Ergodic Exploration of Distributed Information

Lauren M. Miller, Yonatan Silverman, Malcolm A. MacIver, and Todd D. Murphey

***Abstract*—This paper presents an active search trajectory synthesis technique for autonomous mobile robots with nonlinear measurements and dynamics. The presented approach uses the ergodicity of a planned trajectory with respect to an expected information density map to close the loop during search. The ergodic control algorithm does not rely on discretization of the search or action spaces and is well posed for coverage with respect to the expected information density whether the information is diffuse or localized, thus trading off between exploration and exploitation in a single-objective function. As a demonstration, we use a robotic electrolocation platform to estimate location and size parameters describing static targets in an underwater environment. Our results demonstrate that the ergodic exploration of distributed information algorithm outperforms commonly used information-oriented controllers, particularly when distractions are present.**

***Index Terms*—Biologically inspired robots, information-driven sensor planning, motion control, search problems.**

I. Introduction

In the context of exploration, ergodic trajectory optimization computes control laws that drive a dynamic system along trajectories such that the amount of time spent in regions of the state space is proportional to the expected information gain in those regions. Using ergodicity as a metric encodes both exploration and exploitation—both the need for nonmyopic search when variance is high and convexity is lost, as well as myopic search when variance is low and the problem is convex. By encoding these needs into a metric [1], generalization to nonlinear dynamics is possible using tools from optimal control. We show here that different dynamical systems can achieve nearly identical estimation performance using ergodic exploration of distributed information (EEDI).

The SensorPod robot [see Fig. 1(a)], which we use as a motivating example and an experimental platform in Section V, measures disturbances in a self-generated weak electric field. This sensing modality, referred to as electrolocation, is inspired by a type of freshwater tropical fish (see Fig. 1(b) and [2]–[4]) and relies on the coupled emission and detection of a weak electric field. Electrolocation is ideally suited for low-velocity mobile vehicles operating in dark or cluttered environments [4]–[6]. The sensing range for electrolocation is small, however, so the fish or robot must be relatively close to an object to localize it. In addition, as the sensors are rigid with respect to the body, the movement of those sensors involves the dynamics of the entire robot. As we will see in Section IV, the measurement model for electrolocation is also highly nonlinear, and the dynamics of both biological fish and underwater robots are generally nonlinear. Consideration of sensor physics and robot dynamics when planning exploration strategies is, therefore, particularly important. The same applies to many near-field sensors such as tactile sensors, ultra-shortwave sonar, and most underwater image sensors (e.g., [7]). Experiments carried out using the SensorPod robot demonstrate that the EEDI algorithm is successful in several challenging search scenarios where other algorithms fail.

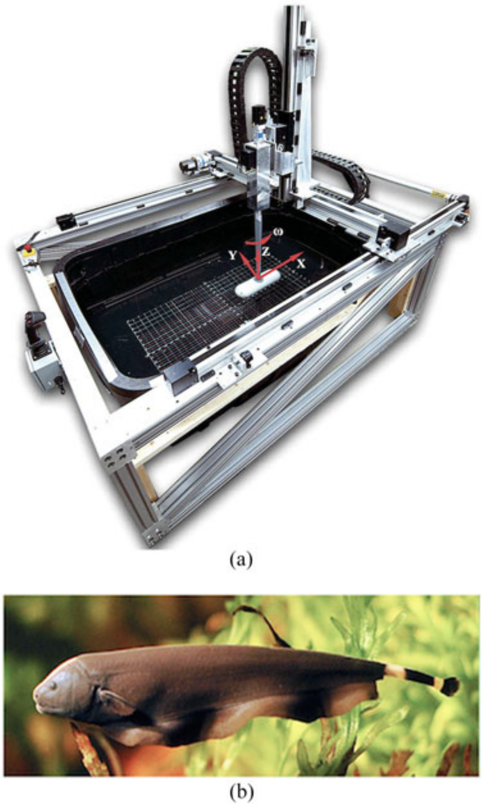


Fig. 1. SensorPod (a) uses a sensing modality inspired by weakly electric fish such as the black ghost knifefish (b). The SensorPod is mounted on a 4-DOF gantry and submerged within a 1.8 m × 2.4 m × 0.9 m (l, w, h) tank (see multimedia attachment).

Manuscript received April 18, 2015; revised September 20, 2015; accepted November 6, 2015. Date of publication December 9, 2015; date of current version February 3, 2016. This paper was recommended for publication by Associate Editor P. R. Giordano and Editor D. Fox upon evaluation of the reviewers' comments. This work was supported by Army Research Office grant W911NF-14-1-0461, National Science Foundation grants IOB-0846032, CMMI-0941674, and CMMI-1334609, and Office of Naval Research Small Business Technology Transfer grants N00014-09-M-0306 and N00014-10-C0420 to M. A. MacIver.

L. M. Miller, Y. Silverman, and T. D. Murphey are with the Department of Mechanical Engineering, Northwestern University, Evanston, IL 60208 USA (e-mail: LMiller@u.northwestern.edu; yoni.h.silverman@gmail.com; t-murphey@northwestern.edu).

M. A. MacIver is with the Department of Mechanical Engineering and the Department of Biomedical Engineering, Northwestern University, Evanston, IL 60208 USA (e-mail: maciver@northwestern.edu).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org. The material consists of video that includes an animated description of the measurement model, overview of the Sensorpod Robot, and the EEDI algorithm applied to search for stationary targets.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TRO.2015.2500441

The contributions of this paper can be summarized as follows:

1) application of ergodic exploration for general, nonlinear, deterministic control systems to provide closed-loop coverage with respect to the evolving expected information density (EID);
2) validation of ergodic search in an experimental and simulated underwater sensing setting. We demonstrate both that ergodic search performs as well as alternative algorithms in nominal scenarios, and that ergodic search outperforms alternatives when distractors are present.

Section II begins with a discussion of related work. Ergodicity as an objective for active sensing is presented in Section III, including the benefits and distinguishing features of ergodic trajectory optimization. Section III-B includes an overview of ergodic trajectory optimization. In Section IV, we describe the SensorPod experimental platform and nonlinear measurement model and introduce the stationary target localization task used to demonstrate EEDI. We also discuss the components of closed-loop EEDI for target localization using the SensorPod in Section IV. In Section V, we present data from multiple estimation scenarios, including comparison with several alternative algorithms, and closed-loop EEDI implementation using different dynamic models for the SensorPod. We also include a multimedia video attachment with an extended description of the SensorPod platform and measurement model used in Sections IV and V and an animated overview of the steps of the EEDI algorithm for this system.

II. Motivation and Related Work

The ability to actively explore and respond to uncertain scenarios is critical in enabling robots to function autonomously. In this paper, we examine the problem of control design for mobile sensors carrying out active sensing tasks. Active sensing [8], [9] or sensor path planning [10] refers to control of sensor parameters, such as position, to acquire information or reduce uncertainty. Applications include prioritized decision making during search and rescue [11], [12], inspection for flaws [13], mine detection [10], object recognition/classification [14]–[16], next-best-view problems for vision systems [17]–[19], and environmental modeling/field estimation [20]–[22]. Planning for search/exploration is challenging as the planning step necessarily depends not only on the sensor being used but on the quantity being estimated, such as target location versus target size. Methods for representing and updating the estimate and associated uncertainty—the belief state—and a way of using the belief state to determine expected information are, therefore, required.

Fig. 2 illustrates the high-level components for a general estimation or mapping algorithm that iteratively collects sensor measurements, updates an expected information map, and decides how to acquire further measurements based on the information map. In this section, we touch on related work for components A–C, although the differentiating feature of the EEDI algorithm is the way in which control decisions are made based on the expected information (step C in Fig. 2). The advantages of EEDI are discussed in Section III.

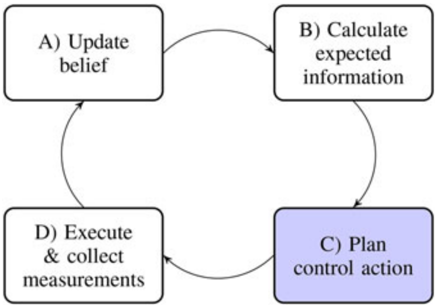

A) Update belief
B) Calculate expected information
C) Plan control action
D) Execute & collect measurements

Fig. 2. Illustration of the necessary components for a general closed-loop information-based sensing algorithm. Our primary contribution in this paper is using ergodic trajectory optimization for estimation (step C). We demonstrate implementation of closed-loop estimation for a particular sensing task using the SensorPod robot [see Fig. 1(a)], where the sensing task motivates choice of steps A, B, D. Section II discusses alternative choices for steps A through C.

A. Representing the Belief State

The best choice for representing and updating the belief state for a given application will depend on robot dynamics, sensor physics, and the estimation task (modeling a field versus target localization). Designing appropriate representations for active sensing is a well-studied area of research. For many applications, such as active sensing for localization, parametric filters (e.g., the extended Kalman filter) [23]–[26] may be used. When the posterior is not expected to be approximately Gaussian, nonparametric filters, e.g., Bayesian filters [27], [28], histogram filters [29], or particle filters [30]–[32] are often used. Mapping applications often use occupancy grids [33], [34] or coverage maps [29], and much of the most recent work utilizes Gaussian processes to represent spatially varying phenomena or higher dimensional belief spaces, and the associated uncertainty [20], [21], [35]–[38]. For the experimental work presented in this paper using the SensorPod robot, we use a Bayesian filter as it is appropriate for general (non-Gaussian) PDFs, sufficient for representing stationary estimation objectives, and allows us to take into account sensor physics and uncertainty in the estimate (see Section IV). The differentiating feature of the EEDI algorithm however—the ergodic trajectory optimization—does not depend on the choice of belief representation, so long as the choice enables calculation of an EID map.

B. Calculating Expected Measurement Utility

For a given sensing task and belief state, not all measurements are equally informative. The quality of a measurement depends on the sensor and may be distance, orientation, or motion dependent. To ensure useful measurements are obtained given realistic time or energy restrictions, sensing strategies for mobile sensors should seek to optimize measurement quality [39], [40]. In some cases, it is sufficient to consider only sensor field of view (i.e., useful measurements can be obtained anywhere within a given distance from a target), often called “geometric sensing” [32], [41]–[43]. In many scenarios—if the search domain is significantly larger than the sensor range—a geometric sensing approach is sufficient. Many sensors, however, have sensitivity characteristics within the range threshold that affect sensing efficacy. Infrared range sensors, for example, have a maximum sensitivity region [44], and cameras have an optimal orientation and focal length [45].

There are several different entropy-based measures from information theory and optimal experiment design that can be used to predict expected information gain prior to collecting measurements. Shannon entropy has been used to measure uncertainty in estimation problems [9], [15], [17], [19], as well as entropy-related metrics including Renyi Divergence [8], [24], mutual information, [11], [14], [31], [35], [42], [46]–[48], entropy reduction or information gain maximization [13], [49]. In our problem formulation, we use Fisher information [50]–[53] to predict measurement utility. Often used in maximum likelihood estimation, Fisher information quantifies the ability of a random variable, in our case a measurement, to estimate an unknown parameter [50], [51], [54]. Fisher information predicts that the locations where the ratio of the derivative of the expected signal to the variance of the noise is high will give more salient data (see Appendix B and the multimedia attachment) and thus will be more useful for estimation.

In this paper, the Bayesian update mentioned in Section II-A and the use of Fisher information to formulate an information map are tools that allow us to close the loop on ergodic control (update the map, step A in Fig. 2), in a way that is appropriate for the experimental platform and search objective (see Appendix A). The Bayesian update and the Fisher information matter only in that they allow us to create a map of expected information for the type of parameter estimation problems presented in the examples in Section V. Ergodic exploration could, however, be performed over the expected information calculated using different methods of representing the belief and expected information, and for different applications such as those mentioned in II-A.

C. Control for Information Acquisition

In general, the problem of exhaustively searching for an optimally informative solution over sensor state space and belief state is a computationally intensive process, as it is necessary to calculate an expectation over both the belief and the set of candidate control actions [24], [35], [46], [55]. Many algorithms, therefore, rely on decomposing/discretizing the search space, the action space, or both, and locally selecting the optimal sensing action myopically (selecting only the optimal next configuration or control input) [8], [25]. The expected information gain can, for example, be locally optimized by selecting a control action based on the gradient of the expected information [30], [32], [33], [47]. As opposed to local information maximization, a sensor can be controlled to move to that state which maximizes the expected information globally over a bounded workspace [17], [26], [28], [50], [56]. Such global information maximizing strategies are generally less sensitive to local minima than local or gradient based strategies, but can result in longer, less efficient trajectories when performed sequentially [29], [31]. While myopic information maximizing strategies have an advantage in terms of computational tractability, they are typically applied to situations where the sensor dynamics are not considered [17], [26], [28], [50], [56] and even the global strategies are likely to suffer when uncertainty is high and information diffuse (as argued in [29], [37], and [57], when discussing replanning periods), as we will see in Section V.

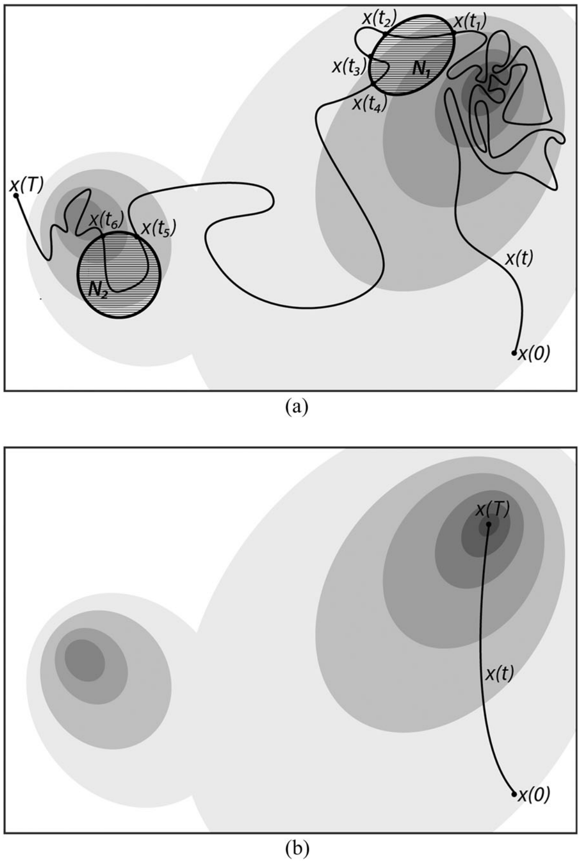
To avoid sensitivity of single-step optimization methods to local optima, methods of planning control actions over longer time horizons—nonmyopic approaches—are often used. A great deal of research in search strategies points out that the most general approach to solving for nonmyopic control signals would involve solving a dynamic programming problem [20], [35], [37], which is generally computationally intensive. Instead, various heuristics are used to approximate the dynamic programming solution [20], [29], [36], [37]. Variants of commonly used sampling-based motion planners for maximizing the expected information over a path for a mobile sensor have also been applied to sensor path-planning problems [10], [23], [24], [49], [58], [59].

Search-based approaches are often not suitable for systems with dynamic constraints; although they can be coupled with low-level (e.g., feedback control) planners [48], [60], or dynamics can be encoded into the cost of connecting nodes in a search graph (“steering” functions) [58], solutions are not guaranteed to be optimal even in a local sense—both in terms of the dynamics and the information—without introducing appropriate heuristics [20], [29], [36], [37]. As we will see in Section V-G, one of the advantages of EEDI is that it naturally applies to general nonlinear systems. We take advantage of trajectory optimization techniques, locally solving for a solution to the dynamic programming problem—assuming that the current state of the system is approximately known.

Use of an ergodic metric for determining optimal control strategies was originally presented in [1] for a nonuniform coverage problem. The strategy in [1] involves discretizing the exploration time and solving for the optimal control input at each time step that maximizes the rate of decrease of the ergodic metric. A similar method is employed in [61], using a Mix Norm for coverage on Riemannian manifolds. While our objective function includes the same metric as [1], the optimal control problem and applications are different, notably in that we compute the ergodic trajectory for the entire planning period T, and apply it to a changing belief state. Additionally, the feedback controllers derived in [1] are specific to linear first- or second-order integrator systems, whereas our method applies to general nonlinear dynamic systems.

III. Ergodic Optimal Control

Ergodic theory relates the time-averaged behavior of a system to the space of all possible states of the system and is primarily used in the study of fluid mixing and communication. We use ergodicity to compare the statistics of a search trajectory to a map of EID. The idea is that an efficient exploration strategy—the path followed by a robot—should spend more time exploring regions of space with higher expected information, where useful measurements are most likely to be found. The robot



(a)

(b)

Fig. 3. Two candidate trajectories $x(t)$ for exploring the EID (depicted as level sets) are plotted in (a) and (b), both from $t = 0$ to $t = T$. Ergodic control provides a way of designing trajectories that spend time in areas proportional to how useful potential measurements are likely to be in those areas (a). This is in contrast with many alternative algorithms, which directly maximize integrated information gain over the trajectory based on the current best estimate, as in (b). As illustrated in (a), a trajectory $x(t)$ is *ergodic* with respect to the PDF (level sets) if the percentage of time spent in any subset N from $t = 0$ to $t = T$ is equal to the measure of N; this condition must hold for all possible subsets.

should not, however, only visit the highest information region [see Fig. 3(b)], but distribute the amount of time spent searching proportional to the overall EID [see Fig. 3(a)]. This is the key distinction between using ergodicity as an objective and previous work in active sensing (e.g., information maximization); the ergodic metric encodes the idea that, unless the EID is a delta function, measurements should be *distributed* among regions of high expected information. Information maximizing strategies (that are also nonmyopic) otherwise require heuristics in order to force subsequent measurements away from previously sampled regions so as not to only sample the information maxima.

As mentioned in Section II, many commonly used algorithms for active sensing, e.g., [25], [28], [33], [62], involve a version of the type of behavior illustrated in Fig. 3(b), iteratively updating the EID and maximizing information gain based on the current best estimate, whether or not that estimate is correct. While computationally efficient, globally information maximizing approaches are likely to fail if the current estimate of the EID is wrong. In Section V, for example, we show that even when the information map is updated while calculating the information maximizing control, the estimate may get trapped in a local maxima, e.g., when there is a distractor object that is similar but not exactly the same as the target object.

Many sampling-based algorithms for information gathering, therefore, rely on heuristics related to assuming submodularity between measurements, e.g., assuming no additional information will be obtained from a point once it has already been observed [23], [35], [58]. This assumption forces subsequent measurements away from previously sampled regions so as not to only sample the information maxima. As another way to distribute measurements, many nonmyopic strategies select a set of waypoints based on the expected information and drive the system through those waypoints using search-based algorithms [38], [41], [42], [48], [49], [57], [60]. Such approaches result in a predefined sequence that may or may not be compatible with the system dynamics. If the ordering of the waypoints is not predefined, target-based search algorithms may require heuristics to avoid the combinatoric complexity of a traveling salesman problem [63], [64]. In some cases, search algorithms are not well-posed unless both an initial and final (goal) position are specified [20], [42], which is not generally the case when the objective is exploration.

Ergodic control enables how a robot searches a space to depend directly on the dynamics and is well posed for arbitrary dynamical systems. In the case of nonlinear dynamics and nontrivial control synthesis, encoding the search ergodically allows control synthesis to be solved directly in terms of the metric, instead of in a hierarchy of problems (starting with target selection and separately solving for the control that acquires those targets, for example, [38], [41], [42], [49], [57], [60]). In ergodic trajectory optimization, the distribution of samples results not from introducing heuristics into the trajectory optimization, but from encoding the statistics of the trajectory and the information map directly in the objective. Using methods from optimal control, we directly calculate trajectories that are ergodic with respect to a given information density [65], [66]. It is noteworthy, however, that even if one wants to add waypoints to a search objective, ergodic search is an effective means to drive the system to each waypoint in a dynamically admissible manner (by replacing each waypoint with a low-variance density function, thus avoiding the traveling salesman problem). Further, ergodic control can be thought of as a way to generate a continuum of dynamically compatible waypoints; it is similar to [42], [49], and [57], but allows the number of waypoints to go to ∞, making the control synthesis more tractable for a broad array of systems.

Many active sensing algorithms are formulated to either prioritize exploitation (choosing locations based on the current belief state) or exploration (choosing locations that reduce uncertainty in the belief state); they are best suited for greedy, reactive sampling, requiring a prior estimate [23], or for coverage [35], [67], [68]. Algorithms that balance both exploration and exploitation typically involve encoding the two objectives separately and switching between them based on some condition on the estimate, [37], [69], or defining a (potentially arbitrary) weighted cost function that balances the tradeoff between the two objectives [22], [36], [38], [60]. Using ergodicity as an objective results in an algorithm that is suitable for both exploration-prioritizing coverage sampling or exploitation-prioritizing "hotspot" sampling, without modification (policy switching or user-defined weighted objectives [22], [36], [37], [60], [69]). Moreover, the ergodic metric can be used in combination with other metrics, like a tracking cost or a terminal cost, but does not require either to be well posed.

A. Measuring Ergodicity

We use the *distance from ergodicity* between the time-averaged trajectory and the EID as a metric to be minimized. We assume a bounded n-dimensional workspace (the search domain) $X \subset \mathbb{R}^n$ defined as $[0, L_1] \times [0, L_2] ... \times [0, L_n]$. We define $\boldsymbol{x}(t)$ as the sensor trajectory in workspace coordinates and the density function representing the EID as $\text{EID}(\boldsymbol{x})$.

The spatial statistics of a trajectory $\boldsymbol{x}(t)$ are quantified by the percentage of time spent in each region of the workspace

$$C(\boldsymbol{x}) = \frac{1}{T} \int_0^T \delta\left[\boldsymbol{x} - \boldsymbol{x}(t))\right] dt \tag{1}$$

where δ is the Dirac delta [1]. The goal is to drive the spatial statistics of a trajectory $\boldsymbol{x}(t)$ to match those of the distribution $\text{EID}(\boldsymbol{x})$; this requires choice of a norm on the difference between the distributions $\text{EID}(\boldsymbol{x})$ and $C(\boldsymbol{x})$. We quantify the difference between the distributions, i.e., the distance from ergodicity, using the sum of the weighted squared distance between the Fourier coefficients ϕ_k of the EID, and the coefficients c_k of distribution representing the time-averaged trajectory.[1] The ergodic metric will be defined as $\mathcal{E}$, as follows:

$$\mathcal{E}(\boldsymbol{x}(t)) = \sum_{\boldsymbol{k}=0\in\mathbb{Z}^n}^{\boldsymbol{K}\in\mathbb{Z}^n} \Lambda_k \left[c_{\boldsymbol{k}}(\boldsymbol{x}(t)) - \phi_{\boldsymbol{k}}\right]^2 \tag{2}$$

where $\boldsymbol{K}$ is the number of coefficients calculated along each of the n dimensions, and $\boldsymbol{k}$ is a multi-index $(k_1, k_2, ..., k_n)$. Following [1], $\Lambda_{\boldsymbol{k}} = \frac{1}{(1+||\boldsymbol{k}||^2)^s}$ is a weight, where $s = \frac{n+1}{2}$, which places larger weight on lower frequency information.

Note that the notion of ergodicity used here does not strictly require the use of Fourier coefficients in constructing an objective function. The primary motivation in using the norm on the Fourier coefficients to formulate the ergodic objective is that it provides a metric that is differentiable with respect to the trajectory $\boldsymbol{x}(t)$. This particular formulation is not essential—any differentiable method of comparing the statistics of a desired EID to the spatial distribution generated by a trajectory will suffice, however finding such a method is nontrivial. The Kullback–Leibler divergence or Jensen–Shannon (J–S) divergence [21], for example, commonly used metrics on the distance between two distributions, are not differentiable with respect to the trajectory $\boldsymbol{x}(t)$.[2] On the other hand, by first decomposing both distributions into their Fourier coefficients, the inner product between the transform and the expression for the time-averaged distribution results in an objective that is differentiable with respect to the trajectory.

B. Trajectory Optimization

For a general, deterministic, dynamic model for a mobile sensor $\dot{\boldsymbol{x}}(t) = f(\boldsymbol{x}(t), \boldsymbol{u}(t))$, where $\boldsymbol{x} \in \mathbb{R}^m$ is the state and $\boldsymbol{u} \in \mathbb{R}^m$ the control, we can solve for a continuous trajectory that minimizes an objective function based on both the measure of the ergodicity of the trajectory with respect to the EID and the control effort, defined as

$$J(\boldsymbol{x}(t)) = \underbrace{\gamma\mathcal{E}[x(t)]}_{\text{ergodic cost}} + \underbrace{\int_0^T \frac{1}{2} u(\tau)^T R u(\tau)}_{\text{control effort}}. \tag{3}$$

In this equation, $\gamma \in \mathbb{R}$ and $R(\tau) \in \mathbb{R}^{m\times m}$ are arbitrary design parameters that affect the relative importance of minimizing the distance from ergodicity and the integrated control effort. The choice of ratio of γ to R plays the exact same role in ergodic control as it does in linear quadratic control and other methods of optimal control; the ratio determines the balance between the objective—in this case ergodicity—and the control cost of that objective. Just as in these other methods, changing the ratio will lead to trajectories that perform better or worse with either more or less control cost.

In [65], we show that minimization of (3) can be accomplished using an extension of trajectory optimization [70] and derive the necessary conditions for optimality. The extension of the projection-based trajectory optimization method from [70] is not trivial as the ergodic metric is not a Bolza problem; however, Miller and Murphey [65] prove that the first-order approximation of minimizing (3) subject to the dynamics $\dot{\boldsymbol{x}}(t) = f(\boldsymbol{x}(t), \boldsymbol{u}(t))$ is a Bolza problem and that trajectory optimization can be applied to the ergodic objective. The optimization does not require discretization of search space or control actions in space or time. While the long-time horizon optimization we use is more computationally expensive than the myopic, gradient-based approach in [1], each iteration of the optimization involves a calculation with known complexity. The EID map and ergodic objective function could, however, also be utilized within an alternative trajectory optimization framework (e.g., using sequential quadratic programming). Additionally, for the special case of $\dot{\boldsymbol{x}} = \boldsymbol{u}$, sample-based algorithms [58] may be able to produce locally optimal ergodic trajectories that are equivalent (in ergodicity) to the solution obtained using trajectory optimization methods; this would not, however, be the case for general nonlinear dynamics $\dot{\boldsymbol{x}}(t) = f(\boldsymbol{x}(t), \boldsymbol{u}(t))$.

Ergodic optimal control allows for the time of exploration to be considered as an explicit design variable. It can, of course, be

[1] The Fourier coefficients ϕ_k of the distribution $\Phi(\boldsymbol{x})$ are computed using an inner product, $\phi_{\boldsymbol{k}} = \int_X \phi(\boldsymbol{x}) F_{\boldsymbol{k}}(\boldsymbol{x}) d\boldsymbol{x}$, and the Fourier coefficients of the basis functions along a trajectory $\boldsymbol{x}(t)$, averaged over time, are calculated as $c_{\boldsymbol{k}}(\boldsymbol{x}(t)) = \frac{1}{T}\int_0^T F_{\boldsymbol{k}}(\boldsymbol{x}(t)) dt$, where T is the final time and F_k is a Fourier basis function.

[2] Due to the Dirac delta in (1), the J–S divergence ends up involving evaluating the EID along the trajectory $\boldsymbol{x}(t)$. In general we do not expect to have a closed form expression for the EID, so this metric is not differentiable in a way that permits trajectory optimization. Alternatively, replacing the Dirac delta in (1) with a differentiable approximation (e.g., a Gaussian) would expand the range of metrics on ergodicity, but would introduce additional computational expense of evaluating an N-dimensional integral when calculating the metric and its derivative.

of short duration or long duration, but our motivation is largely long duration. The idea is that one may want to execute a long exploration trajectory prior to replanning. The most straightforward motivation is that integrating measurements and updating the belief may be the more computationally expensive part of the search algorithm [23], [29], [31], [37]. Overly reactive/adaptive strategies—strategies that incorporate measurements as they are received—are also likely to perform poorly when the estimate uncertainty is high [29], [37], [57] or in the presence of (inevitable) modeling error. If, for example, the measurement uncertainty is not perfectly captured by the measurement model, the idealized Bayesian update can lead to overly reactive control responses. Instead, one may wish to take enough data such that the central limit theorem can be applied to the measurement model so that the measurement model is only anticipated to be applicable on average over the length of the exploratory motion [71]. Future work will involve exploring the effects of reducing the replanning horizon on the success of the estimation algorithm.

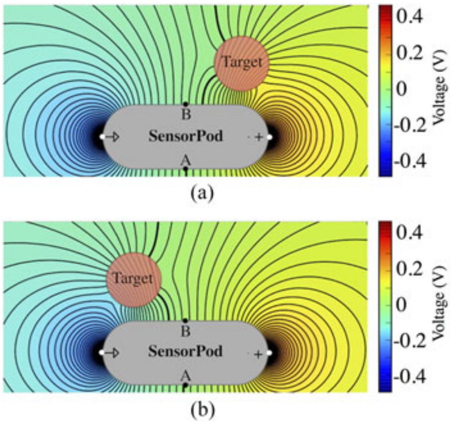
C. Assumptions: Ergodic Trajectory Optimization

Ergodic trajectory optimization requires a controllable motion model for the robot and an EID function defined over the sensor state space. The motion model can be nonlinear and/or dynamic, one of the primary benefits of a trajectory optimization approach. For this paper, we consider calculating search trajectories in one and two dimensions (although the sensor dynamics can be higher dimensional). The trajectory optimization method can be extended to search in higher dimensional search spaces such as $\mathbb{R}^3$ and $SE(2)$, so long as a Fourier transform [72] exists for the manifold [66]. We consider only uncertainty of static environmental parameters (e.g., fixed location and radius of an external target) assuming noisy measurements. We assume deterministic dynamics.

IV. Experimental Methods: Search for Stationary Targets Using the SensorPod Robot

Although ergodic trajectory optimization is general to sensing objectives with spatially distributed information, we describe an application where the belief representation and EID calculation (steps A, B, and D in Fig. 2) are chosen for active localization of stationary targets using the SensorPod robot. This allows us to experimentally test and validate a closed-loop version of the ergodic optimal control algorithm, described in Section III, against several established alternatives for planning control algorithms based on an information map.

Inspired by the electric fish, the SensorPod [see Fig. 1(a)] has two excitation electrodes that create an oscillating electric field. We use a single pair of voltage sensors—hence, a *1-D signal* —on the body of the SensorPod to detect perturbations in the field due to the presence of underwater, stationary, nonconducting spherical targets. The expected measurement depends on the location, size, shape, and conductivity of an object, as well as the strength of the electric field generated by the robot; for details, see [73]. The perturbed electric fields and resulting differential measurements for targets in two locations relative to the SensorPod are shown in Fig. 4, and the differential voltage measurement is plotted in Fig. 5(a). Fig. 5(b) shows the expected differential measurement for two candidate sensor trajectories. The multimedia attachment provides additional intuition regarding the SensorPod and the observation model. The solid line trajectory is more informative, as measured using Fisher Information, than the dashed line; our goal is to automatically synthesize trajectories that are similarly more informative.



(a)

(b)

Fig. 4. The SensorPod (gray) measures the difference between the field voltage at two sensors (labeled A and B above). The field (simulated isopotential lines are plotted in black above) is generated by two excitation electrodes on the SensorPod body. The field changes in the presence of a target with different conductivity than the surrounding water. The 0 V line is bolded. The perturbation caused by an object results in a different differential measurement between sensors A and B based on the position of the object relative to the SensorPod. Note that the SensorPod is not measuring the field itself (which is emitted by, and moves with, the robot), but the voltage differential between two sensors induced by disturbances in the field. In (a), a $+0.2$ mV expected voltage difference between the sensors (A and B) on the SensorPod for a target located as shown; in (b), a -0.2 mV difference is expected. Please refer to color version online. For more information and an animation of this plot, please see the multimedia video attachment.

The objective in the experimental results presented in Section V is to estimate a set of unknown, static, parameters describing individual spherical underwater targets. Details and assumptions for implementation of both the Bayesian filter and the calculation of the EID for the SensorPod robot, including for the multiple target case, can be found in Appendix A; an overview of the algorithm is provided here, corresponding to the diagram in Fig. 2. For a graphical animated overview of the algorithm, please also see the attached multimedia video.

The algorithm is initialized with the sensor state at the initial time $\boldsymbol{x}(0)$ and an initial probability distribution $p(\boldsymbol{\theta})$ for the parameters $\boldsymbol{\theta}$. We represent and update the parameter estimate using a Bayesian filter, which updates the estimated belief based on collected measurements (see Fig. 2, step A). The initial distribution can be chosen based on prior information or, in the case of no prior knowledge, assumed to be uniform on bounded domains. At every iteration of the EEDI algorithm, the EID is calculated by taking the expected value of the Fisher information with respect to the belief $p(\boldsymbol{\theta})$ (see Fig. 2, step B). For estimation of multiple parameters, we use the D-optimality metric on the expected Fisher information matrix (FIM), equivalent to maximizing the determinant of the expected information

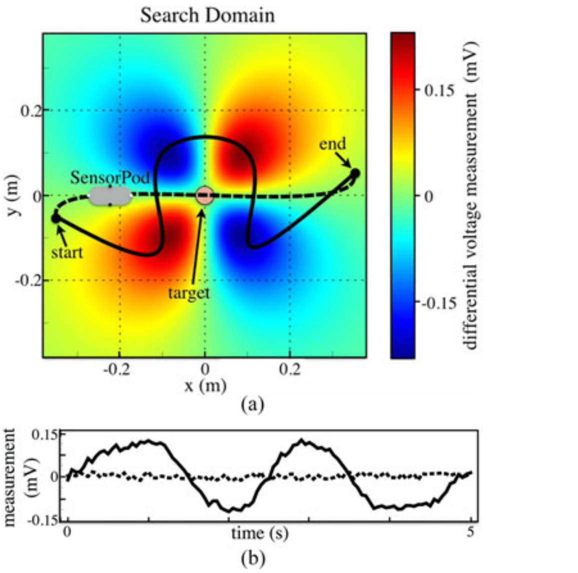


Fig. 5. Measurements collected by the SensorPod have a nonlinear and nonunique mapping to target location. In (a), the expected differential voltage measurement (between A and B from Fig. 4) is plotted as a function of robot centroid for a target (pink) located at $X, Y = (0, 0)$. Please refer to color version online. Two possible SensorPod trajectories are plotted in black (solid and dashed). The target is placed below the robot's plane of motion to prevent collisions. In (b), simulated differential voltage measurements for the trajectories in (a) are plotted as a function of time. The dashed trajectory in (a) yields uninformative measurements (i.e., that would be possible to observe for many potential target locations); the solid trajectory in (a) produces a series of measurements that are unique for that target position and therefore useful for estimation.

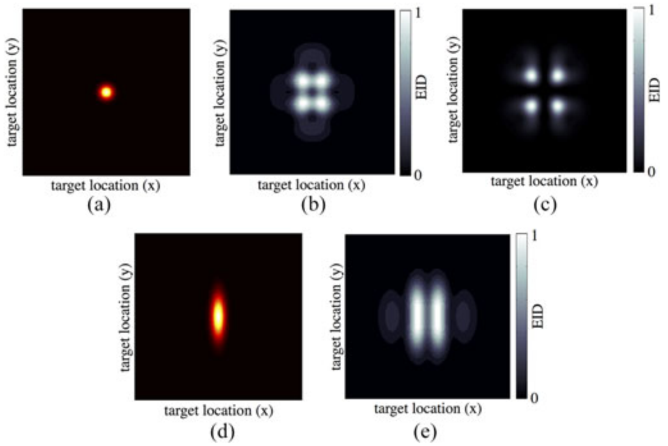


Fig. 6. The EID is dependent on the measurement model and the current probabilistic estimate. (b), (c), and (e) show examples of the EID for different PDFs, shown in (a) and (d), and estimation tasks for the SensorPod measurement model. The EID is calculated according to (12). In (a), a low-variance PDF of 2-D target location is shown. (b) shows the EID for target location for the PDF in (a). In (c), the EID for target location and radius is plotted; the projection of the corresponding PDF (defined in 3-D) onto the 2-D location space would be similar to (a). In (d), a higher variance PDF of 2-D target location. is shown, and (e) shows the EID map for the PDF in (d). In all cases, calculation of the EID produces a map over the search domain, regardless of the estimation task.

[51].[3] In Fig. 6, the corresponding EIDs for two different belief maps for 2-D target location (see Fig. 6(b) and (e)), as well as the EID for estimating both 2-D target location and target radius [see Fig. (6c)], are shown. The EID is always defined over the sensor configuration space (2-D), although the belief map may be in a different or higher dimensional space (e.g., over the 2-D workspace and the space of potential target radii). The normalized EID is used to calculate an optimally ergodic search trajectory for a finite time horizon (see Fig. 2, step C). The trajectory is then executed, collecting a series of measurements (see Fig. 2, step D, for time T). Measurements collected in step D are then used to update the belief $p(\boldsymbol{\theta})$, which is then used to calculate the EID in the next EEDI iteration. The algorithm terminates when the norm of the estimate falls below a specified value.

For localizing and estimating parameters for multiple targets, we initialize the estimation algorithm by assuming that there is a single target present, and only when the norm on the variance of the parameters describing that target falls below the tolerance ϵ do we introduce a second target into the estimation. The algorithm stops searching for new targets when one of two things happen: 1) parameters for the last target added converge to values that match those describing a target previously estimated (this would only happen if all targets have been found, as the EID takes into account expected measurements from previous targets), or 2) parameters converge to an invalid value (e.g., a location outside of the search domain), indicating failure. The algorithm terminates when the entropy of the belief map for all targets falls below a chosen value; for the 0 target case, this means that the SensorPod has determined that there are no objects within the search space. Note that the EID for new targets takes into account the previously located targets.

A. Assumptions: Stationary Target Localization using the SensorPod (Ergodic Exploration of Distributed Information Example)

We make a number of assumptions in choosing steps A, B, and D in Fig. 2, detailed in Appendix A. We assume a measurement is made according to a known differentiable measurement model (a function of sensor location and parameters) and assume the measurements have zero-mean, Gaussian, additive noise.[4] We assume independence between individual measurements, given that the SensorPod state is known and the measurement model is not time varying. Measurement independence is commonly assumed, for example, in occupancy grid problems [29]; however, more sophisticated likelihood functions that do not rely on this assumption of independence [74] could be used without significantly changing the structure of the algorithm.

For the single-target case, we maintain a joint probability distribution for parameters describing the same target as they are likely to be highly correlated. In order to make the problem of

[3] Note that alternative choices of optimality criteria may result in different performance for different problems based on, for example, the conditioning of the information matrix. D-optimality is commonly used for similar applications, and we found it to work well experimentally; however, the rest of the EEDI algorithm is not dependent on this choice of optimality criterion.

[4] Related work in active electrosense has shown that zero-mean Gaussian is a reasonable assumption for sensor noise [5].

finding an arbitrary number of objects tractable, we assume that the parameters describing different targets are independent and that a general additive measurement model may be used, similar to [24], [28], and [75]. Although the voltage perturbations from multiple objects in an electric field do not generally add linearly, we make the assumption that the expected measurement for multiple objects can be approximated by summing the expected measurement for individual objects, which simplifies calculations in Appendix A.[5] While the computation of Fisher Information and the likelihood function used for the Bayesian update depend on the assumptions mentioned above, the ergodic optimal control calculations do not, and only depend on the existence of an EID map.

The SensorPod is attached to a 4-degree-of-freedom (DOF) gantry system, which allows use of a kinematic model of the SensorPod in (3), i.e., the equations of motion for the experimental system are $\dot{\boldsymbol{x}}(t) = \boldsymbol{u}(t)$, where $\boldsymbol{x}$ is the SensorPod position in 1-D (see Sections V-A and V-E) or 2-D (see Sections V-B–V-D). The kinematic model and 2-D search space also enable comparison with other search methods; however, it should be noted that EEDI is applicable to dynamic nonlinear systems as well, as will be demonstrated in simulation in Section V-G.

Ergodic trajectory optimization, presented in Section III, calculates a trajectory for a fixed-length time horizon T, assuming that the belief, and therefore the EID map, remains stationary over the course of that time horizon. In the following experiments, this means that each iteration of the closed-loop algorithm illustrated in Fig. 2 involves calculating a trajectory for a fixed time horizon, executing that trajectory in its entirety, and using the series of collected measurements to update the EID map before calculating the subsequent trajectory. The complete search trajectory, from initialization until termination, is therefore comprised of a series of individual trajectories of length T, where the belief and EID are updated in between (this is also true for the alternative strategies used for comparison in Section V). The EID map could alternatively be updated, and the ergodic trajectory replanned after each measurement or subset of measurements, in a more traditional receding horizon fashion, or the time horizon (for planning and updating) could be optimized.

B. Performance Assessment

In the experiments in Section V, we assess performance using *time to completion* and *success rate*. Time to completion refers to the time elapsed before the termination criterion is reached, and a successful estimate obtained. We present results for time until completion as the "slowdown factor." The *slowdown factor* is a normalization based on minimum time until completion for a particular set of experiments or simulation. For a trial to be considered successful, the mean of the estimate must be within a specified range of the true target location, and in Section V-C, the number of targets found must be correct. The tolerance used on the distance of the estimated parameter mean to the true parameter values was 1 cm for the 1-D estimation experiments and 2 cm for 2-D experiments. In both cases, this distance was more than twice the standard deviation used for our termination criterion.

A maximum runtime was enforced in all cases (100 s for 1-D and 1000 s for 2-D experiments). For simple experimental scenarios, e.g., estimation of the location of a single target in 2-D (see Section V-B), these time limits were longer than the time to completion all algorithms in simulation. Additional motivation for limiting the runtime were constraints on the physical experiment and the observation that when algorithms failed, they tended to fail in such a way that the estimate variance never fell below a certain threshold (excluding the random walk controller), and the success criteria listed above could not be applied.

V. Trial Scenarios and Results

Experiments were designed to determine whether the EEDI algorithm performs at least as well as several alternative choices of controllers in estimating of the location of stationary target(s), and whether there were scenarios where EEDI outperforms these alternative controllers, e.g., in the presence of distractor objects or as the number of targets increases. Experiments in Sections V-A–V-D are performed using the kinematically controlled SensorPod robot and simulation, and these results are summarized in Section V-F. In Section V-G, we transition to simulation-only trials to demonstrate successful closed-loop estimation of target location, but compare trajectories and performance using three models of the robot: the kinematic model of the experimental system, a kinematic unicycle model, and a dynamic unicycle model.

In Sections V-A-V-D, we compare the performance of EEDI to four different algorithms (1–4) based on the following three implementations of information maximizing controllers and a random walk controller:

1) *Information gradient ascent controller (IGA):* The IGA controller drives the robot in the direction of the gradient of the EID at a fixed velocity of 4 cm/s, inspired by controllers used in [30], [32], [33], and [47].
2) *Information maximization controller (IM):* The SensorPod is controlled to the location of the EID maximum, at a constant velocity for time T, similar to [26], [28], [50], [56].
3) *Greedy expected entropy reduction (gEER):* At each iteration, 50 locations are randomly selected, within a fixed distance of the current position. The SensorPod is controlled to the location that maximizes the expected change in entropy, integrated over the time horizon T.[6] This approach is similar to the method of choosing control actions in [8], [9], [25], and [38].

[5] Additional experimental work (not shown), demonstrated that at a minimum of 6-cm separation between objects, there is no measurable error using this approximation; in the experimental and simulated trials, we use a minimum separation of 12 cm.

[6] The expected entropy reduction is $H(\theta) - E[H(\theta)|V^+(t)]$, where $H(\theta) = -\int p(\theta)\log p(\theta) d\theta$ is the entropy of the unknown parameter θ [46], [75], and $V^+(t)$ is the expected measurement, calculated for each candidate trajectory $x^+(t)$, the current estimate $p(\theta)$, and the measurement model.

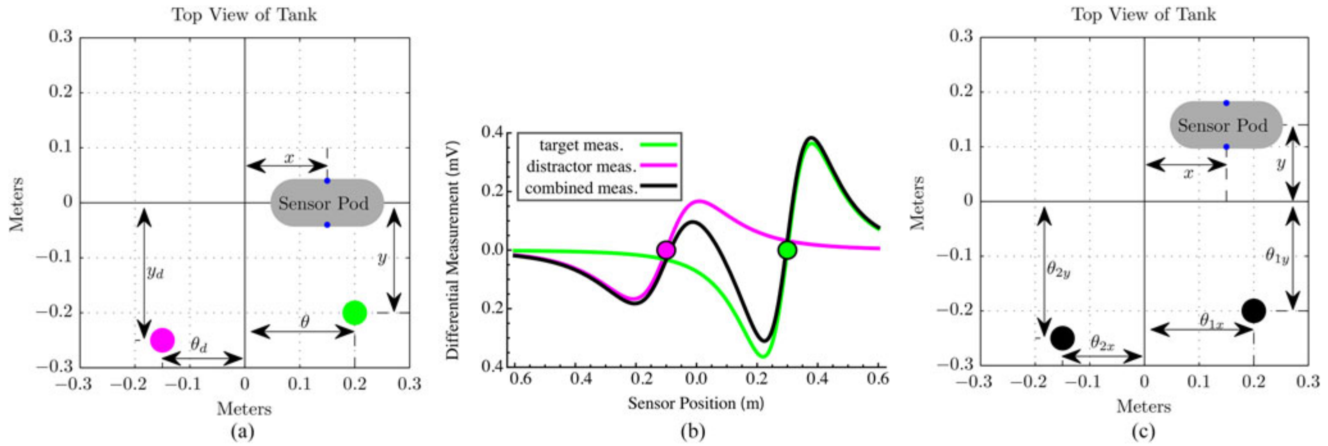


Fig. 7. In (a), the tank configuration is illustrated for estimation of target location in 1-D (see Sections V-A and V-E). The target object (green) was placed at a fixed distance of y $=$ 0:2 m from the SensorPod line of motion, and the distractor (pink) at $\mathrm{y_d}$ $=$ 0:25 m. In (b), the expected voltage measurement over 1-D sensor state for the target (pink) and distractor (green) objects alone, as well as when both target and distractor are present (black), are shown. In (c), an example of the tank configuration for 2-D localization of two targets is shown. For all trials, SensorPod and object locations are measured from the center of the tank. Targets were placed below the robot's plane of motion to prevent collisions. The orientation of the robot is held constant. The voltage sensors sample at 100 Hz, with an assumed standard deviation of 100 μV for the measurement noise, the experimentally observed noise level of the SensorPod sensors.

4) *Random walk (RW):* The SensorPod executes a randomly chosen constant velocity trajectory from the current sensor position for time T, similar to [5].

The planning horizon T was the same for all controllers so that the same number of measurements is collected.

Alternative algorithms, for example, a greedy gradient-based controller (IGA) or a random walk (RW), produce control signals with less computational overhead than the EEDI algorithm because the EEDI involves solving a continuous trajectory optimization problem and evaluating an additional measure on ergodicity. In the next section, we demonstrate several scenarios in which the tradeoff in computational cost is justified if the estimation is likely to fail or suffer significantly in terms of performance using less expensive control algorithms. Additionally, the alternative algorithms, while appropriate for the kinematically-controlled SensorPod robot, do not generalize in an obvious way to nonlinear dynamic models. This is one of our reasons for desiring a measure of nonmyopic search that can be expressed as an objective function (i.e., ergodicity). Given an objective, optimal control is a convenient means by which one makes dynamically dissimilar systems behave similarly to each other according to a metric of behavior. In the case of exploration, the measure is of coverage relative to the EID—however, it is constructed.

TABLE I
PERFORMANCE MEASURES FOR ESTIMATION OF SINGLE-TARGET LOCATION IN 1-D

Description	EEDI	gEER	IM	IGA	RW
Exp. Success %	100	50	60	50	80
Sim. Success %	100	60	71	66	99
Exp. Slowdown Factor	1	1.4	2.1	2.7	2.7
Sim. Slowdown Factor	1	2.1	2.1	2.3	6.3

Results for time until completion (slowdown factor) are only shown for successful trials. Slowdown factor of 1 corresponds to 15.2 s in experiment, 7.6 s in simulation. Results are shown for 100 simulated and 10 experimental trials.

A. Performance Comparison for One-Dimensional Target Estimation in the Presence of an Unmodeled Distractor

In this section, the robot motion is constrained to a single dimension, and the estimation objective is the 1-D location θ of a single stationary target with known radius. A *distractor object* is placed in the tank, as an unmodeled disturbance, in addition to the (modeled) *target object*. Both the target and the distractor were identical nonconductive, 2.5-cm-diameter spheres, placed at different fixed distances from the SensorPod's line of motion [see Fig. 7(a)]. The voltage signal from the distractor object is similar but not identical to that of the target [see Fig. 7(b)]. Placing the distractor object further from the SensorPod line of motion results in decreased magnitude and rate of change of the voltage trace. Introducing an unmodeled distractor even in a 1-D sensing task was enough to illustrate differences in the performance of the EEDI Algorithm and Algorithms 1–4.

We performed 100 trials in simulation and 10 in experiment, with the target position held constant and the distractor's position along the SensorPod's line of motion randomized.[7] The results for success rate and average slowdown factor (for successful trials), averaged over all trials in simulation and experiment, are summarized in Table I. The slowdown factor is the average time until completion, normalized by the minimum average time until completion in experiment or simulation. Results presented in Table I demonstrate that the EEDI algorithm localizes the target successfully 100% of the time, and does so more quickly than Algorithms 1–4.

[7] The only additional consideration in the experimental scenario was that the tank walls and the water surface have a nonnegligible effect on measurements. We compensate for this by collecting measurements on a fine grid in an empty tank, and subtracting these measurements at each measurement point during estimation.

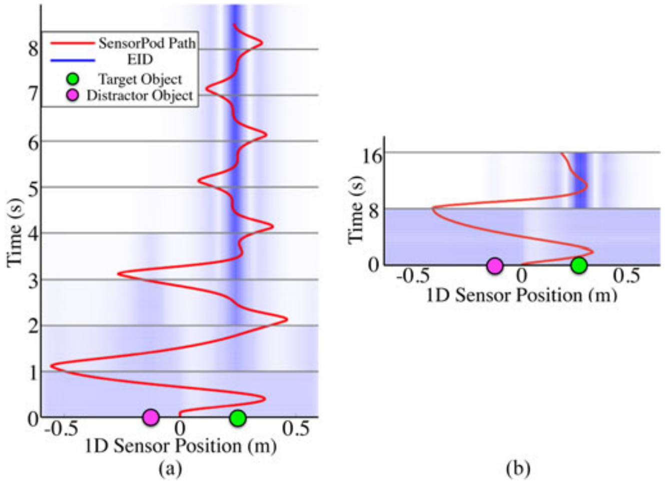


Fig. 8. Examples of closed-loop optimally ergodic search in (a) simulation and (b) experiment are shown above. The EID is shown as a density plot in blue, and the search trajectory in red (refer to color version online). The belief and trajectory are recalculated every second in simulation and every 8 s in experiment.

Differences in time to completion between experimental and simulated trials are due to experimental velocity constraints. In simulation, the time horizon used for trajectory optimization, and therefore between PDF updates, was one $T = 1$ s. A longer ($T = 8$ s) time horizon was used for experimental trajectory optimization, avoiding the need to incorporate fluid dynamics in the measurement model; at higher velocities, the water surface is sufficiently disturbed to cause variations in the volume of fluid the field is propagating through, causing unmodeled variability in sensor measurements.

Fig. 8 shows experimental and simulated examples of closed-loop 1-D trajectories generated using the EEDI algorithm. Given no prior information (a uniform belief), the ergodic trajectory optimization initially produces uniform-sweep-like behavior. In the experimental trial shown in Fig. 8(b), the termination criteria on the variation of the PDF are reached in only two iterations of the EID algorithm, a result of the longer time horizon and resulting higher density measurements. The distributed sampling nature of the EEDI algorithm can be better observed in the simulated example shown in Fig. 8(a), where shorter time horizons and therefore more sparse sampling over the initial sweep require more iterations of shorter trajectories. As the EID evolves in Fig. 8(a), the shape of the sensor trajectory changes to reflect the distribution of information. For example, the sensor visits the local information maximum resulting from voltage perturbations due to the target and the local information maximum due to the distractor between 1 and 4 s. Experimental results for this trial were presented in [76].

B. Performance Comparison for Estimating the Two-Dimensional Location of a Single Target

In this section, the robot is allowed to move through a 2-D workspace and the objective was to compare the performance of EEDI to gEER and RW for 2-D, stationary target localization, i.e., $\boldsymbol{\theta} = (\theta_x, \theta_y)$. No distractor object was present as the difference in performance between algorithms was notable even without introducing a distractor object. Fig. 7(c) shows an example tank configuration for multiple target estimation in 2-D. We omit comparison with IGA and IM for 2-D experiments; RW is the simplest controller to calculate and resulted in high success percentage for 1-D trials, and gEER, with performance similar to IGA and IM on average in 1-D trials, is qualitatively similar to our approach and more commonly used.

Ten trials were run for each of the EEDI, gEER, and RW algorithms, both in experiment and simulation, with the target location randomly chosen. Fig. 9 shows the convergence of the belief at 10 s intervals ($T = 10$), as well as the corresponding EID and ergodic trajectory. The performance measures for experimental and simulated trials using the EEDI, gEER, and RW algorithms are shown in Table II. In simulation, all three algorithms have 100% success rate, while the gEER and RW controllers have a 10% lower success rate in the experimental scenario. The gEER controller requires roughly 10–20% more time to meet the estimation criteria, whereas the RW controller requires about two to three times more time. As mentioned in the previous section, although gEER performs well in this scenario, it did not perform as well with distractors.

C. Performance Comparison for Estimating the Two-Dimensional Location of Multiple Targets

Having demonstrated that the EEDI algorithm modestly outperforms gEER and drastically outperforms RW (in terms of time) for localizing a single stationary target in 2-D, we next sought to compare EEDI performance localizing multiple targets [see Fig. 7(c)]. We compare the EEDI algorithm to the gEER and RW controllers, again leaving out IM and IGA because of their poor performance in Section V-A. We performed localization estimation for scenarios where there were either 0, 1, 2, or 3 targets in the tank, all 2.5-cm diameter. We conducted five trials in simulation and experiment, for each number of targets (with all locations selected randomly). Fig. 10 shows the percentage of successful trials and average slowdown factor as a function of the number of targets in the tank. The slowdown factor is calculated by normalizing average time until completion by the minimum average time until completion for all algorithms and all target numbers.

In the experimental scenario [see Fig. 10(a)], the EEDI algorithm had a higher success rate than both the gEER and RW controllers for higher numbers of objects. The slowdown factor using the EEDI algorithm was very similar to the gEER algorithm for 0–2 objects (the gEER controller never successfully localized 3 objects), and much shorter than the RW controller. In simulation [see Fig. 10(b)], the success rate of the EEDI algorithm matched that of the RW; however, the RW slowdown factor was much greater.

D. Performance Degradation With Signal-to-Noise Ratio

The next trial is used to demonstrate an extension of the EEDI Algorithm to nonlocation parameters and to examine performance degradation as a function of the signal-to-noise

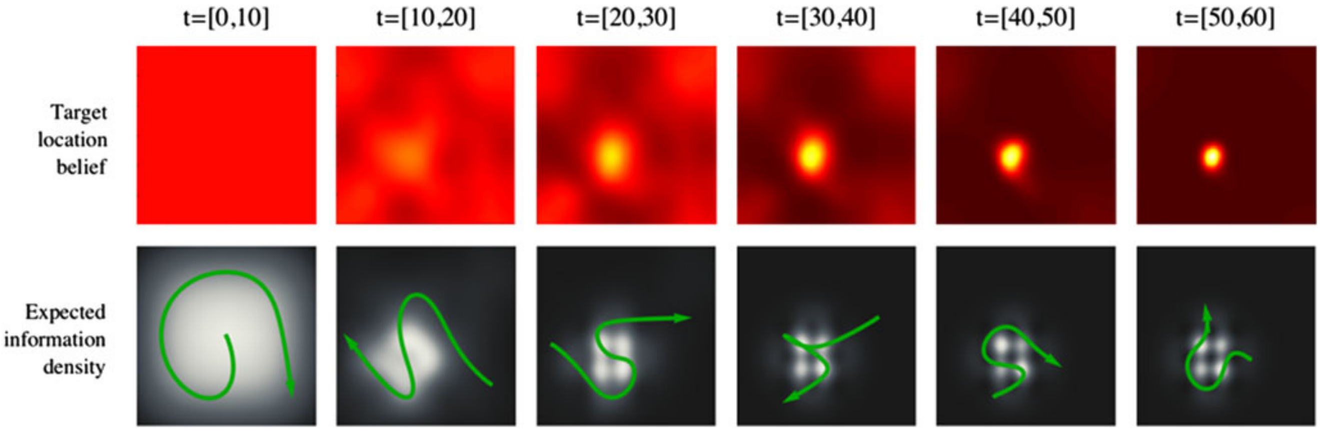


Fig. 9. Progression of the estimate of the 2-D target location using the EEDI algorithm. As the algorithm progresses, collected measurements evolve the estimate from a uniform distribution over the workspace (top-leftmost figure), to a concentrated distribution at the correct location. At each interval, the EID is calculated from the updated estimate, which is then used to calculate an ergodic search trajectory.

TABLE II
PERFORMANCE MEASURES FOR ESTIMATION OF SINGLE-TARGET LOCATION IN 2-D

Description	EEDI	gEER	RW
Exp. Success %	100	90	90
Sim. Success %	100	100	100
Exp. Slowdown Factor	1	1.2	2.9
Sim. Slowdown Factor	1	1.1	2.6

Results for time until completion (slowdown factor) are only shown for successful trials. Slowdown factor of 1 corresponds to 64 s in experiment, 65.2 s in simulation. Results are shown for ten simulated and ten experimental trials.

ratio. As mentioned, the EEDI algorithm can also be used to determine additional parameters characterizing target shape, material properties, etc. The only requirement is that there be a measurement model dependent on—and differentiable with respect to—that parameter. Parameters are incorporated into the algorithm in the same way as the parameters for the spatial location of a target (see Appendix B). We, therefore, demonstrate effectiveness of the EEDI algorithm for estimation of target radius as well as 2-D target location. We estimated target location and radius for ten different radii varying between 0.5 and 1.5 cm. Five trials were performed for each target radius, with randomly chosen target locations. By varying the radius of the target, which for our sensor results in scaling the signal-to-noise ratio,[8] we are able to observe relative performance of several comparison algorithms as the signal-to-noise ratio drops off. Trials were performed in simulation only. Fig. 11 shows the mean success rate of the five simulated trials as a function of target radius. For EEDI, gEER, and RW, the success rate decreased as the radius decreased. This is expected, as the magnitude of the voltage perturbation, and therefore the signal-to-noise ratio, scales with r^3. For objects with $r < 0.9$ cm, the peak of the expected voltage perturbation is less than the standard deviation of the sensor noise. Nevertheless, the EEDI algorithm had a higher success rate than gEER and RW for radii between 0.5 and 1 cm.

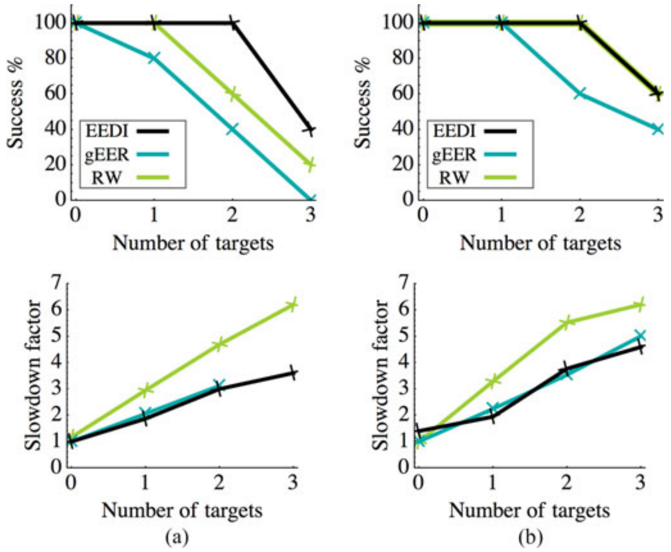


Fig. 10. Performance measures for estimation of multiple target locations in 2-D for (a) five experimental and (b) five simulated trials. Slowdown factor of 1 corresponds to 50 s in simulation, 60 s in experiment.

E. Comparison of Sensitivity to Initial Conditions

Finally, we use the 1-D estimation scenario (the same as that in Section V-A) to illustrate the relative sensitivity of the EEDI algorithm and Algorithms 1–4 to the initial conditions of the sensor with respect to locations of the target and an unmodeled disturbance. This captures the likelihood of different controllers to become stuck in local minima resulting from the presence of a distractor object which produces a measurement similar but not identical to the target.

We executed a total of 110 simulated trials for each algorithm. Ten trials were simulated for 11 equally spaced target locations. For each target location, the distractor location was randomized, with a minimum distance of 25-cm distance from the target

[8] The signal drops approximately with the fourth power of the distance from a spherical target and increases with the third power of target radius [77].

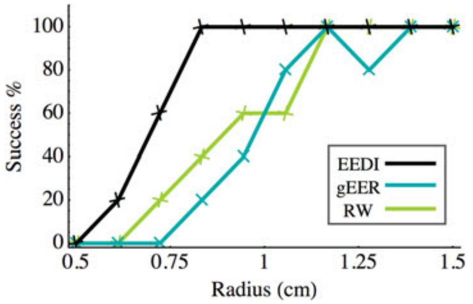


Fig. 11. Success rate for estimation of location and radius, as a function of target radius, for simulated trials only.

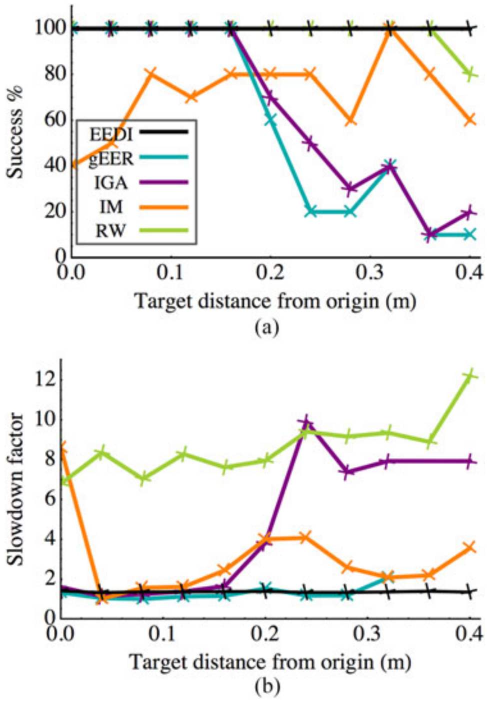


Fig. 12. Performance measures for estimation of single-target location in 1-D are shown for the EEDI algorithm and Algorithms 1–4. Results of 110 simulated trials are shown for each algorithm; for each of 11 target locations, ten simulated trials were performed with the 1-D distractor object location randomized (a fixed distance from the SensorPod line of motion was maintained). A slowdown factor of 1 corresponds to 5.61 s; slowdown factor is not shown for target distances with less than 10% success rate.

(along the SensorPod line of motion, to prevent electrosensory occlusion). One hundred and ten trials allowed significant separation of the results from different controllers. For all trials, the SensorPod position was initialized to $(x, y) = 0$.

Fig. 12 shows the performance measures for Algorithms 1–4. The slowdown factor is calculated by normalizing average time until completion by the minimum average time over all algorithms and all target locations. When the target was located near the SensorPod initial position, EEDI, gEER, IGA, and RW perform comparably in terms of success percentage and time, with the exception being the RW controller, which is predictably slower. Success rate drops off using gEER and IGA for target positions further from the SensorPod initial position. Note that IM performs poorly if the target is located exactly at the robot start position, due to the nonlinear characteristics of electrolocation. A 0-V measurement would be observed for a target located at the sensor position or anywhere sufficiently far from the sensor; this means that the initial step of the IM algorithm has a high probability of driving the sensor to a position far from the target. EEDI, on the other hand, localized the target in essentially constant time and with 0% failure rate regardless of location. The RW algorithm performs as well as the EEDI algorithm in terms of success rate, but is approximately eight times slower.

F. Summary of Experimental Results

In Sections V-A and V-B, we provide examples of successful estimation trajectories for the EEDI algorithm. In the 2-D estimation problem in Section V-B, we observe that both success rate and time until completion are comparable using both EEDI and gEER algorithms (with time being much longer for the random walk controller). While this scenario illustrates that our algorithm performs at least as well as a greedy algorithm in a simple setting, and more efficiently than a random controller, where we really see the benefit in using the EEDI algorithm is when the robot is faced with more difficult estimation scenarios. Experiments in Section V-E showed that the EEDI algorithm was robust with respect to initial conditions (i.e., whether or not the sensor happens to start out closer to the distractor or target object) where Algorithms 1–4 are sensitive. For Algorithms 1–4, the further the target was from the initial SensorPod position, the more likely the estimation was to fail or converge slowly due to local information maxima caused by the distractor. Similarly, when the estimation objective was target localization for varying numbers of targets in Section V-C (a scenario where many local information maxima are expected), the success rate of the EEDI algorithm is higher than expected entropy reduction and completion time is shorter than the random walk as the number of targets increased. Finally, the success rate of the EEDI algorithm degraded the least quickly as the signal-to-noise ratio declined. In addition to outperforming alternative algorithms in the scenarios described, the ergodic trajectory optimization framework enables calculation of search trajectories for nonlinear dynamically constrained systems.

G. Comparison of Different Dynamic Models

One of the benefits of ergodic optimal control is that the control design does not change when we switch from a kinematic robot model to a dynamic robot model. While the physical SensorPod robot is controlled kinematically due to the gantry system, we can simulate nonlinear and dynamic models to see how dynamics might influence information gathering during untethered movement for future SensorPod iterations. We simulate automated target localization using the EEDI algorithm for the SensorPod robot using three different models for the robot dynamics. All parameters in the ergodic optimal control algorithm are exactly the same in all three cases: the weights on minimizing control effort versus maximizing ergodicity, in (3), were set to $\gamma = 20$, $R = 0.01\mathbb{I}$, (where $\mathbb{I}$ is a 2×2 identity matrix), and the planning time horizon was $T = 10$. In all three cases below,

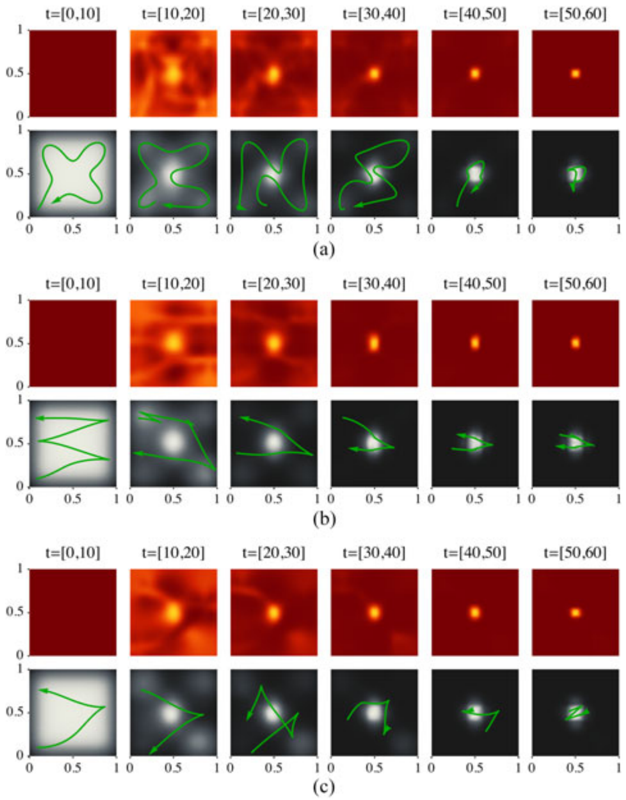


Fig. 13. Progression of the estimate of the 2-D target location using the EEDI algorithm, in simulation, for three different systems performing the same task; (a) linear, kinematic system, (b) kinematic unicycle model (nonlinear, kinematic system), and (c) dynamic unicycle model (nonlinear, dynamic system). As the algorithm progresses, collected measurements evolve the estimate (heatmap) from a uniform distribution over the workspace [top-leftmost figure in each (a)-(c)] to a concentrated distribution at the correct location. At each interval, the EID (grayscale) is calculated from the updated estimate, which is then used to calculate an ergodic search trajectory (green).

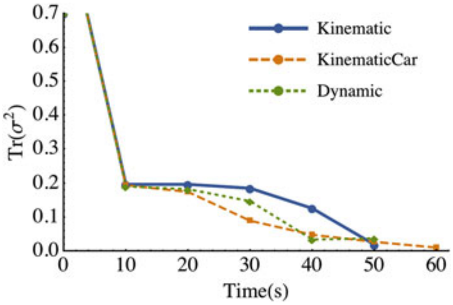


Fig. 14. Trace of the covariance of the 2-D target location estimate is plotted as a function of time. We observe similar overall convergence behavior for all three systems for this particular set of initial conditions and weighted objective function. The covariance is updated after executing each 10-s long trajectory.

the measurement model was identical and defined relative to the XY position of the robot, although the system state differs. The only changes in the implementation are the robot's state and equations of motion for the three systems, defined below.

1) Linear Kinematic System: The state is $\boldsymbol{x}(t) = (x(t), y(t))$, where $x(t)$ and $y(t)$ are Cartesian coordinates, and the equations of motion are $\dot{\boldsymbol{x}}(\boldsymbol{t}) = \boldsymbol{u}(t)$. The initial conditions were $\boldsymbol{x}(0) = (0.1, 0.1)$.

2) Nonlinear Kinematic System: We use the standard kinematic unicycle model. The state is $\boldsymbol{x}(t) = (x(t), y(t), \theta(t))$, where $x(t)$ and $y(t)$ are Cartesian coordinates and $\theta(t)$ is a heading angle, measured from the x-axis in the global frame. The control $\boldsymbol{u}(t) = (v(t), \omega(t))$ consists of a forward velocity $v(t)$ and angular velocity $\omega(t)$. The equations of motion are

$$\dot{\boldsymbol{x}}(t) = \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \end{bmatrix} = \begin{bmatrix} v(t)\cos\theta(t) \\ v(t)\sin\theta(t) \\ \omega(t) \end{bmatrix}. \tag{4}$$

The initial conditions were $\boldsymbol{x}(0) = (0.1, 0.1, 0)$.

3) Nonlinear Dynamic System: We use a dynamic variation on the unicycle model. In this case, the state is $\boldsymbol{x}(t) = (x(t), y(t), \theta(t), v(t), \omega(t))$, where x, y, θ, v, ω are the same as in the kinematic unicycle model. The control inputs are $\boldsymbol{u}(t) = (a(t), \alpha(t))$, with the equations of motion

$$\dot{\boldsymbol{x}}(t) = \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \\ \dot{v}(t) \\ \dot{\omega}(t) \end{bmatrix} = \begin{bmatrix} v(t)\cos\theta(t) \\ v(t)\sin\theta(t) \\ \omega(t) \\ \frac{1}{2}a(t) \\ \alpha(t) \end{bmatrix}. \tag{5}$$

The initial conditions were $\boldsymbol{x}(0) = (0.1, 0.1, 0, 0, 0)$. Fig. 13 illustrates the progression of the EEDI algorithm for static, single-target localization for all three systems. In all cases, we use a finite time horizon of $T = 10$ s for trajectory optimization, and the PDF is initialized to a uniform distribution. While the types of trajectories produced are qualitatively different because of the different dynamic constraints, we observe similar convergence behavior for all three systems for this particular set of initial conditions and weights in the objective function. In Fig. 14, the trace of the estimate covariance is plotted as a function of EEDI iterations.

VI. Conclusion

We have presented a receding horizon control algorithm for active estimation using mobile sensors. The measurement model and belief on the estimates are used to create a spatial map of expected information gain. We implement our algorithm on a robot that uses a bioinspired sensing approach called electrolocation [4]. Ergodic trajectory optimization with respect to the expected information distribution, as opposed to information maximization, is shown to outperform alternative information maximization, entropy minimization, and random walk controllers in scenarios when the signal-to-noise ratio is low or in the presence of disturbances.

One major advantage of ergodic trajectory optimization is that the formulation is suitable for systems with linear or nonlinear, kinematic, or dynamic motion constraints, as shown in Section V-G. Additionally, the method does not formally rely on discretization of the search space, the action space, or the belief space. Although numerical integration schemes are used in solving differential equations or updating the belief, discretization is an implementation decision as opposed to a part of the problem statement or its solution. Another benefit is that neither assuming information submodularity [23], [35], [58] nor selecting waypoints [42], [49] is required to distribute measurements among different regions of high expected information when planning over a long-time horizon. Using ergodicity as an objective also means that the algorithm is suitable for both coverage [67], [68] or "hotspot" sampling, without modification. If the information density is very concentrated, the optimally ergodic trajectory will be similar to an information maximizing solution.[9] On the other hand, if the information density is diffuse (or the planning time horizon very long), the optimally ergodic solution will approximate a coverage solution. In Figs. 9 and 13, coverage-like solutions are observed for the initial nearly uniform belief; although the belief converges, the EID does not converge to a unimodal distribution due to nonlinearities in the measurement model.

This paper deals exclusively with finding information about a finite set of stationary targets. However, ergodic search generalizes to both time-varying systems as well as estimation of a continuum of targets (e.g., fields [20], [21]) in a reasonably straightforward fashion. Field exploration can be achieved by using an appropriate choice of measurement model and belief update in the EID calculation [20], [21], [35]–[38]. Time can be incorporated into the measurement model describing not just *where* information about a parameter might be obtained, but also *when*—by extending the state in Section III-B to use time as a state.

The formulation of ergodic exploration provided in this paper also assumes that the dynamics are deterministic. However, the determinism restriction primarily makes calculations and exposition simpler. Adding stochastic process noise to the model can be achieved by replacing the deterministic, finite-dimensional equations of motion with the Fokker–Planck equations [71] for the nonlinear stochastic flow, without changing the mathematical formulation of ergodic control. Moreover, stochastic flows can be efficiently computed [78], [79] for a wide variety of robotic problems. Even when they cannot be, a wide class of stochastic optimal control problems are easily computable [80], [81], although for different objectives than ergodicity. Although generalization will be easier in some cases than others, the generalization of ergodic control to uncertain stochastic processes may be initially approached rather procedurally. Generalizing ergodic control to more general uncertain (nonstochastic) systems, such as robust control strategies [82], would substantially complicate matters and would require a much more challenging generalization that would be a very promising avenue of future research.

In addition to the various limiting assumptions mentioned in Sections III-C and IV-A in constructing the EEDI algorithm for target localization, one of the major limitations of the current formulation is computational expense. Computational expense stems both from the need to calculate a map of the EID over the workspace in order to formulate the ergodic objective function, and the need to calculate trajectories over a finite-time horizon. The projection-based trajectory optimization involves solving a set of differential equations, which scale quadratically with the state. This is not necessarily a problem for applications where offline control calculations are acceptable, or in a receding horizon framework that uses efficient numerical methods. To that end, preliminary work has explored solving a discrete version of ergodic trajectory optimization using variational integrators [83]. Nevertheless, for applications that have a linear measurement model, trivial dynamics, and a simple environment, standard strategies like gradient-based approaches that only locally approximate the expected information [30], [32], [33], [47] would be effective and much more computationally efficient. The advantage of using ergodic trajectory optimization is that it is possible to formulate and solve exploration tasks whether or not the environment is simple or the measurement model linear, and to perform robustly when these "nice" conditions cannot be guaranteed, as in the experimental work featured in this paper.

APPENDIX
ERGODIC EXPLORATION OF DISTRIBUTED INFORMATION FOR STATIONARY TARGET LOCALIZATION USING THE SENSORPOD ROBOT

A. Bayesian Probabilistic Update

The goal is to estimate a set of m unknown, static, parameters $\boldsymbol{\theta} = [\theta_1, \theta_2, ..., \theta_m]$ describing individual underwater targets. We assume a measurement V is made according to a known measurement model $V = \Upsilon(\boldsymbol{\theta}, \boldsymbol{x}) + \delta$, where the measurement model $\Upsilon(\cdot)$ is a differentiable function of sensor location and target parameters, and δ represents zero-mean Gaussian noise. Specifically, we use a previously derived measurement model for submerged, sufficiently isolated, nonconducting spheres [73]. The joint distribution $p(\boldsymbol{\theta})$ is updated every iteration k of the EEDI algorithm using a Bayesian filter based on the measurement $V_k(t)$, the measurement model, and the sensor trajectory $x_k(t)$ over the planning period T,

$$p_{k+1}\left(\boldsymbol{\theta}|V_k(t), \boldsymbol{x}_k(t)\right) = \eta\, p\left(V_k(t)|\boldsymbol{\theta}, \boldsymbol{x}_k(t)\right) p_k(\boldsymbol{\theta}). \qquad (6)$$

$p(\boldsymbol{\theta})$ is the PDF calculated at the previous iteration, p_k $(\boldsymbol{\theta}, V_{k-1}(t), \boldsymbol{x}_{k-1}(t)))$, η is a normalization factor, and $p(V_k(t)|\boldsymbol{\theta}, \boldsymbol{x}_k(t))$ is the likelihood function for $\boldsymbol{\theta}$ given $V_k(t)$.

Assuming independence between individual measurements, given that the SensorPod state is known and the measurement model is not time varying, the likelihood function for all measurements taken along $\boldsymbol{x}_k(t)$ is the product of the likelihood of taking a single measurement $V_k(t_j)$ at time t_j, for all times

[9] Note that this would only happen for measurement models that cause the EID to converge to a low-variance unimodal distribution that approximates a delta function (where the equivalence between an information maximizing solution and an ergodic solution follows directly from their definitions); this does not happen in the examples shown in Section V. Because of the highly nonlinear measurement model, the EID converges to a multimodal density function, as shown in Fig 6(b).

$t_j \in [t_0, T]$. Assuming a Gaussian likelihood function, this is

$$p(V_k(t)|\boldsymbol{\theta}, \boldsymbol{x}_k(t)) = \prod_{j=t_0}^{T} \frac{1}{\sqrt{2\pi}\sigma} \times \exp\left[-\frac{(V_k(t_j) - \Upsilon(\boldsymbol{\theta}, \boldsymbol{x}_k(t_j)))^2}{2\sigma^2}\right]. \tag{7}$$

1) Probabilistic Update for Multiple Targets: We assume an additive measurement model $h(\boldsymbol{\Theta})$ to describe the expected measurement for multiple targets

$$h(\boldsymbol{\Theta}, x(t_j)) = \Upsilon(\boldsymbol{\theta_1}, \boldsymbol{x}(t_j)) + \Upsilon(\boldsymbol{\theta_2}, \boldsymbol{x}(t_j)) + \cdots + \Upsilon(\boldsymbol{\theta_m}, \boldsymbol{x}(t_j)), \tag{8}$$

where $\boldsymbol{\theta}_i$ is an $m \times 1$ vector made up of m parameters describing the ith target, and $\boldsymbol{\Theta}$ is the M length set of vectors $[\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, ..., \boldsymbol{\theta}_M]$, where M is the number of targets. Because we assume the measurements $\Upsilon(\boldsymbol{\theta_i})$ for different targets are independent of each other, we use different Bayesian filter updates for each target parameter set, evaluating M instances of (6). The likelihood function for parameter set $\boldsymbol{\theta}_i$ is

$$p(V(t)|(\boldsymbol{\theta}_i)) = \prod_{j=1}^{T} \frac{1}{\sqrt{2\pi}\sigma} \exp\left[\frac{(H(\boldsymbol{\theta}_i, \boldsymbol{x}(t_j)) - V(t_j))^2}{\sigma^2}\right]$$

where $H(\boldsymbol{\theta}_i, x(t_j))$ is the marginalization [71] of $h(\boldsymbol{\Theta}, x(t_j))$ over the parameters describing all other targets $\boldsymbol{\theta}_{j \neq i}$.

B. Expected Information Density

Fisher information [54] defines the informative regions of the search space based on the measurement model, a function of the parameter θ. Assuming Gaussian noise, the Fisher information for estimation of θ reduces to

$$\mathcal{I}(\theta, x) = \left(\frac{\partial \Upsilon(\theta, x)}{\partial \theta} \cdot \frac{1}{\sigma}\right)^2. \tag{9}$$

The Fisher information $\mathcal{I}(\theta, x)$ is the amount of information a measurement provides at location x for a given estimate of θ (based on the measurement model).

For estimation of multiple parameters $\boldsymbol{\theta}$ from a random variable v, the Fisher information is represented as an $m \times m$ matrix. For a measurement model $\Upsilon(\boldsymbol{\theta}, \boldsymbol{x})$ each element of the FIM can be simplified to

$$\mathcal{I}_{i,j}(\boldsymbol{x}, \boldsymbol{\theta}) = \frac{1}{\sigma^2} \frac{\partial^2 \Upsilon(\boldsymbol{\theta}, \boldsymbol{x})}{\partial \theta_i \partial \theta_j}. \tag{10}$$

Note that while we assume Gaussian noise to simplify the expression in (10), use of Fisher information in this context does not strictly require Gaussian noise. The Fisher information can be calculated offline and stored based on the measurement model, which reduces the number of integrations necessary at each iteration.

Since the estimate of $\boldsymbol{\theta}$ is represented as a probability distribution function, we take the expected value of each element of $\mathcal{I}(\boldsymbol{x}, \boldsymbol{\theta})$ with respect to the joint distribution $p(\boldsymbol{\theta})$ to calculate the expected information matrix, $\Phi(\boldsymbol{x})$. This is an $m \times m$ matrix, where the (i, j)th element is

$$\Phi_{i,j}(\boldsymbol{x}) = \frac{1}{\sigma^2} \int_{\theta_i} \int_{\theta_j} \frac{\partial^2 \Upsilon(\boldsymbol{\theta}, \boldsymbol{x})}{\partial \theta_i \partial \theta_j} p(\theta_i, \theta_j)\, d\theta_j d\theta_i. \tag{11}$$

This expression can be approximated as a discrete sum as required for computational efficiency.

Using the D-optimality metric [51] on the expected information matrix, the EID that is

$$\text{EID}(\boldsymbol{x}) = \det \Phi(\boldsymbol{x}). \tag{12}$$

1) Expected Information Density for Multiple Targets: Since the total information is additive for independent observations, we can write

$$I(\boldsymbol{\Theta}, x) = I(\boldsymbol{\theta}_1, x) + \cdots + I(\boldsymbol{\theta}_M, x) \tag{13}$$

where each term is calculated as in (10).

The EID for all parameters, for all targets, can be calculated as the sum of the determinants of the expected value of each term in (13), $\Phi_i(x)$, given the set of independent probabilities $p(\boldsymbol{\theta}_i)$ for each set of parameters $\boldsymbol{\theta}_i$ describing a single target

$$\text{EID}(x) = \eta \sum_{i}^{M} \det \Phi_i(x). \tag{14}$$

Since the FIM is positive semidefinite, the determinant of the FIM for each target is nonnegative. Note that this is just one approach of combining the expected information from several independent sources into a single map. Another option would have been to calculate the D-optimality for each target individually and calculate the subsequent trajectory based only on EID for the target with the highest integrated information (prior to normalization).

References

[1] G. Mathew and I. Mezic, “Metrics for ergodicity and design of ergodic dynamics for multi-agent system,” *Phys. D-Nonlinear Phenomena*, vol. 240, nos. 4/5, pp. 432–442, Feb. 2011.

[2] R. Krahe and E. Fortune, “Electric fishes: Neural systems, behavior and evolution,” *J. Exp. Biol.*, vol. 216, pp. 2363–2364, 2013.

[3] M. E. Nelson and M. A. MacIver, “Sensory acquisition in active sensing systems,” *J. Comp. Physiol. A*, vol. 192, no. 6, pp. 573–586, 2006.

[4] I. D. Neveln, Y. Bai, J. B. Snyder, J. R. Solberg, O. M. Curet, K. M. Lynch, and M. A. MacIver, “Biomimetic and bio-inspired robotics in electric fish research,” *J. Exp. Biol.*, vol. 216, no. Pt 13, pp. 2501–2514, Jul. 2013.

[5] J. R. Solberg, K. M. Lynch, and M. A. MacIver, “Active electrolocation for underwater target localization,” *Int J. Robot. Res.*, vol. 27, no. 5, pp. 529–548, 2008.

[6] M. A. MacIver, E. Fontaine, and J. W. Burdick, “Designing future underwater vehicles: principles and mechanisms of the weakly electric fish,” *IEEE J. Ocean. Eng.*, vol. 29, no. 3, pp. 651–659, Jul. 2004.

[7] S. Cowen, S. Briest, and J. Dombrowski, “Underwater docking of autonomous undersea vehicles using optical terminal guidance,” in *Proc. MTS/IEEE OCEANS Conf.*, Oct. 1997, vol. 2, pp. 1143–1147.

[8] C. Kreucher, K. Kastella, and A. O. Hero, “Sensor management using an active sensing approach,” *Signal Process.*, vol. 85, no. 3, pp. 607–624, 2005.

[9] D. Fox, W. Burgard, and S. Thrun, “Active Markov localization for mobile robots,” *Robot. Auton. Syst.*, vol. 25, no. 3–4, pp. 195–207, 1998.

[10] C. Cai and S. Ferrari, “Information-driven sensor path planning by approximate cell decomposition,” *IEEE Trans. Syst., Man, Cybern.*, vol. 39, no. 3, pp. 672–689, Jun. 2009.

[11] J. Toh and S. Sukkarieh, “A Bayesian formulation for the prioritized search of moving objects,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2006, pp. 219–224.

[12] J. Cooper and M. Goodrich, "Towards combining UAV and sensor operator roles in UAV-enabled visual search," in *Proc. IEEE Int. Conf. Human Robot Interaction*, 2008, pp. 351–358.

[13] G. A. Hollinger, B. Englot, F. S. Hover, U. Mitra, and G. S. Sukhatme, "Active planning for underwater inspection and the benefit of adaptivity," *Int. J. Robot. Res.*, vol. 32, no. 1, pp. 3–18, 2013.

[14] J. Denzler and C. Brown, "Information theoretic sensor data selection for active object recognition and state estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 2, pp. 145–157, Feb. 2002.

[15] T. Arbel and F. Ferrie, "Viewpoint selection by navigation through entropy maps," in *Proc. IEEE Int. Conf. Comput. Vision*, 1999, vol. 1, vol. 1, pp. 248–254.

[16] Y. Ye and J. K. Tsotsos, "Sensor planning for 3D object search," *Comput. Vision Image Understanding*, vol. 73, no. 2, pp. 145–168, 1999.

[17] P.-P. Vázquez, M. Feixas, M. Sbert, and W. Heidrich, "Viewpoint selection using viewpoint entropy," in *Proc. Vision Model. Visual. Conf.*, 2001, vol. 1, pp. 273–280.

[18] N. A. Massios and R. B. Fisher, "A best next view selection algorithm incorporating a quality criterion," in *Proc. Brit. Mach. Vision Conf.*, 1998, pp. 78.1–78.10.

[19] Y. Takeuchi, N. Ohnishi, and N. Sugie, "Active vision system based on information theory," *Syst. Comput. Japan*, vol. 29, no. 11, pp. 31–39, 1998.

[20] N. Cao, K. H. Low, and J. M. Dolan, "Multi-robot informative path planning for active sensing of environmental phenomena: A tale of two algorithms," in *Proc. Int. Conf. Auton. Agents Multi-agent Syst.*, 2013, pp. 7–14.

[21] A. Bender, S. B. Williams, and O. Pizarro, "Autonomous exploration of large-scale benthic environments," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013, pp. 390–396.

[22] R. Marchant and F. Ramos, "Bayesian optimisation for informative continuous path planning," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014, pp. 6136–6143.

[23] R. Sim and N. Roy, "Global a-optimal robot exploration in SLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2005, pp. 661–666.

[24] C. Leung, S. Huang, N. Kwok, and G. Dissanayake, "Planning under uncertainty using model predictive control for information gathering," *Robot. Auton. Syst.*, vol. 54, no. 11, pp. 898–910, Nov. 2006.

[25] H. J. S. Feder, J. J. Leonard, and C. M. Smith, "Adaptive mobile robot navigation and mapping," *Int. J. Robot. Res.*, vol. 18, no. 7, pp. 650–668, 1999.

[26] J. Vander Hook, P. Tokekar, and V. Isler, "Cautious greedy strategy for bearing-based active localization: Experiments and theoretical analysis," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 1787–1792.

[27] R. Marchant and F. Ramos, "Bayesian optimisation for intelligent environmental monitoring," in *Proc. IEEE Int. Conf. Intell. Robots Syst.*, Oct. 2012, pp. 2242–2249.

[28] E.-M. Wong, F. Bourgault, and T. Furukawa, "Multi-vehicle Bayesian search for multiple lost targets," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2005, pp. 3169–3174.

[29] C. Stachniss and W. Burgard, "Exploring unknown environments with mobile robots using coverage maps," in *Proc. Int. Joint Conf. Artif. Intell.*, 2003, pp. 1127–1134.

[30] C. Kreucher, J. Wegrzyn, M. Beauvais, and R. Conti, "Multiplatform information-based sensor management: an inverted UAV demonstration," in *Proc. SPIE Defense Transform. Netw.-Centric Syst.*, 2007, vol. 6578, pp. 65780Y-1–65780Y-11.

[31] N. Roy and C. Earnest, "Dynamic action spaces for information gain maximization in search and exploration," in *Proc. Amer. Controls Conf.*, Jun. 2006, p. 6.

[32] W. Lu, G. Zhang, S. Ferrari, R. Fierro, and I. Palunko, "An information potential approach for tracking and surveilling multiple moving targets using mobile sensor agents," in *Proc. SPIE Unmanned Syst. Technol.*, 2011, vol. 8045, pp. 80450T-1–80450T-13.

[33] F. Bourgault, A. A. Makarenko, S. Williams, B. Grocholsky, and H. Durrant-Whyte, "Information based adaptive robotic exploration," in *Proc. IEEE Int. Conf. Intell. Robot. Syst.*, 2002, vol. 1, pp. 540–545.

[34] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol. 22, no. 6, pp. 46–57, Jun. 1989.

[35] A. Singh, A. Krause, C. Guestrin, and W. J. Kaiser, "Efficient informative sensing using multiple robots," *J. Artif. Intell. Res.*, vol. 34, pp. 707–755, 2009.

[36] T. N. Hoang, K. H. Low, P. Jaillet, and M. Kankanhalli, "Nonmyopic ϵ-Bayes-optimal active learning of gaussian processes," in *Proc. Int. Conf. Mach. Learning*, 2014, pp. 739–747.

[37] K. H. Low, J. M. Dolan, and P. Khosla, "Adaptive multi-robot wide-area exploration and mapping," in *Proc. Conf. Auton. Agents Multiagent Systems*, 2008, pp. 23–30.

[38] J. Souza, R. Marchant, L. Ott, D. Wolf, and F. Ramos, "Bayesian optimisation for active perception and smooth navigation," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2014, pp. 4081–4087.

[39] R. Bajcsy, "Active perception," *Proc. IEEE*, vol. 76, no. 8, pp. 996–1005, Aug. 1988.

[40] J. R. Spletzer and C. J. Taylor, "Dynamic sensor planning and control for optimally tracking targets," *Int. J. Robot. Res.*, vol. 22, no. 1, pp. 7–20, 2003.

[41] B. DasGupta, J. P. Hespanha, J. Riehl, and E. Sontag, "Honey-pot constrained searching with local sensory information," *Nonlinear Anal.: Theory, Methods Appl.*, vol. 65, no. 9, pp. 1773–1793, 2006.

[42] G. Zhang and S. Ferrari, "An adaptive artificial potential function approach for geometric sensing," in *Proc. IEEE Int. Conf. Decision Control*, 2009, pp. 7903–7910.

[43] G. Hager and M. Mintz, "Computational methods for task-directed sensor data fusion and sensor planning," *Int. J. Robot. Res.*, vol. 10, no. 4, pp. 285–313, 1991.

[44] G. Benet, F. Blanes, J. Simó, and P. Pérez, "Using infrared sensors for distance measurement in mobile robots," *Robot. Auton. Syst.*, vol. 40, no. 4, pp. 255–266, 2002.

[45] J. Denzler, M. Zobel, and H. Niemann, "Information theoretic focal length selection for real-time active 3d object tracking," in *Proc. IEEE Int. Conf. Comput. Vision*, Oct. 2003, vol. 1, pp. 400–407.

[46] J. Tisdale, Z. Kim, and J. K. Hedrick, "Autonomous UAV path planning and estimation," *IEEE Robot. Autom. Mag.*, vol. 16, no. 2, pp. 35–42, Jun. 2009.

[47] B. Grocholsky, J. Keller, V. Kumar, and G. Pappas, "Cooperative air and ground surveillance," *IEEE Robot. Autom. Mag.*, vol. 13, no. 3, pp. 16–25, Sep. 2006.

[48] W. Lu, G. Zhang, and S. Ferrari, "An information potential approach to integrated sensor path planning and control," *IEEE Trans. Robot.*, vol. 30, no. 4, pp. 919–934, Aug. 2014.

[49] G. Zhang, S. Ferrari, and M. Qian, "An information roadmap method for robotic sensor path planning," *J. Intell. Robot. Syst.*, vol. 56, nos. 1/2, pp. 69–98, 2009.

[50] X. Liao and L. Carin, "Application of the theory of optimal experiments to adaptive electromagnetic-induction sensing of buried targets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 8, pp. 961–972, Aug. 2004.

[51] A. Emery and A. V. Nenarokomov, "Optimal experiment design," *Meas. Sci. Technol.*, vol. 9, no. 6, p. 864, 1998.

[52] D. Ucinski and J. Korbicz, "Path planning for moving sensors in parameter estimation of distributed systems," in *Proc. Workshop Robot Motion Control*, 1999, pp. 273–278.

[53] D. Ucinski, "Optimal sensor location for parameter estimation of distributed processes," *Int. J. Control*, vol. 73, no. 13, pp. 1235–1248, 2000.

[54] R. B. Frieden, *Science from Fisher Information: A Unification*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[55] N. Atanasov, B. Sankaran, J. Le Ny, G. Pappas, and K. Daniilidis, "Nonmyopic view planning for active object classification and pose estimation," *IEEE Trans. Robot.*, vol. 30, no. 5, pp. 1078–1090, Oct. 2014.

[56] Y. F. Li and Z. G. Liu, "Information entropy based viewpoint planning for 3-D object reconstruction," *IEEE Trans. Robot.*, vol. 21, no. 3, pp. 324–327, Jun. 2005.

[57] M. Rahimi, M. Hansen, W. Kaiser, G. Sukhatme, and D. Estrin, "Adaptive sampling for environmental field estimation using robotic sensors," in *Proc. IEEE Int. Conf. Intell. Robot. Syst.*, Aug. 2005, pp. 3692–3698.

[58] G. A. Hollinger and G. S. Sukhatme, "Sampling-based robotic information gathering algorithms," *Int. J. Robot. Res.*, vol. 33, no. 9, pp. 1271–1287, 2014.

[59] A. Ryan and J. K. Hedrick, "Particle filter based information-theoretic active sensing," *Robot. Auton. Syst.*, vol. 58, no. 5, pp. 574–584, 2010.

[60] R. Martinez-Cantin, N. de Freitas, E. Brochu, J. Castellanos, and A. Doucet, "A Bayesian exploration-exploitation approach for optimal online sensing and planning with a visually guided mobile robot," *Auton. Robots*, vol. 27, no. 2, pp. 93–103, 2009.

[61] H. Jacobs, S. Nair, and J. Marsden, "Multiscale surveillance of Riemannian manifolds," in *Proc. Amer. Controls Conf.*, 2010, pp. 5732–5737.

[62] A. D. Wilson, J. A. Schultz, and T. D. Murphey, "Trajectory synthesis for fisher information maximization," *IEEE Trans. Robot.*, vol. 30, no. 6, pp. 1358–1370, Dec. 2014.

[63] D. Song, C.-Y. Kim, and J. Yi, “Simultaneous localization of multiple unknown and transient radio sources using a mobile robot,” *IEEE Trans. Robot.*, vol. 28, no. 3, pp. 668–680, Jun. 2012.

[64] C.-Y. Kim, D. Song, Y. Xu, J. Yi, and X. Wu, “Cooperative search of multiple unknown transient radio sources using multiple paired mobile robots,” *IEEE Trans. Robot.*, vol. 30, no. 5, pp. 1161–1173, Oct. 2014.

[65] L.M. Miller and T.D. Murphey, “Trajectory optimization for continuous ergodic exploration,” in *Proc. Amer. Controls Conf.*, 2013, pp. 4196–4201.

[66] L. M. Miller and T. D. Murphey, “Trajectory optimization for continuous ergodic exploration on the motion group SE(2),” in *Proc. IEEE Int. Conf. Decision Control*, 2013, pp. 4517–4522.

[67] E. Acar, H. Choset, Y. Zhang, and M. Schervish, “Path planning for robotic demining: Robust sensor-based coverage of unstructured environments and probabilistic methods,” *Int. J. Robot. Res.*, vol. 22, nos. 7/8, pp. 441–466, Jul. 2003.

[68] H. Choset, “Coverage for robotics—A survey of recent results,” *Ann. Math. Artif. Intell.*, vol. 31, no. 1-4, pp. 113–126, 2001.

[69] A. Krause and C. Guestrin, “Nonmyopic active learning of gaussian processes: An exploration-exploitation approach,” in *Proc. Int. Conf. Mach. Learning*, 2007, pp. 449–456.

[70] J. Hauser, “A projection operator approach to the optimization of trajectory functionals,” in *Proc. IFAC World Cong.*, 2002, vol. 4, pp. 3428–3434.

[71] G. S. Chirikjian, *Stochastic Models, Information Theory, and Lie Groups, Volume 1: Classical Results and Geometric Methods*. New York, NY, USA: Springer Science & Business Media, 2009.

[72] G. S. Chirikjian and A. B. Kyatkin, *Engineering Applications of Noncommutative Harmonic Analysis*. Boca Raton, FL, USA: CRC Press, 2000.

[73] Y. Bai, J. B. Snyder, M. Peshkin, and M. A. MacIver, “Finding and identifying simple objects underwater with active electrosense,” *Int. J. Robot. Res.*, vol. 1, no. 23, 2015.

[74] S. Thrun, “Learning occupancy grid maps with forward sensor models,” *Auton. Robots*, vol. 15, no. 2, pp. 111–127, 2003.

[75] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge, MA, USA: MIT Press, 2005.

[76] Y. Silverman, L. M. Miller, M. A. MacIver, and T. D. Murphey, “Optimal planning for information acquisition,” in *Proc. IEEE Int. Conf. Intell. Robots Syst.*, 2013, pp. 5974–5980.

[77] M. E. Nelson and M. A. MacIver, “Sensory acquisition in active sensing systems,” *J. Comparative Physiol. A*, vol. 192, no. 6, pp. 573–586, 2006.

[78] Y. Zhou and G. S. Chirikjian, “Probabilistic models of dead-reckoning error in nonholonomic mobile robots,” in *Proc IEEE Int. Conf. Robot. Autom.*, 2003, vol. 2, pp. 1594–1599.

[79] Y. Wang and G. Chirikjian, “A diffusion-based algorithm for workspace generation of highly articulated manipulators,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2002, vol. 2, pp. 1525–1530.

[80] E. Todorov and W. Li, “A generalized iterative lqg method for locally-optimal feedback control of constrained nonlinear stochastic systems,” in *Proc. Amer. Controls Conf.*, 2005, pp. 300–306.

[81] M. B. Horowitz and J. W. Burdick, “Semidefinite relaxations for stochastic optimal control policies,” in *Proc. Amer. Controls Conf.*, 2014, pp. 3006–3012.

[82] K. Zhou and J. C. Doyle, *Essentials of Robust Control*, vol. 180. Upper Saddle River, NJ, USA: Prentice-Hall, 1998.

[83] A. Prabhakar, K. Flaßkamp, and T. Murphey, “Time discretization in optimal ergodic control,” in *Proc. IEEE Int. Conf. Decision Control*, in press.

Lauren M. Miller received the joint A.B./B.E. degree in engineering science from Dartmouth College, Hanover, NH, USA, in 2009, and the M.S. and Ph.D. degrees from Northwestern University, Evanston, IL, USA, in 2013 and 2015, respectively.

She is a Postdoctoral Researcher with the University of California, Berkeley, CA, USA. Her research interests include optimal control, autonomous search and exploration, and active information acquisition.

Dr. Miller is currently the Chair of the IEEE Robotics and Automation Society Student Activities Committee and a Member of the RAS Administrative Committee.

Yonatan Silverman received the B.S. degree from Johns Hopkins University in 2010 and the M.S. degree from Northwestern University, Evanston, IL, USA, in 2013.

He is currently working in software development with McMaster-Carr, Elmhurst, IL.

Malcolm A. MacIver received the B.Sc. degree in computer science and the M.A. degree in philosophy from University of Toronto, Toronto, ON, Canada, in 1991 and 1992, respectively, and the Ph.D. degree in neuroscience from University of Illinois at Urbana-Champaign, Champaign, IL, USA, in 2001.

He was a Postdoctoral Fellow in mechanical engineering and the Computation and Neural Systems Program with California Institute of Technology, Pasadena, CA, USA, from 2001 to 2003. He joined Northwestern University, Evanston, IL, USA, in 2003 with a joint appointment with the Department of Mechanical Engineering and the Department of Biomedical Engineering, where he is currently an Associate Professor. His research interests include the mechanical and neural basis of animal behavior, particularly the intersection of information harvesting and biomechanics, utilizing a variety of approaches spanning behavioral analysis, computer simulations, robotics, neurobiology, and mechanics.

Todd D. Murphey received the B.S. degree in mathematics from University of Arizona, Tucson, AZ, USA, and the Ph.D. degree in control and dynamical systems from California Institute of Technology, Pasadena, CA, USA.

He is an Associate Professor of mechanical engineering with Northwestern University, Evanston, IL, USA. His laboratory is part of the Neuroscience and Robotics Laboratory. His research interests include robotics, control, computational methods for biomechanical systems, and computational neuroscience.

Dr. Murphey received the National Science Foundation CAREER Award in 2006, membership in the 2014-2015 DARPA/IDA Defense Science Study Group, and Northwestern’s Professorship of Teaching Excellence. He is a Senior Editor of IEEE TRANSACTIONS ON ROBOTICS.