A Scenario Approach to Robust Simulation-based Path Planning

Shaunak D. Bopardikar Vaibhav Srivastava

***Abstract*— We consider a prototypical path planning problem on a graph with uncertain cost of mobility on its edges. At a given node, the planning agent can access the true cost for edges to its neighbors and uses a noisy simulator to estimate the cost-to-go from the neighboring nodes. The objective of the planning agent is to select a neighboring node such that, with high probability, the cost-to-go is minimized for the worst possible realization of uncertain parameters in the simulator. By modeling the cost-to-go as a Gaussian process (GP) for every realization of the uncertain parameters, we apply a scenario approach in which we draw fixed independent samples of the uncertain parameter. We present a *scenario-based iterative algorithm* using the upper confidence bound (UCB) of the fixed independent scenarios to compute the choice of the neighbor to go to. We characterize the performance of the proposed algorithm in terms of a novel notion of regret defined with respect to an additional draw of the uncertain parameter, termed as *scenario regret under re-draw*. In particular, we characterize a high probability upper bound on the regret under re-draw for any finite number of iterations of the algorithm, and show that this upper bound tends to zero asymptotically with the number of iterations. We supplement our analysis with numerical results.**

I. INTRODUCTION

Several motion planning problems involve optimization of costs computed out of complex simulations. In these problems, one rarely has access to the actual cost-to-go and computing gradients of the cost can be infeasible. In a bandit setting, one can only measure the cost at any future location on a roadmap and obtain a noise-corrupted value. A standard metric to analyze the performance of algorithms for bandit problems is the notion of regret, which is the average difference between the cost evaluated at multiple future locations and the unknown optimal cost. This paper applies a scenario-based framework to optimize the cost in a motion planning problem and performs finite-time and asymptotic analysis of a novel robust notion of regret.

Gaussian processes (GP) offer a rigorous framework for function approximation [1]. GPs can capture both epistemic (due to limited data) and aleatoric (modeling) uncertainties [2]. They have been used extensively in path planning to model mobile obstacles [3], for data-efficient learning [4] as well as in the motion planning problem [5]. The key idea is to use smooth continuous-time trajectories as samples from a GP and then view the planning problem as one of probabilistic inference [6]. GPs have been used to design robust controllers for motion planning [7] and for learning-based exploration [8]. Robust motion planning techniques require efficient evaluation of the *belief* of the robot about its own location and about the environment, e.g., [9], [10]. Monte Carlo tree search performs an asymmetric expansion of a search tree using a UCB-based policy [11], [12]. The technique has been applied to robotic motion planning problems in [13], [14], [15]. This paper differs in the sense that the cost to go can be simulated by querying only neighboring locations in presence of uncertainty in terms of the measurement noise as well as unknown but random parameters in the simulation.

For a cost function that is defined jointly over the future locations selected by the user and uncertain parameters that are inaccessible to the user during the simulation (and are selected later by the nature), there are no theoretical guarantees (to the best of our knowledge) on how the iterative solution will perform against the actual realization of the uncertain parameter. We adopt a scenario approach for this problem wherein several realizations of the uncertain parameters are sampled and the function is optimized for the worst scenario. Scenario approaches have been used extensively to solve computationally complex robust optimization arising in control design (cf. the review papers [16], [17]).

This paper introduces a scenario approach applied to a robust motion planning problem over a graph in a bandit setting. We assume that the cost-to-go from every neighboring vertex can be modeled as a GP for every realization of the uncertain parameter. The key contributions of this paper are three-fold. First, we formalize an approach based on scenario optimization, in which we draw fixed independent samples of the uncertain parameter and examine a novel notion of regret, termed as the *scenario regret under re-draw*. Second, we present a scenario-based iterative algorithm using the UCB of multiple scenarios to decide at what future location(s) to simulate the cost-to-go. Third, we derive a high probability upper bound on the regret for any finite number of iterations of the algorithm. This bound is a multiple of the regret in the nominal case of no uncertainty and the extra factor scales logarithmically with the cardinality of the input and the uncertainty set. We show that the regret tends to zero asymptotically with the number of iterations, with high probability. We supplement our analysis with two case studies arising in motion planning for which the scenario approach consistently provides a lower regret compared to a baseline comprising the standard UCB algorithm applied to the average value of the uncertain parameter, at the expense of additional computation.

This work has been supported in part by NSF Awards ECCS-2030556, ECCS-2024649 and IIS-1734272. The authors are with the ECE Department, Michigan State University, USA. E-mail: {shaunak,vaibhav}@egr.msu.edu.

This paper is organized as follows. Section II presents the problem formulation and formally introduces the notion of regret. Section III presents the Scenario UCB algorithm and analytic results on the regret. Section IV illustrates the results on the two case studies. Section V summarizes the work done and outlines future directions. *The appendix contains the proofs of all the theoretical results.*

II. Problem set-up

We consider an environment modeled as a graph with a given start location S, a destination D and several intermediate vertices connected by a set of edges. The location S is connected to a set of neighboring vertices $\mathcal{N}(S) \subset \mathbb{R}^d \times \cdots \times \mathbb{R}^d$, which we number from $\{1, \ldots, M\}$, without any loss of generality. Each edge $e_{S,i}, \forall i \in \{1, \ldots, M\}$, has a reward $W_{S,i} \in \mathbb{R}$ associated with it. The optimal path computation from any vertex i to D involves running a simulation to obtain a realization of the *reward-to-go* $V(x_{S,i}, \delta)$, where $x_{S,i}$ is the location of the i-th neighbor of S and $\delta \in \Delta$ represents the uncertainty involved in the reward-to-go. The value function V represents the reward over the optimal path to the destination, i.e., the sequence of edges with the highest sum of edge rewards. In particular, δ comprises the uncertainty in the reward over all of the set of edges of the graph. Given a realization of δ, the optimal choice of vertex to go to from S is obtained by solving the following Bellman equation,

$$x_{S,i^*} \in \text{argmax}_{i \in \{1,\ldots,M\}} \{W_{S,i} + V(x_{S,i}, \delta)\}, \quad (1)$$

where i^* is the index of a vertex that maximizes the net reward from S. Since Equation (1) holds for any current vertex S from where a robust version of the path planning problem to the destination is to be solved, for ease of exposition, we will drop the S dependence in Equation (1).

Since δ is uncertain, this paper poses a *robust* version of the above equation which can be written as

$$x_{i^*} \in \text{argmax}_{i \in \{1,\ldots,M\}} \min_{\delta \in \Delta} \{W_i + V(x_i, \delta)\}. \quad (2)$$

Thus, Problem (2) can be written equivalently as

$$\max_{x \in X} \min_{\delta \in \Delta} F(x, \delta), \quad (3)$$

where the real-valued function $F(x_i, \delta) := W_i + V(x_i, \delta)$ and X represents the set of neighbors of the current node.

To be able to derive analytic results, we make the following assumptions on the uncertainty δ and the function F.

Assumption 2.1 (Probabilistic uncertainty): The parameter δ is a random variable taking values in a finite, discrete set Δ with probability distribution function, $\mathbb{P}_\delta : \Delta \to \mathbb{R}_{\geq 0}$. □

Assumption 2.1 is required to establish an analytic bound on the number of scenarios required (cf. Theorem 3 from [18] that leads to Corollary 5.1) to establish probabilistic guarantees. However, in practice, the set Δ may be continuous.

We consider three main categories of uncertainty. The first is the *endogenous* uncertainty that corresponds to fundamental differences in environments of interest. For example, in the context of road transportation, endogenous uncertainty may correspond to variability due to differences in layouts of road networks. The second is the *exogenous* uncertainty that corresponds to variability due to external factors such as number of vehicles in the road network. Together, these two uncertainties describe the underlying objective function that maps to the realization of the GP in our framework. The third is the uncertainty in accessing the realized value of the objective function and is accounted for by the measurement noise. The endogeneous and exogeneous uncertainties are of the epistemic type, while the measurement noise captures the aleatoric uncertainty in our model [2].

In the sequel, we capture the exogenous uncertainty via the uncertain parameter δ and drop the term exogenous for brevity. Specifically, we make the following assumptions on the two sources of uncertainty.

Assumption 2.2 (Robust Gaussian Process): For any fixed value of δ, the function $F(\cdot, \delta)$ is a realization of a spatial GP with the mean function equal to $\mu^\delta(\cdot)$ and a kernel $k^\delta(\cdot,\cdot)$, i.e., $F(\cdot, \delta) \sim \text{GP}(\mu(.), k^\delta(\cdot,\cdot))$. In the following, we will denote such a realization by $\omega \in \Omega$. □

Assumption 2.3 (Decoupled uncertainty): The GP parameter ω and the uncertainty parameter δ can be sampled independently of each other in a decoupled manner. □

Assumption 2.3 holds when the function F, defined over a discrete set of cardinality $|X|$, is a vector of function values given by $\boldsymbol{\mu}^\delta(\cdot) + \Sigma^\delta(\cdot,\cdot)\omega$, where $\boldsymbol{\mu}^\delta(\cdot)$ is a vector of length $|X|$, $\Sigma^\delta(\cdot,\cdot)$ is a full rank square matrix of order $|X|$ and ω is a realization of $|X|$-variate Gaussian with zero mean and identity covariance matrix. Indeed, such F is a realization of a Gaussian process with mean $\boldsymbol{\mu}^\delta(\cdot)$ and kernel function matrix $\Sigma^\delta(\cdot,\cdot)\Sigma^\delta(\cdot,\cdot)^\top$.

Since the realization of the GP is determined by both ω and δ, a natural question to ask is why should we adopt robust optimization instead of learning the actual realization of the GP using the GP optimization framework [19]? Recall that the value of the function at a query point is computed using a noisy simulator that may have access to endogenous uncertainty, but not to exogenous uncertainty. Therefore, we want the simulated values to be robust with respect to the actual realization of the exogenous uncertainty. Furthermore, when the policy that we design using the simulator is applied to the physical system in *real time*, we need to ensure that the performance guarantees (measured via regret bounds) for the simulator also hold for the physical system. Towards this end, we introduce a novel notion of robust regret.

Let $\Delta_N := \{\delta_i \mid i \in \{1, \ldots, N\}\} \in \Delta^N$ be the set of N independent and identically drawn samples of δ. In scenario approach to robust optimization, the robustness is computed with respect to an additional $(N+1)$-th sample of the uncertain parameter δ_{N+1}. We refer to this notion of robustness as *robustness under re-draw*.

Consider the following scenario version of Problem (3),

$$J(\Delta_N) := \max_{x \in X} \min_{\delta \in \Delta_N} F(x, \delta). \quad (4)$$

We define the following notion of regret corresponding to robustness under re-draw.

Definition 2.1 (Scenario regret under re-draw): Given an algorithm generating a sequence of decisions $\{\bar{x}_t(\Delta_N)\}_{t\in\{1,\ldots,T\}}$, the *scenario regret under re-draw* with the set $\Delta_{N+1} := \Delta_N \cup \delta_{N+1}$, is

$$R_T(\Delta_{N+1}) := \frac{1}{T}\sum_{t=1}^{T}(J(\Delta_{N+1}) - \min_{\delta\in\Delta_{N+1}} F(\bar{x}_t, \delta)). \quad (5)$$

Note that in R_T, the sequence of decisions is computed using only N scenarios and the regret is computed after incorporating the $(N + 1)$-th scenario. Recall that these scenarios are unknown realizations of GP and need to be learned in a bandit setting. There are three sources of randomness in the above formulations of the robustness/regret: (i) the random draw of N scenarios; (ii) the random draw of $(N+1)$-th scenario; and (iii) the noise in function evaluation at query point.

We seek to design an algorithm and compute an associated upper bound on R_T. We will focus on finite time as well as asymptotic analysis of the regret associated with the algorithm. Although we assume that the set X is finite and discrete, the asymptotic case remains of interest due to measurement noise. In particular, we are interested in consistent algorithms that are defined as follows.

Definition 2.2 (Consistent Algorithm): An algorithm is said to be *consistent* if it generates a sequence of decisions $\{\bar{x}_t\}$, such that the regret under re-draw asymptotically tends to zero as $T \to +\infty$. □

III. Scenario Regret under Re-draw

In this section, we will first establish robustness properties for the scenario regret under re-draw. Then, we will present an algorithm and establish associated guarantees on the scenario regret under re-draw.

A. Robustness of scenario regret under re-draw

Proposition 3.1 (Probabilistic robustness guarantees): Given the parameters $\eta, \zeta \in (0, 1)$, the scenario regret under re-draw defined in (5) with $N = \lceil \frac{1}{\eta}(\ln|X| + \ln|\Delta| - \ln\zeta)\rceil$, for discrete sets X and Δ, satisfies

$$\mathbb{P}_{\delta_{N+1}}\big\{R_T(\Delta_{N+1}) = R_T(\Delta_N)\big\} > 1 - \eta,$$

with probability at least $1 - \zeta$. □

In Proposition 3.1, the outer probability is with respect to N-sample, and the inner probability is defined with respect to the $(N+1)$-th sample. Proposition 3.1 implies that if we have a large number of samples of uncertain parameter δ, then with high probability, the regret with respect to sampled set Δ_N remains the same with the addition of a new $(N+1)$-th sample δ_{N+1}.

B. A UCB based Algorithm

In view of the robustness guarantees from Proposition 3.1, we now focus on solving problem (4). For ease of exposition, we present our results for the case of the prior mean $\boldsymbol{\mu}^\delta(\cdot) = 0$. Our approach is described in Algorithm 1, in which we maintain a separate GP for each sampled scenario i. Let the simulation for $F(x, \delta_i)$ yield a Gaussian random variable with mean $F(x, \delta_i)$ and variance ρ^2. Suppose that until time t, $F(\cdot, \delta_i)$ has been simulated t_i times at points $A_t^i := \{x_1^i, \ldots, x_{t_i}^i\}$ to obtain noisy simulation results $\mathbf{y}_t^i := [y_1^i, \ldots, y_{t_i}^i]'$. Then, the posterior over F is a Gaussian with mean $\mu_t^i(x)$ and covariance $k_t^{\delta_i}(x, x')$ given by

$$\begin{aligned}\mu_t^i(x) &= \mathbf{k}_t^i(x)^T(K_t^i + \rho^2 I)^{-1}\mathbf{y}_t^i,\\ k_t^i(x, x') &= k^{\delta_i}(x, x') - \mathbf{k}_t^i(x)^t(K_t^i + \rho^2 I)^{-1}\mathbf{k}_t^i(x'),\\ \sigma_t^i(x) &= \sqrt{k_t^i(x, x)},\end{aligned} \quad (6)$$

where $k^{\delta_i}(\cdot,\cdot)$ is the kernel function, the vector $\mathbf{k}_t^i(x) := [k^{\delta_i}(x_1^i, x)\ldots k(x_t^i, x)]^T$, and K_t^i is the positive semi-definite kernel matrix $[k^{\delta_i}(x, x')]_{x,x'\in A_t^i}$.

At each time t, Algorithm 1 maintains an upper confidence bound based surrogate function $\mu_{t-1}^i(x) + \sqrt{\beta_t}\sigma_{t-1}^i(x)$ for $F(x, \delta_i)$ and selects the GP to be updated $F(\cdot, \delta_{i_t})$ and the sampling point x_t for simulation in algorithm steps `7:` and `6:`, respectively. In algorithm step `8:`, $n_t \sim \mathcal{N}(0, \rho^2)$ is the simulation noise. The key feature of Algorithm 1 is that *only one* GP realization is updated at each iteration making the implementation scalable from a computational viewpoint.

Algorithm 1 Scenario Upper Confidence Bound

1: **Input:** N scenarios, $\mathbb{P}_\delta(\cdot)$ and a simulator for $F(\cdot,\cdot)$.
2: Draw a multi-sample $\delta_1, \ldots, \delta_N$ using $\mathbb{P}_\delta(\cdot)$.
3: Choose initial values of GP parameters and the kernel functions, $\{(\mu_0^i(\cdot), \sigma_0^i(\cdot))\}_{i=1}^N$ and $\{k^{\delta_i}(\cdot,\cdot)\}_{i=1}^N$.
4: Set, $\mathcal{D}_0^i = \emptyset, \forall i \in \{1, \ldots, N\}$.
5: **for** $t = 1, 2, \ldots$ **do**
6: Set $x_t := \operatorname{argmax}_{x\in X} \min_{i\in\{1,\ldots,N\}} \mu_{t-1}^i(x) + \sqrt{\beta_t}\sigma_{t-1}^i(x)$
7: Set $i_t \in \operatorname{argmin}_{i\in\{1,\ldots,N\}} \mu_{t-1}^i(x_t) + \sqrt{\beta_t}\sigma_{t-1}^i(x_t)$
8: Obtain $y_t = F(x_t, \delta_{i_t}) + n_t$
9: Update the data, $\mathcal{D}_t := \mathcal{D}_{t-1} \cup \{x_t, y_t, i_t\}$
10: Update GP parameters $(\mu_t^{i_t}, \sigma_t^{i_t})$ using (6).
11: **end for**

We now present bounds on the regret under re-draw (5) for Algorithm 1.

Theorem 3.1 (Regret under re-draw): Let $\hat{\lambda}_j^i$'s denote eigenvalues of the kernel matrix k^{δ_i} evaluated over the set $X \times X$, $\forall i \in \{1, \ldots, N\}$. For Algorithm 1 applied to problem (4) with $\beta_t := 2\log(|X|\pi^2 t^2/(3\epsilon))$ and regret under re-draw (5), the following statement holds.

1) For any $T \geq 1$,

$$\mathbb{P}\Big\{R_T(\Delta_N) \leq \sqrt{\frac{8\beta_T\gamma_T}{T\log(1 + 1/\rho^2)}}\Big\} \geq 1 - \epsilon,$$

where, $\gamma_T^i \in O(\rho^{-2}(T\sum_{j=2}^{|X|}\hat{\lambda}_j^i + \log\big(T\sum_{j=1}^{|X|}\hat{\lambda}_j^i\big)))$ and $\gamma_T = \sum_{i=1}^N \gamma_T^i$.

2) For all $T > |X|$, the claim in 1) holds with the choice of $\gamma_T^i \in O(|X|\log\big(T\sum_{j=1}^{|X|}\hat{\lambda}_j^i\big)$.

□

Proposition 3.1 states that the regret is unaffected by an additional redraw with a high probability, i.e., $R_T(\Delta_{N+1}) =$

$R_T(\Delta_N)$, and Theorem 3.1 provides an explicit upper bound on $R_T(\Delta_N)$ that decays monotonically to zero with T. Thus, we can combine Proposition 3.1 and Theorem 3.1 to obtain the following robustness property.

Corollary 3.1: For the sequence $\{x_t\}_{t=1}^T$ obtained from Algorithm 1 solving problem (4) with $N = \lceil \frac{1}{\eta}(\ln |X| + \ln |\Delta| - \ln \zeta) \rceil$ scenarios, with probability at least $1-\zeta$,

$$\mathbb{P}_{\delta_{N+1}}\Big\{\mathbb{P}\Big\{R_T(\Delta_{N+1}) \leq \sqrt{\frac{8\beta_T\gamma_T}{T\log(1+1/\rho^2)}}\Big\} \geq 1-\epsilon\Big\} \geq 1-\eta. \ \square$$

Note the three levels of probability in Corollary 3.1. The outermost probability of $(1-\zeta)$ is with respect to the N samples drawn in the scenario approach, the middle probability is the probability that measures robustness with respect to the additional draw δ_{N+1}, and the innermost probability is with respect to the temporal realizations of the T measurement noise sequences. We conclude this section with the following result.

Corollary 3.2 (Consistency): With high probability, the regret scales as

$$O\Big(\sqrt{\frac{(\ln |X| + \ln |\Delta| - \ln \zeta)}{\eta} \times \frac{|X|}{T} \log \frac{|X|T^2}{\epsilon} \log(|X|T)}\Big)$$

In other words, Algorithm 1 is consistent in the sense of Definition 2.2. $\square$

The proof of this claim follows from Corollary 3.1 which implies that the regret under re-draw scales as $O(\sqrt{N|X|\log(|X|T^2/\epsilon)\log(|X|T)/T})$ in the general case of any arbitrary $T \geq 1$, and N satisfies the requirement in Corollary 3.1.

Remark 3.1 (Scaling law): In comparison with the regret upper bound for the nominal GP-UCB algorithm [19] that does not account for parameter uncertainty, an extra term $\sqrt{\frac{(\ln |X| + \ln |\Delta| - \ln \zeta)}{\eta}}$ appears in the upper bound in Corollary 3.2. The logarithmic scaling of this term with desired confidence ζ and cardinalities $|X|$ and $|\Delta|$ makes the proposed scenario approach particularly appealing. $\square$

IV. Application to Path Planning

We now present numerical results of implementing Algorithm 1 to a path planning problem over a roadmap between a source and a destination. We present its application to two problems – the first involving minimum Euclidean distance between source and destination and the second involving minimum time based on traffic data from Google Maps. Although our analytic results require discrete Δ, we allow Δ to be a continuum in our simulations.

A. Minimizing Euclidean Distance to Destination

In this problem, we denote the coordinates of each neighboring location of the start node as the optimization variable x and the distance to the destination from that neighbor as the output y. We picked a squared exponential kernel to model the dependence between any two points that are at given distances x and $\tilde{x}$ from the start location, using the dependence $k^{\delta_i}(x,\tilde{x}) := v_i \exp(\|x-\tilde{x}\|^2/\ell_i^2)$. The *hyperparameters* $\{(v_i, \ell_i)\}$ are learnt from the training data by maximizing the log-likelihood function using GPML [20].

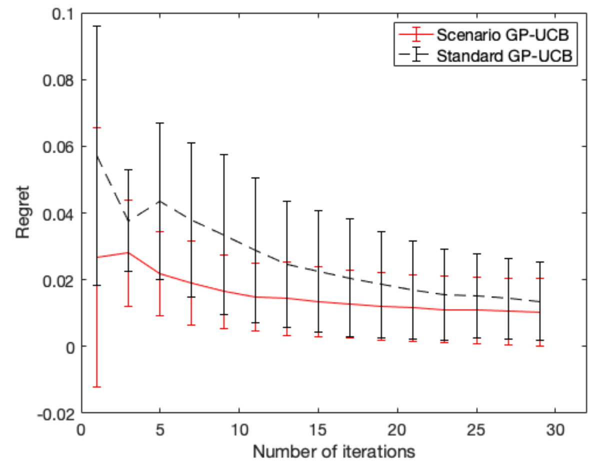

Fig. 1. Numerical evolution of the regret under re-draw using the Scenario UCB (Algorithm 1) and a baseline version of UCB.

The environment in this example is assumed to be a unit square. The start location S is $(0.25, 0.25)$, with 32 neighboring nodes. The destination is $(1,1)$, and the edge costs W_i are set to be equal to the Euclidean distance between S and i. The uncertainty δ comprises the set of random variables $\delta_{j,k}$ for each edge of the graph leading to the actual edge cost given by $W_{j,k} + \delta_{j,k}$. Given a neighboring node i of S, the function $V(x_i, \delta)$ is assumed to be the (negative of the) Euclidean length of the shortest path from x_i to the destination under this uncertainty model, using the standard Dijkstra algorithm.

The numerical value of the scenario regret under redraw (Definition 2.1) are reported in Figure 1. In this experiment, $\delta_{j,k} \sim$ Uniform$[0, 0.05]$ and $N = 25$ scenarios were used. The value of the process noise variance is $\sigma = 0.01$. As a baseline, we consider the standard UCB algorithm [19] applied to the function $F(\cdot, \bar{\delta}_N)$, where $\bar{\delta}_N := \frac{1}{N}\sum_{i=1}^N \delta_i$. The result is reported in Figure 1.

B. Minimizing Travel Time to Destination

This application involves applying the method to traffic data collected from Google Maps. In this experiment, we fixed the start point to New York City and destination point to Upper Manhattan and selected multiple intermediate points out of a given set as described in Table I. In this problem, we denote the distance of each neighboring location from the start node as the optimization variable x and the time to reach to the destination from that neighbor as the output y. Akin to the previous problem, we use the squared exponential kernel and train its hyperparameters using a similar approach.

We now report the scenario regret under redraw from Definition 2.1, $\forall t \geq 1$. We provide a comparison with our proposed scenario UCB approach from Algorithm 1 and the baseline comprising the standard UCB algorithm [19] applied to the function $F(\cdot, \bar{\delta}_N)$, where $\bar{\delta}_N := \frac{1}{N}\sum_{i=1}^N \delta_i$.

TABLE I

DATA FROM GOOGLE MAPS TO GO FROM NEW YORK (ZIP: 10001) TO UPPER MANHATTAN (COLLECTED ON 07/25/2019 AT 10:50AM EDT.

Location id	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Distance	1.5	1.5	1.0	1.2	1.0	0.8	0.5	0.7	0.4	0.4	1.0	1.8	1.7	1.2	0.7
Travel Time	39	41	41	38	34	26	28	36	33	28	37	33	41	39	33

The results are summarized in Figure 2. In this experiment, $\delta_{j,k} \sim \text{Uniform}[0, 0.5]$ and $N = 40$ scenarios were used.

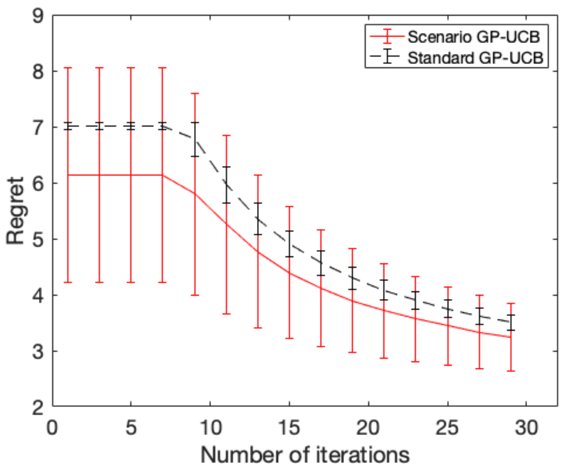

Fig. 2. Numerical comparison of the regret using the proposed scenario UCB approach (Algorithm 1) and using the standard GP-UCB for the path planning example from Table I.

Thus, in both case studies, we observe that *the scenario approach consistently provides a lower regret under redraw compared to a baseline comprising the standard UCB algorithm applied to the average value of the uncertainty*.

V. CONCLUSION AND FUTURE DIRECTIONS

We applied the scenario approach to the problem of robust motion planning when the cost-to-go is to be computed out of a simulation, in a bandit setting. The cost-to-go is assumed to be modeled as a GP for every realization of an uncertain parameter. We developed a scenario approach in which we draw fixed independent samples of the uncertain parameter and introduced a notion of regret under re-draw. We formalized a variant of a scenario-based iterative algorithm using the UCB of multiple scenarios to decide at what future location to evaluate the cost-to-go. We characterized a high probability upper bound on the regret under re-draw for any finite number of iterations of the algorithm and further characterized conditions under which the regret, with high probability, tends to zero asymptotically with the number of iterations. Finally, we supplemented our analysis with numerical results on two case studies in path planning.

Future directions include improved and tighter bounds on the high probability guarantees, and extensions to multi-agent robust motion planning problems.

REFERENCES

[1] C. E. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[2] J. Umlauft, A. Lederer, T. Beckers, and S. Hirche, "Real-time uncertainty decomposition for online learning control," *arXiv preprint arXiv:2010.02613*, 2020.

[3] C. Fulgenzi, C. Tay, A. Spalanzani, and C. Laugier, "Probabilistic navigation in dynamic environment using rapidly-exploring random trees and Gaussian processes," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2008, pp. 1056–1062.

[4] M. P. Deisenroth, D. Fox, and C. E. Rasmussen, "Gaussian processes for data-efficient learning in robotics and control," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 2, pp. 408–423, 2013.

[5] M. Mukadam, X. Yan, and B. Boots, "Gaussian process motion planning," in *IEEE International Conference on Robotics and Automation*, 2016, pp. 9–15.

[6] M. Mukadam, J. Dong, X. Yan, F. Dellaert, and B. Boots, "Continuous-time Gaussian process motion planning via probabilistic inference," *The International Journal of Robotics Research*, vol. 37, no. 11, pp. 1319–1340, 2018.

[7] F. Berkenkamp, A. P. Schoellig, and A. Krause, "Safe controller optimization for quadrotors with Gaussian processes," in *IEEE International Conference on Robotics and Automation*, 2016, pp. 491–496.

[8] T. Koller, F. Berkenkamp, M. Turchetta, and A. Krause, "Learning-based model predictive control for safe exploration," in *IEEE Conference on Decision and Control*, 2018, pp. 6059–6066.

[9] B. D. Luders, S. Karaman, and J. P. How, "Robust sampling-based motion planning with asymptotic optimality guarantees," in *AIAA Guidance, Navigation, and Control (GNC) Conference*, 2013, p. 5097.

[10] V. Indelman, "Towards cooperative multi-robot belief space planning in unknown environments," in *Robotics Research*. Springer, 2018, pp. 441–457.

[11] C. B. Browne, E. Powley, D. Whitehouse, S. M. Lucas, P. I. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakis, and S. Colton, "A survey of Monte Carlo tree search methods," *IEEE Trans. on Comp. Intell. and AI in games*, vol. 4, no. 1, pp. 1–43, 2012.

[12] S. Edelkamp, M. Gath, C. Greulich, M. Humann, O. Herzog, and M. Lawo, "Monte-Carlo tree search for logistics," in *Commercial Transport*. Springer, 2016, pp. 427–440.

[13] D. Lenz, T. Kessler, and A. Knoll, "Tactical cooperative planning for autonomous highway driving using Monte-Carlo Tree Search," in *2016 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2016, pp. 447–453.

[14] B. Hefferan, O. M. Cliff, and R. Fitch, "Adversarial patrolling with reactive point processes," in *Proceedings of the ARAA Australasian Conf. on Robotics and Automation (ARAA, 2016)*, 2016, pp. 39–46.

[15] G. Best, O. M. Cliff, T. Patten, R. R. Mettu, and R. Fitch, "Dec-MCTS: decentralized planning for multi-robot active perception," *Int. Jour. of Robotics Research*, vol. 38, no. 2-3, pp. 316–337, 2019.

[16] G. C. Calafiore, F. Dabbene, and R. Tempo, "Research on probabilistic methods for control system design," *Automatica*, vol. 47, no. 7, pp. 1279–1293, 2011.

[17] M. C. Campi, S. Garatti, and M. Prandini, "The scenario approach for systems and control design," *Annual Reviews in Control*, vol. 33, no. 2, pp. 149–157, 2009.

[18] T. Alamo, R. Tempo, A. Luque, and D. R. Ramirez, "Randomized methods for design of uncertain systems: Sample complexity and sequential algorithms," *Automatica*, vol. 52, pp. 160–172, Feb 2015.

[19] N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger, "Information-theoretic regret bounds for gaussian process optimization in the bandit setting," *IEEE Transactions on Information Theory*, vol. 58, no. 5, pp. 3250–3265, 2012.

[20] C. E. Rasmussen and H. Nickisch, "Documentation for GPML Matlab Code version 4.2." [Online]. Available: http://www.gaussianprocess.org/gpml/code/matlab/doc/

APPENDIX: MATHEMATICAL PROOFS

Consider the following reformulation of Problem (3).

$$\max_{x \in X, \tau \in \mathbb{R}} \tau \quad \text{subject to} \quad F(x, \delta) \geq \tau, \; \forall \, \delta \in \Delta. \tag{7}$$

For finite, discrete domains X and Δ with cardinalities $|X|$ and $|\Delta|$, respectively, it follows that (τ, x) in (7) belongs to a discrete set with cardinality $|X| \times |\Delta|$.

An immediate consequence of Theorem 3 from [18] when applied to Problem (4) is the following.

Corollary 5.1: Given $\eta, \zeta \in (0,1)$, if

$$N \geq \frac{1}{\eta}(\ln|X| + \ln|\Delta| - \ln\zeta),$$

then, with probability[1] at least $1-\zeta$, for every $\hat{x} \in X$,

$$\mathbb{P}_\delta\Big\{F(\hat{x},\delta) \leq \min_{i\in\{1,\ldots,N\}} F(\hat{x},\delta_i)\Big\} \leq \eta.$$

A. Proof of Proposition 3.1

From Assumption 2.3 and from Corollary 5.1, we conclude that with probability at least $1-\zeta$, for every $x \in X$

$$\mathbb{P}_{\delta_{N+1}}\{F(x,\delta_{N+1}) \geq \min_{\delta\in\Delta_N} F(x,\delta)\} \geq 1-\eta.$$

This is equivalent to

$$\mathbb{P}_{\delta_{N+1}}\Big\{\min_{\delta\in\Delta_N} F(x,\delta) = \min_{\delta\in\Delta_{N+1}} F(x,\delta)\Big\} \geq 1-\eta. \quad (8)$$

Since $\min_{\delta\in\Delta_N} F(x,\delta) = \min_{\delta\in\Delta_{N+1}} F(x,\delta), \forall x \in X$

$$\Rightarrow \max_{x\in X}\min_{\delta\in\Delta_N} F(x,\delta) = \max_{x\in X}\min_{\delta\in\Delta_{N+1}} F(x,\delta),$$

we conclude from (8) and from Definition 2.1 that

$$\mathbb{P}_{\delta_{N+1}}\big\{R_T(\Delta_N \cup \delta_{N+1}) = R_T(\Delta_N)\big\} \geq 1-\eta,$$

with probability at least $1-\zeta$.

B. Proof of Theorem 3.1

At the t-th iteration, define the instantaneous regret r as

$$r(x_t,\delta_{i_t}) := \max_{x\in X}\min_{\delta\in\Delta_N} F(x,\delta) - F(x_t,\delta_{i_t}).$$

We begin with establishing the following bound on $r(x_t,\delta_{i_t})$.

Lemma 5.1 (Instantaneous regret bound): For any $t \geq 1$, the instantaneous regret satisfies,

$$\mathbb{P}\{r(x_t,\delta_{i_t}) \leq 2\sqrt{\beta_t}\sigma_{t-1}^{i_t}(x_t)|\mathcal{D}_{t-1}\} \geq 1 - 2|X|\mathrm{e}^{-\beta_t/2},$$

where the underlying random variable corresponds to the posterior distribution of the GP $F(x,\delta_{i_t})$ conditioned on $\mathcal{D}_{t-1}$ defined in Algorithm 1.

Proof: Assumption 2.2 implies that the posterior of $F(x,\delta_i)$ conditioned on $\mathcal{D}_{t-1}$ is Gaussian with mean $\mu_{t-1}^i(x)$ and standard deviation $\sigma_{t-1}^i(x)$. For any t and for any i:

$$F(x,\delta_i)|\mathcal{D}_{t-1} \sim \mathcal{N}(\mu_{t-1}^i(x), (\sigma_{t-1}^i(x))^2).$$

From the concentration of measures inequality for Gaussian random variables, for any i, we have

$$\mathbb{P}\{|F(x,\delta_i) - \mu_{t-1}^i(x)| > \sqrt{\beta_t}\sigma_{t-1}^i(x)|\mathcal{D}_{t-1}\} \leq 2\mathrm{e}^{-\beta_t/2}.$$

Using the union bound, we have for any i and for all $x \in X$,

$$\mathbb{P}\{|F(x,\delta_i)-\mu_{t-1}^i(x)| > \sqrt{\beta_t}\sigma_{t-1}^i(x)|\mathcal{D}_{t-1}\} \leq 2|X|\mathrm{e}^{-\beta_t/2}.$$

Now consider

$$\begin{aligned} r(x_t,\delta_{i_t}) &= F(x^*[N],\delta^*[n]) - F(x_t,\delta_{i_t}) \\ &\leq F(x^*[N],\delta_{i_t}) - F(x_t,\delta_{i_t}), \end{aligned}$$

where $(x^*[N],\delta^*[N])$ is a solution to the problem,

$$\max_{x\in X}\min_{\delta\in\Delta_N} F(x,\delta).$$

Therefore, with probability at least $1-2|X|\mathrm{e}^{-\beta_t/2}$,

$$\begin{aligned} r(x_t,\delta_{i_t}) &\leq \mu_{t-1}^{i_t}(x^*[N]) + \sqrt{\beta_t}\sigma_{t-1}^{i_t}(x^*[N]) \\ &\qquad - \mu_{t-1}^{i_t}(x_t) + \sqrt{\beta_t}\sigma_{t-1}^{i_t}(x_t) \\ &\leq \mu_{t-1}^{i_t}(x_t) + \sqrt{\beta_t}\sigma_{t-1}^{i_t}(x_t) - \mu_{t-1}^{i_t}(x_t) + \sqrt{\beta_t}\sigma_{t-1}^{i_t}(x_t) \\ &= 2\sqrt{\beta_t}\sigma_{t-1}^{i_t}(x_t), \end{aligned}$$

where the second inequality follows from the definition of x_t in step 6 of Algorithm 1. This completes the proof. □

Proof of Theorem 3.1: With $\beta_t = 2\log(|X|\pi^2t^2/(3\epsilon))$, it follows from Lemma 5.1 that

$$\mathbb{P}\{r(x_t,\delta_{i_t}) > 2\sqrt{\beta_t}\sigma_{t-1}^{i_t}(x_t)\} \leq 6\epsilon/\pi^2t^2.$$

By applying union bound,

$$\mathbb{P}\{r(x_t,\delta_{i_t}) > 2\sqrt{\beta_t}\sigma_{t-1}^{i_t}(x_t), \text{ for some } t \in \mathbb{N}\} \leq \epsilon.$$

Now following steps similar to those in [19], with probability at least $1-\epsilon$,

$$\begin{aligned} \sum_{t=1}^T r(x_t,\delta_{i_t})^2 &\leq \sum_{t=1}^T 4\beta_t(\sigma_{t-1}^{i_t}(x_t))^2 \\ &\leq 4\beta_T\sum_{t=1}^T\sum_{i=1}^N(\sigma_{t-1}^i(x_t))^2\mathbf{1}(i_t=i) \\ &\leq \frac{4\beta_T}{\log(1+1/\rho^2)}\sum_{i=1}^N\sum_{t=1}^T\log\Big(1+\frac{(\sigma_{t-1}^i(x_t))^2}{\rho^2}\Big), \end{aligned}$$

where the last inequality follows since $s^2 \leq \log(1+s^2)/(\rho^2\log(1+1/\rho^2))$ for each $s \in [0,1/\rho^2]$ [19].

Let $\gamma_T^i = \frac{1}{2}\sum_{t=1}^T\log\big(1+\frac{(\sigma_{t-1}^i(x_t))^2}{\rho^2}\big)$. It follows from [19, Theorems 1 & 4] (with the choice of $T_* = 1$ therein),

$$\gamma_T^i \in O\Big(\rho^{-2}\big(T\sum_{j=2}^{|X|}\hat{\lambda}_j^i + \log(T\sum_{j=1}^{|X|}\hat{\lambda}_j^i)\big)\Big).$$

From Cauchy-Schwartz inequality, it follows that

$$R_T^{\text{re-draw}}(\Delta_N) \leq \frac{1}{T}\sqrt{T\sum_{t=1}^T r(x_t,\delta_{i_t})^2} \leq \sqrt{\frac{8\beta_T\gamma_T}{T\log(1+1/\rho^2)}},$$

which establishes the first claim of this theorem. To establish the second claim, observe that in the regime of $T \geq |X|$, with the choice of $T_* = |X|$ in [19][Theorems 1 & 4], we have

$$\gamma_T^i \in O\Big(\frac{|X|}{\rho^2}\log\Big(T\sum_{j=1}^{|X|}\hat{\lambda}_j^i\Big)\Big),$$

which leads to the second claim. □

[1]This probability is with respect to the multi-sample $\delta_1,\ldots,\delta_N$.