})$. Informed Sampling effectively resorts to uniform random sampling in this case. However, note that sampling in $\mathcal{X}_{\text{rel}}$ is not a necessary condition for improving the current solution, i.e., there may be points $x \in \mathcal{X}_{\text{inf}}$ such that $x \notin \mathcal{X}_{\text{rel}}$ which may improve the current solution. Relevant Region sampling is thus utilized in conjunction with Informed/Uniform Sampling. As shown in the numerical examples later on, this interplay of exploration by Informed Sampling, combined with focusing properties of Relevant Region, leads to accelerated convergence.

3 SAMPLING IN THE RELEVANT REGION

Since $\mathcal{X}_{\text{rel}}$ depends on $\mathcal{T}$, a direct sampling strategy is not possible. Hence, the proposed sampling strategy proceeds by first selecting a relevant vertex $v_p \in V_{\text{rel}}$, sampling a random direction $\hat{e}$, $\|\hat{e}\|_2 = 1$ and finding the maximum magnitude of travel $\gamma_{\text{rel}} > 0$ along $\hat{e}$, so that for all $\gamma \in (0, \gamma_{\text{rel}})$ the new sample $x = v_p + \gamma \hat{e} \in B_{\text{rel}}(v_p)$. Please see Fig. 4. Note that Theorem 1 guarantees the existence of $\gamma_{\text{rel}}$. Concretely, the following optimization problem needs to be solved:

$$\sup_{\gamma \in (0, \epsilon)} \gamma,$$

subject to: $\hat{f}_{v_p}(v_p + \gamma \hat{e}) < c_i$. (10)

3.1 Case 1: Uniform Cost-Map

Consider the problem (10) with $C(x) = 1$ for all $x \in \mathcal{X}$. Using the $L_2$-norm heuristic in (7), the inequality in (10) yields,

$$\hat{f}_{v_p}(v_p + \gamma \hat{e}) = \gamma + \hat{g}(v_p) + \|v_p + \gamma \hat{e} - x_g\|_2 < c_i.$$ (11)
Rearrange the terms in (11) to obtain

\[ \|v_p + \gamma \hat{e} - x_g\|_2 < c_i - g_T(v_p) - \gamma. \] (12)

To ensure that the RHS in (12) is positive, choose

\[ \gamma < c_i - g_T(v_p). \] (13)

Let \( x_{pg} = v_p - x_g \) and \( g_{gp} = c_i - g_T(v_p) \). Also note that \( x_{pg}^T x_{pg} = h^2(v_p, x_g) \) and \( x_{pg}^T \hat{e} = h(v_p, x_g) \cos \theta \), where \( \theta \) is the angle between the vectors \( x_{pg} \) and \( \hat{e} \).

Squaring both sides in (12) yields,

\[ h^2(v_p, x_g) + 2 \gamma x_{pg}^T \hat{e} + \gamma^2 < \gamma g_{gp}^2 - 2 \gamma g_{gp} + \gamma^2 \]

and hence

\[ \gamma < \frac{g_{gp}^2 - h^2(v_p, x_g)}{2(x_{pg}^T \hat{e} + g_{gp})}. \]

Define the RHS in the above inequality as

\[ \gamma_{uni} = \frac{(c_i - g_T(v_p))^2 - h^2(v_p, x_g)}{2[h(v_p, x_g) \cos \theta + (c_i - g_T(v_p))]}. \] (14)

Note that \( \gamma_{uni} > 0 \) for \( v_p \in V_{rel} \), and attains its maximum value \( \gamma_{uni} \) at \( \theta = \pi \), in which case,

\[ \gamma_{uni} = \frac{(c_i - g_T(v_p) + h(v_p, x_g))/2}{2}. \]

and also, \( \gamma_{uni} < c_i - g_T(v_p) \) for \( v_p \in V_{rel} \), satisfying (13). Thus, the solution to problem (10) for uniform cost-map is

\[ \gamma_{rel} = \min(\gamma_{uni}, \epsilon). \] (15)

### 3.2 Case 2: General Cost-Maps

Now consider the problem (10) with \( C(x) > 1 \) for all \( x \in X \). The following inequality needs to be solved for \( \gamma \),

\[ \gamma \int_0^1 C(v_p + \gamma \hat{e} s) ds + g_T(v_p) + h(v_p + \gamma \hat{e}, x_g) < c_i. \] (16)
Often, \( C \) may not have a tractable closed-form expression and hence the planner has access only to the value of \( C \) at any point in the search space. In order to avoid a computationally expensive procedure to solve (16), we let

\[
\gamma \int_0^1 C(v_p + \gamma \hat{e}) ds \approx \gamma C(v_p) + \frac{1}{2} C'(v_p) \gamma^2 \hat{e}.
\]

(17)

For small \( \gamma \), it then follows that,

\[
\gamma C(v_p) + g_T(v_p) + \|v_p + \gamma \hat{e} - x_g\|_2 < c_i,
\]

(18)

or,

\[
\|v_p + \gamma \hat{e} - x_g\|_2 < c_i - g_T(v_p) - \gamma C(v_p).
\]

(19)

To ensure that the RHS of (19) is positive, choose

\[
\gamma < (c_i - g_T(v_p))/C(v_p).
\]

(20)

Let again \( x_{pg} = v_p - x_g \); \( g_{gp} = c_i - g_T(v_p) \). \( x_{pg}^T x_{pg} = h^2(v_p, x_g) \) and \( x_{pg}^T \hat{e} = h(v_p, x_g) \cos \theta \), where \( \theta \) is the angle between the vectors \( x_{pg} \) and \( \hat{e} \). Squaring both sides in (19) and simplifying yields,

\[
\gamma^2 (C^2(v_p) - 1) - 2\gamma (g_{gp} C(v_p) + x_{pg}^T \hat{e}) + g_{gp}^2 - h^2(v_p, x_g) > 0.
\]

(21)

Let \( \gamma_1, \gamma_2 \) be the roots of the quadratic equation corresponding to inequality (21), and assume \( \gamma_2 > \gamma_1 \).

\[
\gamma_2 = \frac{g_{gp} C(v_p) + h(v_p, x_g) \cos \theta + \sqrt{\Delta}}{(C^2(v_p) - 1)}
\]

\[
\gamma_1 = \frac{g_{gp} C(v_p) + h(v_p, x_g) \cos \theta - \sqrt{\Delta}}{(C^2(v_p) - 1)}
\]

\[
\Delta = (g_{gp} C(v_p) + h(v_p, x_g) \cos \theta)^2 - (C^2(v_p) - 1)(g_{gp}^2 - h^2(v_p, x_g)).
\]

(22)

The maximum and minimum values of the radicand \( \Delta \) are obtained at \( \theta = 0 \) and \( \theta = \pi \), respectively, where

\[
(g_{gp} - h(v_p, x_g) C(v_p))^2 \leq \Delta \leq (g_{gp} + h(v_p, x_g) C(v_p))^2.
\]

(23)

Hence, \( \gamma_1, \gamma_2 \in \mathbb{R}_{\geq 0} \) for \( v_p \in V_{rel} \). Then (21) yields,

\[
(\gamma - \gamma_1)(\gamma - \gamma_2) > 0, \text{ equivalently, } \gamma > \gamma_2 \text{ or } \gamma < \gamma_1.
\]

(24)

Consider the larger root \( \gamma_2 \) from (22). The minimum value of \( \gamma_2 \) is attained when \( \theta = \pi \), so that

\[
\gamma_2 \geq \frac{g_{gp} C(v_p) - h(v_p, x_g) + \sqrt{\Delta}}{(C^2(v_p) - 1)}.
\]

(25)
Algorithm 1: Sampling Algorithm Flow

1. $V \leftarrow \{x_s\}; E \leftarrow \emptyset; G \leftarrow (V, E);
2. for $i = 1 : N$ do
3.   $c_i \leftarrow \min_{v \in V_{rel}} g_T(v);
4.   $u_{rand} \sim U(0, 1);
5.   if $u_{rand} < p_{rel}$ and $c_i < \infty$ then
6.     $v_p \leftarrow \text{chooseVertex}(V_{rel});$
7.     $e \leftarrow \text{generateDirection}();$
8.     $\gamma_{rel} \leftarrow \text{RelevantStepLimit}(v_p, e);$
9.     $u_{rand} \sim U(0, 1);$
10.    $x_{rand} \leftarrow v_p + (u_{rand})^{\frac{1}{2}} \gamma_{rel} e;$
11.   else
12.     $x_{rand} \leftarrow \text{InformedSampling}()$
13.    $x_{new} \leftarrow \text{Extend}(x_{rand});$
14.    $\text{Exploitation}(G);$
15. return $G$

Define the RHS in (25) as $\gamma_2$. Simplifying yields,

$$\gamma_2 = \begin{cases} 
\frac{g_{gp} + h(v_p, x_g)}{C(v_p) + 1}, & g_{gp} < h(v_p, x_g) C(v_p), \\
\frac{g_{gp} - h(v_p, x_g)}{C(v_p) - 1}, & g_{gp} > h(v_p, x_g) C(v_p). 
\end{cases}$$

(26)

Note that $\gamma_2 > g_{gp}/C(v_p)$. This implies $\gamma > \gamma_2$ is an infeasible solution of (18). Next, consider $\gamma_1$. Differentiating with respect to $\theta$, the extrema are obtained at $\theta = 0, \pi$. Calculating the second derivative yields, $\gamma_1''(\theta = 0) > 0$ and $\gamma_1''(\theta = \pi) < 0$. The maximum value of $\gamma_1$ obtained at $\theta = \pi$ is given by

$$\gamma_1 = \begin{cases} 
\frac{g_{gp} + h(v_p, x_g)}{C(v_p) + 1}, & g_{gp} < h(v_p, x_g) C(v_p), \\
\frac{g_{gp} - h(v_p, x_g)}{C(v_p) - 1}, & g_{gp} > h(v_p, x_g) C(v_p). 
\end{cases}$$

(27)

Now, $\gamma_1 < g_{gp}/C(v_p)$. This implies $\gamma_1 < g_{gp}/C(v_p)$. It follows that $\gamma < \gamma_1$ satisfies (20). Thus, the solution to problem (10) with the approximation in (18) is

$$\gamma_{rel} = \min(\gamma_1, \epsilon).$$

(28)

For the special case when $\Delta = 0$ and $\gamma_1 = \gamma_2 = \gamma_c$, inequality (21) simplifies to $(\gamma - \gamma_c)^2 > 0$. Considering (20) yields $\gamma_{rel} = \min(g_{gp}/C(v_p), \epsilon)$. Note that if $C(v_p) = 1$, then inequality (10) reduces to (11) and the analysis for uniform cost-maps is applicable.

4 PROPOSED ALGORITHM

The outline of the proposed algorithm in given in Algorithm 1. The procedure initializes a vertex at the start state $x_s$. At every iteration, the current best
solution cost $c_i$ is updated (line 3). If a sub-optimal solution exists ($c_i$ is finite), with probability $p_{rel}$ (line 5), Relevant Region sampling is employed to generate a new random sample $x_{rand}$. Otherwise, conventional Informed Sampling is used. Relevant Region sampling consists of first choosing a relevant vertex $v_p$, generating a random direction $\hat{e}$ and calculating the maximum magnitude of travel along $\hat{e}$ (line 6-8). If $C(v_p) = 1$, then (15) is used for obtaining $\gamma_{rel}$ along $\hat{e}$, else (28) is used. After $x_{rand}$ is generated, conventional SBMP modules incorporate a new vertex $x_{new}$ in $G$ (line 13). These include: a) finding the nearest neighbor $x_{nearest}$ to $x_{rand}$ in $G$; b) local steering from $x_{nearest}$ in the direction of $x_{rand}$ to obtain $x_{new}$; c) ensuring feasibility of edge-connections in the neighborhood of $x_{new}$. This is followed by the exploitation module (local/global rewiring, etc). The chooseVertex module selects a relevant vertex to be expanded from the set $V_{rel}$. Similar to the procedure in Guided-ESTs [16] a weight $q_v$ is allocated for each $v \in V_{rel}$.

$$q_v = \lambda_1 p_v + \lambda_2 d_v + \lambda_3 \left( g_T(v) + h(v, x_g) \right) / c_i.$$  

(29)

Here, $p_v$ represents the number of times $v$ has been selected in the past. This penalizes multiple selections and the exploration of the region around a particular vertex. The second term, $d_v$ is the number of edges connected to $v$. It promotes sampling in relatively unexplored regions. The last term $0 < \left( g_T(v) + h(v, x_g) \right) / c_i < 1$ is the estimate of the solution cost through $v$, normalized by the current best cost. This prioritizes exploration of regions with low solution cost estimates. The parameters $(\lambda_1, \lambda_2, \lambda_3) > 0$ modulate the behavior of the selection algorithm. A large value of $\lambda_3$ leads to a greedy focus on low solution cost areas, whereas increasing $\lambda_1, \lambda_2$ promotes exploration. A binary heap is used to update and sort $V_{rel}$ according to the weight in (29). A relevant vertex $v_p$ is selected by choosing randomly from the top $n_q$ elements in the sorted list. This injects randomness in the selection process and promotes desirable exploration.

5 NUMERICAL EXPERIMENTS

The performance of the proposed sampling method was benchmarked against direct Informed Sampling [12] and the third variant of adaptive rejection sampling (described in [13]) in uniform cost-space environments (length-optimal planning). For all experiments, the exploration strategies were paired with RRT#’s dynamic programming based global rewiring for exploitation. In general cost-map environments, benchmarking was done against Informed Sampling and T-RRT# (combining conventional RRT# with the transition-test described in [20]) with different initial temperatures $T_{init}$. All the algorithms were implemented in C++ using the popular OMPL framework [24], and the tests were run using OMPL’s standardized benchmarking tools [25]. Please see, https://github.gatech.edu/DCSL/relevant_region. A 64-bit desktop PC with 64 GB RAM and an Intel Xeon(R) Processor running Ubuntu 16.04 OS was used. The data was recorded over 100 trials for all the cases. The
proposed algorithm used the following parameter values: $p_{rel} = 0.5$, $\epsilon = 1.5\eta$, $(\lambda_1, \lambda_2, \lambda_3) = (10, 5, 100)$, $n_q = 10$. A goal bias of 5% was used in all sampling methods. A description of the different environments is provided below.

5.1 General Cost-Map Cases

Terrain Map: A 2D terrain map shown in Fig. 1 consists of rough, high-cost white areas and the easily navigable black regions. The step-size was set to $\eta = 0.3$ for this example.

Potential Cost-Map: The environment in Fig. 2 emulates the problem of finding the shortest path while staying away from danger areas (white regions). The cost function is defined as

$$C(x) = 1 + c_o \left( e^{-\frac{||x - x_{d1}||^2}{\sigma}} + e^{-\frac{||x - x_{d2}||^2}{\sigma}} \right).$$ (30)

Here, $x_{d1}^d = [1, 5]^T$, $x_{d2}^d = [7, 3]^T$ are the center points of the danger regions. A step-size of $\eta = 0.3$ was used in 2D and $\eta = 1.5$ in the 6D version of the environment. For both cases, $c_o = 9$, $\sigma = 5$ were used.

5.2 Uniform Cost-Map Cases

Multiple Obstacle World: This environment is illustrated in Fig. 3. The 2D environment was extended to $\mathbb{R}^6$ by imparting a length of 2 units symmetrically to all of the obstacles. A step-size of $\eta = 0.3$ was used in $\mathbb{R}^2$ and $\eta = 1.2$ in $\mathbb{R}^6$.

Panda Arm: A planning problem for 7-DOF Panda Arm (by Franka Emnika) is illustrated in Fig. 5. The objective was to find a minimum length path in
a 7-dimensional configuration (joint) space with joint limits. These limits and collision checking module were implemented using MoveIt! [26]. The step-size was set to $\eta = 0.7$ for this example.

6 CONCLUSION

This work proposes a new algorithm to sample the Relevant Region set, a subset of the Informed Set, for SBMP. Informed Sampling uses a purely heuristic estimate of the solution cost, which may not be effective in general cost-space environments. The Relevant Region set considers the topology of $X_{\text{free}}$, reduces the dependence on heuristics, and effectively focuses the search to accelerate convergence. Numerical experiments validate the utility of Relevant Region sampling in conjunction with Informed/Uniform Sampling. The proposed method leads to faster convergence in all cases (see Fig. 7). This behavior is observed especially in higher dimensional problem instances. Transition-test based exploration is more effective than Uniform/Informed Sampling for planning on general cost-maps. However, the tendency to (probabilistically) reject samples may hinder exploration in some cases. This can be seen in the terrain cost-map (Fig. 1) which is similar to the cost-space chasms scenario described in [22]. As conveyed in Fig. 6, the transition-test based exploration fails to find a feasible solution in roughly 40% of total trials, whereas the proposed method finds a solution in all trials.

This work presents many avenues for future research. A simulated annealing-like procedure can be implemented to balance Relevant Region and Informed Sampling to eventually focus the search to the Relevant Region. The cost function’s gradient information (if available) can be used to bias the search. Data from past iterations can also be used to infer the nature of cost-map for intelligent exploration.

Acknowledgment: This work has been supported by NSF award IIS-1617630.
References

[1] S. M. LaValle and J. J. Kuffner Jr, “Randomized kinodynamic planning,” The International Journal of Robotics Research, vol. 20, no. 5, pp. 378–400, 2001.

[2] S. Karaman and E. Frazzoli, “Sampling-based algorithms for optimal motion planning,” The International Journal of Robotics Research, vol. 30, no. 7, pp. 846–894, June 2011.

[3] O. Arslan and P. Tsiotras, “Use of relaxation methods in sampling-based algorithms for optimal motion planning,” in IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, May 6–10 2013, pp. 2421–2428.

[4] F. Hauer and P. Tsiotras, “Deformable rapidly-exploring random trees.” in Robotics: Science and Systems, Cambridge, MA, July 12–16 2017.

[5] J. D. Gammell, S. S. Srinivasa, and T. D. Barfoot, “Batch informed trees (BIT*): Sampling-based optimal planning via the heuristically guided search of implicit random geometric graphs,” in IEEE International Conference on Robotics and Automation, Seattle, WA, May 25–30 2015, pp. 3067–3074.

[6] L. Janson, E. Schmerling, A. Clark, and M. Pavone, “Fast marching tree: A fast marching sampling-based method for optimal motion planning in many dimensions,” The International Journal of Robotics Research, vol. 34, no. 7, pp. 883–921, May 2015.

[7] B. Akgun and M. Stilman, “Sampling heuristics for optimal motion planning in high dimensions,” in IEEE/RSJ International Conference on Intelligent Robots and Systems, San Francisco, CA, Sept. 25–30 2011, pp. 2640–2645.

[8] C. Urmson and R. Simmons, “Approaches for heuristically biasing RRT growth,” in IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 2, Las Vegas, NV, Oct. 27–31 2003, pp. 1178–1183.

[9] B. Burns and O. Brock, “Toward optimal configuration space sampling,” in Robotics: Science and Systems, Cambridge, MA, June 8–11 2005, pp. 105–112.

[10] ——, “Single-query entropy-guided path planning,” in IEEE International Conference on Robotics and Automation, Barcelona, Spain, 4 2005, pp. 2124–2129.

[11] R. Diankov and J. Kuffner, “Randomized statistical path planning,” in IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, CA, Oct. 29–31 2007, pp. 1–6.
[12] J. D. Gammell, T. D. Barfoot, and S. S. Srinivasa, “Informed sampling for asymptotically optimal path planning,” *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 966–984, Aug. 2018.

[13] O. Arslan and P. Tsiotras, “Dynamic programming guided exploration for sampling-based motion planning algorithms,” in *IEEE International Conference on Robotics and Automation*, Seattle, WA, May 26–30 2015, pp. 4819–4826.

[14] ——, “Machine learning guided exploration for sampling-based motion planning algorithms,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Hamburg, Germany, Sept. 28–Oct. 2, 2015, pp. 2646–2652.

[15] D. Hsu, J.-C. Latombe, and R. Motwani, “Path planning in expansive configuration spaces,” in *IEEE International Conference on Robotics and Automation*, vol. 3, Albuquerque, NM, April 25–29 1997, pp. 2719–2726.

[16] J. M. Phillips, N. Bedrossian, and L. E. Kavraki, “Guided expansive spaces trees: a search strategy for motion-and cost-constrained state spaces,” in *IEEE International Conference on Robotics and Automation*, New Orleans, LA, April 26–30 2004, pp. 3968–3973.

[17] S. M. Persson and I. Sharf, “Sampling-based A* algorithm for robot path-planning,” *The International Journal of Robotics Research*, vol. 33, no. 13, pp. 1683–1708, 10 2014.

[18] J. Mainprice, E. A. Sisbot, L. Jailliet, J. Cortés, R. Alami, and T. Siméon, “Planning human-aware motions using a sampling-based costmap planner,” in *IEEE International Conference on Robotics and Automation*, Shanghai, China, May 9–13 2011, pp. 5012–5017.

[19] L. Jailliet, F. J. Corcho, J.-J. Pérez, and J. Cortés, “Randomized tree construction algorithm to explore energy landscapes,” *Journal of Computational Chemistry*, vol. 32, no. 16, pp. 3464–3474, 2011.

[20] L. Jailliet, J. Cortés, and T. Siméon, “Sampling-based path planning on configuration-space costmaps,” *IEEE Transactions on Robotics*, vol. 26, no. 4, pp. 635–646, 8 2010.

[21] D. Devaurs, T. Siméon, and J. Cortés, “Enhancing the transition-based RRT to deal with complex cost spaces,” in *IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, May 6–10 2013, pp. 4120–4125.

[22] D. Berenson, T. Siméon, and S. S. Srinivasa, “Addressing cost-space chasms in manipulation planning,” in *IEEE International Conference on Robotics and Automation*, Shanghai, China, May 9–13 2011, pp. 4561–4568.
[23] D. Dévaurs, T. Siméon, and J. Cortés, “Optimal path planning in complex cost spaces with sampling-based algorithms,” *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 415–424, 2015.

[24] I. A. Sucan, M. Moll, and L. E. Kavraki, “The open motion planning library,” *IEEE Robotics & Automation Magazine*, vol. 19, no. 4, pp. 72–82, Dec. 2012.

[25] M. Moll, I. A. Sucan, and L. E. Kavraki, “Benchmarking motion planning algorithms: An extensible infrastructure for analysis and visualization,” *IEEE Robotics & Automation Magazine*, vol. 22, no. 3, pp. 96–102, 2015.

[26] S. Chitta, I. Sucan, and S. Cousins, “Moveit![ROS topics],” *IEEE Robotics & Automation Magazine*, vol. 19, no. 1, pp. 18–19, 2012.