

Portland State University

PDXScholar

Dissertations and Theses

Dissertations and Theses

4-15-2024

Adaptive Sampling for Seabed Identification from Ambient Acoustic Noise

Matthew Sullivan

Portland State University

Follow this and additional works at: https://pdxscholar.library.pdx.edu/open_access_etds



Part of the [Engineering Commons](#)

Let us know how access to this document benefits you.

Recommended Citation

Sullivan, Matthew, "Adaptive Sampling for Seabed Identification from Ambient Acoustic Noise" (2024).
Dissertations and Theses. Paper 6615.
<https://doi.org/10.15760/etd.3747>

This Thesis is brought to you for free and open access. It has been accepted for inclusion in Dissertations and Theses by an authorized administrator of PDXScholar. Please contact us if we can make this document more accessible: pdxscholar@pdx.edu.

Adaptive Sampling for Seabed Identification from Ambient Acoustic Noise

by

Matthew Sullivan

A thesis submitted in partial fulfillment of the
requirements for the degree of

Master of Science
in
Electrical and Computer Engineering

Thesis Committee:
John Lipor, Chair
Bruno Jedynak
Martin Siderius

Portland State University
2024

Abstract

This thesis discusses distinguishing geoacoustic properties of seabeds by adaptively sampling ambient acoustic measurements using an autonomous underwater vehicle. Modern advancements in seabed characterization motivate an online method for distinguishing between two seabed types using Fourier transformed ambient acoustic noise snapshots. These snapshots are assumed to be associated with their spatial location and seabed, making the goal to organize unlabeled seabed locations. The snapshots are transformed to obtain pairwise similarities between locations. Locations with similarities exceeding a threshold are classified together with the goal of identifying all locations with similarities below this threshold using a process known as level set estimation. We propose an adaptive sampling policy that aims to directly reduce the number of locations with unknown level set membership via a lookahead step in addition to minimizing the distance traveled through a nearest neighbors approach. Results on synthetic, single-boundary fields and multi-boundary realistic world data are evaluated by comparing predicted and true level set assignments and demonstrate the benefits of the proposed algorithm. Furthermore, extensions discussing the effects of a two-step and path planning lookahead in addition to the benefits of bathymetry data are included to motivate further research.

To mom and dad

Acknowledgments

I would like to first thank my advisor, Dr. John Lipor, for his support. I could not have undertaken this training without his guidance, patience, and feedback. I would like to express gratitude to Dr. Bruno Jedynak, who significantly expanded my understanding of mathematics, and Dr. Martin Siderius, whose impact has touched every corner of PSU's ECE Department.

It is inarguable that my progression through PSU's ECE graduate program has significantly impacted me in many positive ways. I owe many thanks to Dr. James McNames, whose teachings let me grow as an engineer. I would also like to thank Alex Higgins and Sean O. Stalley, both of whom I looked up to for mentorship. I am also grateful for Dr. Lipor specifically for proofreading my thesis. I am especially grateful for my classmates, whose wittiness and curiosity sparked many fascinating conversations. I feel very fortunate to have had the opportunity to meet them.

Finally, I would like to thank my family for their unending support that allows me to experience opportunities like these.

Table of Contents

Abstract	i
Dedication	ii
Acknowledgments	iii
List of Tables	vi
List of Figures	vii
1 Introduction	1
1.1 Problem Formulation	4
1.2 Contributions	6
2 Literature Review	8
2.1 Introduction	8
2.2 Seabed Characterization	8
2.3 Adaptive Sampling for Level Set Estimation	13
3 Lookahead Uncertain Set Reduction	22
4 Results	28
4.1 Lookahead Sampling	29
4.2 Synthetic Results	31
4.3 Seabed Results	34

5	Extensions	38
5.1	Second-Order LUSR	39
5.2	Continuous Sampling	41
5.3	Bathymetry Data	43
6	Conclusion	47
7	Bibliography	49
8	Appendix: Lookahead Error Bound	56

List of Tables

4.1	Sampling Policies	29
4.2	Mean and median time (s) to compute the 100 th sample with 40 neighbors	36
5.1	Mean and median time (s) to compute the 100 th sample with 40 neighbors	46

List of Figures

1.1	Seabed types near Alaska, USA. Each color represents a seabed type characterized by its geoacoustic properties discussed in Section 2.2.	3
2.1	Seabed types in the northern Pacific Ocean. The seabed consists of a dominant material such as clay (yellow) or medium silt (green). The goal is to discover all locations where the sediment type differs significantly (light/dark blue). The red box denotes the subregion used for experiments in Chapter 4	9
2.2	Estimated mean (black) with confidence widths (shaded) over the true mean (blue). The true mean is shown in blue with the estimate mean (dashed) wavering about it. The measured locations (red) reduce the uncertainty (shaded) about the location of the true mean evident by the dips. Locations that are not near sampled locations are shown to hold more uncertainty.	18
4.1	The solid lines reflect the median value over 6 trials for the first 12 randomly generalized boundaries discussed in Section 4.2. The interquartile range is shaded. In the low noise setting, 20 neighbors (a), 50 neighbors (b), and 80 neighbors (c) all show pessimistic sampling outperforming optimistic and mean sampling by reaching a higher accuracy with lower cost.	30
4.2	Twenty randomly simulated fields used to evaluate each policy (a) and twenty additional fields, each with two boundaries (b). The reference location is the top left of each boundary with yellow indicating the superlevel set and purple indicating the sublevel set. . . .	31

4.3	Accuracy compared to cost for the original synthetic fields. The solid lines reflect the median value over 32 trials for each of the boundaries. The interquartile range is shaded. In a low noise setting with 100 neighbors (a), LUSR achieves a high accuracy with very little cost compared to its competitors. In the high noise setting (b), LUSR moves a bit close to its competitors while still attaining a higher accuracy and lower cost.	33
4.4	Accuracy compared to cost for the additional synthetic fields over 150 samples. In a low noise setting with 100 neighbors (a), LUSR still achieves a higher accuracy than its competitors. In the high noise setting (b), LUSR begins to struggle and performs similarly to VAR.	34
4.5	At the end of 100 iterations, LUSR's median accuracy (a) outperforms the other algorithms with 15 neighbors. Similarly, it travels less than VAR and slightly more than margin. Margin's favorability for the decision boundary helps keep a low traveling cost.	35
4.6	LUSR and VAR perform similarly (a), however, LUSR gets more of an edge over VAR between ,5000 - 7,500 km by finding an unexplored boundary earlier. This is shown in LUSR's uncertain set size (b).	37
5.1	LUSR-2 is shown performing marginally better with the current mean estimate (orange) than one using a lookahead measurement (blue). The lookahead error bound (3.2) is worse than the current mean estimate, justifying the worse performance using the updated mean.	40
5.2	With 40 nearest neighbors, LUSR-2 reaches a near perfect accuracy with less cost than other algorithms (a). The cardinality, however, performed worse (b). Further testing is required if LUSR-2 maintains a higher accuracy with a lower cardinality or obtains a lower cardinality and the higher accuracy in (a) is due to simple boundaries.	41
5.3	Executed paths computed using LUSR Path over a smaller region in the norther Pacific Ocean using $\rho = 20$	42

5.4	Due to the wide sampling range, LUSR Path avoids a dip that LUSR falls for (b) and (c). Despite this, they generally track each other while LUSR Path takes significantly more computation time – often taking days or weeks longer than LUSR.	43
5.5	New region for LSE (a) with its corresponding bathymetry data (b) where white designates land. The seabed types shown in (a) is shown to be correlated to the depth of the ocean in (b).	44
5.6	The results generally track when adding in the new kernel. The new kernel bumps accuracy to .9 quickly (f), and the adaptive sampling policies push past that at the same rate.	45

1 Introduction

Identifying, bounding, and monitoring the spatial extent of an environmental phenomenon has seen significant advancements through the introduction of mobile sensors like unmanned aerial vehicles and autonomous surface vessels [1, 2, 3, 4]. Mobile sensors allow automated and adaptive sampling techniques estimate environmental parameters accurately and efficiently. One increasingly significant problem that could benefit from mobile sensors is estimating seabed parameters [5, 6] which are important for sonar performance [7]. Sonar equation calculations are moving away from previous assumptions of fixed environmental parameters [7, 8] towards dynamic ones [8, 9], where once fixed parameters like grain size and density of seabed sediment now change in response to actual conditions.

Previous seabed characterization solutions found success distinguishing seabeds using their geoacoustic properties and machine learning models with comprehensive datasets, a process known as passive learning [10, 11, 12]. However, passive learning performance is constrained to information captured in a given data set. As a result, it can be difficult to diagnose whether a model performs poorly due to ineffective design or insufficient information contained in the data set [13].

Adaptively sampling an environment falls under a subfield of machine learning called active learning, where a learning algorithm actively queries an environment for data [1]. In passive modeling, the model uses a fixed input data set with known

output predictions called labels. For example, a previous seabed characterization solution collected pressure field data for gravel, sand, silt, and clay sediment types. The passive learning model, composed of a neural network, would receive a pressure field and predict a sediment type [12]. The model’s weights update on a measure of error between the predicted label (e.g. clay) and the known true label. The underlying assumption for active learning relies on an active algorithm outperforming its passive competitors by allowing it to decide which training points it wants labeled [14].

Active learning algorithms work by iteratively assessing a quantified definition of information gain before asking an external source, such as a human or the environment, for a labeled data point. These algorithms may also penalize information gain with an associated cost, such as traveling time or labeling costs. Environmental monitoring has seen benefits of adaptive sampling through successful bounding of the spatial spread of, for example, thermoclines [15, 16, 17], concentrations of harmful chemicals and bacteria [18, 19, 2], and wildfires [4].

Recent advancements in seabed parameter estimation have demonstrated improvement through the use of a binary hypothesis test. In this approach, a sample covariance from ambient acoustic snapshots is compared to the true covariance of a perfectly characterized reference seabed type. The binary hypothesis test classifies locations using the generalized likelihood ratio test, which is equivalent to the Kullback-Leibler divergence between a sample covariance and a perfectly known reference covariance [13]. This motivates an autonomous underwater vehicle (AUV) to adaptively sample and identify the spatial extent of the reference

seabed.

While ambient noise is free to measure, even a low-powered AUV cannot travel over huge oceanic regions within a reasonable amount of time. Furthermore, a given seabed type can show up in multiple disconnected regions, shown in Figure 1.1.

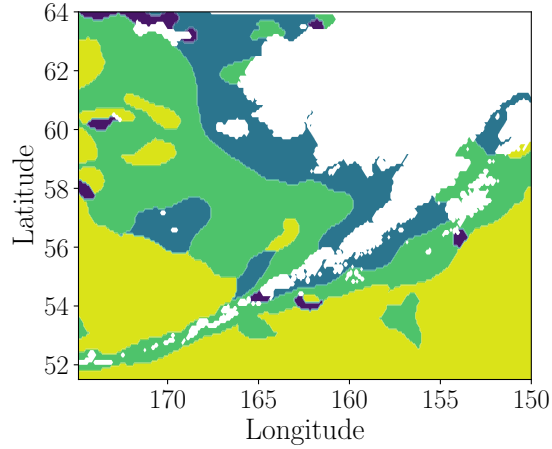


Figure 1.1: Seabed types near Alaska, USA. Each color represents a seabed type characterized by its geoacoustic properties discussed in Section 2.2.

As a result, the AUV must discover boundaries separating regions with snapshots similar to the reference from regions with dissimilar snapshots. Therefore, it is a primary concern that an adaptive sampling strategy recommends decisions to efficiently balance information gain against the costs associated with traveling.

When recommending locations to sample, existing environmental monitoring algorithms consider either local or global views. Local views aim to track a single boundary precisely by sampling nearby regions [15, 20, 17], making them efficient and noise tolerant. They are limited to a case of a single boundary without appropriate means to explore new regions of interest. Using the example of Figure 1.1,

a local view algorithm would struggle to jump between the clay (yellow) islands, where a single island is contained within one boundary. Global views consider the entire field of interest [18, 19, 1, 21, 22, 23], allowing them to discover multiple disconnected boundaries. They perform poorly in high noise environments, and this poor performance often results from over-exploration. By considering the entire field, these approaches may inefficiently sample by jumping across the environment multiple times. The possibility of the reference seabed type showing up in multiple disconnected regions combined with snapshots being an indirect measurement of characterizing seabed parameters motivates a global view algorithm capable of organizing locations from unlabeled seabed parameters.

1.1 Problem Formulation

Combining adaptive sampling with seabed characterization, we consider an algorithm capable of active learning for seabed type identification using the Kullback-Leibler divergence (KLD) to obtain similarities between two locations. Specifically, consider an AUV carrying an array of M receivers continuously capturing ambient acoustic noise, which is assumed to be associated with the vehicle's current location. The time-series pressure recordings are Fourier transformed to obtain snapshots, $z \in \mathbb{C}^M$, at a single frequency.

Within a finite domain of interest, $\mathcal{D} \subset \mathbb{R}^2$, a learning algorithm obtains L snapshots from each visited location. At a given time, these snapshots can be used to form an estimated covariance matrix, $\hat{\Sigma}_t = \frac{1}{L} \sum_{i=1}^L z_i z_i^H$. At the start, the

algorithm first samples a reference location to form $\hat{\Sigma}_0$. As the AUV travels, it will compare its most recent estimated covariance location, $\hat{\Sigma}_t$, to $\hat{\Sigma}_0$ and assess if the seabed types are equivalent. Previous work uses the KLD as a method to compute the generalized likelihood ratio test (GLRT) between a perfectly known distribution and a sample covariance [13]. However, because the Kullbeck-Leibler divergence is asymmetrical, we symmetrize it

$$J\left(\hat{\Sigma}_0\|\hat{\Sigma}_t\right)=\left(D\left(\hat{\Sigma}_0\|\hat{\Sigma}_t\right)+D\left(\hat{\Sigma}_t\|\hat{\Sigma}_0\right)\right) / 2, \quad (1.1)$$

where $D(\cdot\|\cdot)$ represents the Kullbeck-Leibler divergence discussed in Section 2.2. The resulting equation (1.1) is known as the Jefferys divergence, and it is used to quantify separation between our two sample covariances, $\hat{\Sigma}_t$ and $\hat{\Sigma}_0$. At each location, x , we seek to obtain the similarity with reference location, x_0 , defined as $s(x_0, x)$. The goal is to accurately estimate the set of locations sufficiently dissimilar to the reference location [22]

$$\mathcal{S}_\tau=\left\{x \in \mathcal{D}: s\left(x_0, x\right) \leq \tau\right\}, \quad (1.2)$$

where $\tau \geq 0$ is a user-supplied threshold. In our setting, we obtain pairwise noisy similarity measurements,

$$s_t=\exp \left(-J\left(\hat{\Sigma}_0\|\hat{\Sigma}_t\right) / \ell^2\right), \quad (1.3)$$

where ℓ is a tuning parameter controlling the scale of the similarities. Estimating

a solution for (1.2) returns a set of locations with similarities below the threshold – or level – in a process known as level set estimation. A successful algorithm is capable of achieving an estimated level set \hat{S} with the following loss [22]

$$\mathcal{L}_{LSE}(S_\tau, \hat{S}_\tau) = \left| (S_\tau \cap \hat{S}_\tau^C) \cup (S_\tau^C \cap \hat{S}_\tau) \right|, \quad (1.4)$$

where S_τ^C is the complementary set to S such that $S \cup S^C = \mathcal{D}$. The threshold governs the degree of dissimilarity required to claim that location x is sufficiently different than reference location x_0 , indicating they are two separate seabed types and on different levels. As our algorithm samples, it moves inefficiently by doubling back and sampling near previous locations. To avoid excessive traveling, the AUV must sample from its nearest neighbors. As a result, we wish to minimize the distance traveled while obtaining a sufficiently accurate estimation of (1.2). Our problem may be formulated as

$$\min_{x_1, \dots, x_T} \sum_{t=1}^T |x_t - x_{t-1}| \quad s.t. \quad \mathcal{L}_{LSE}(S_\tau, \hat{S}_\tau) < \varepsilon. \quad (1.5)$$

1.2 Contributions

This thesis proposes an approach for adaptive sampling with an autonomous underwater vehicle to determine seabed type by formulating seabed identification as level set estimation from pairwise noisy similarity measurements.

The main contributions of this thesis are:

1. Transforming the problem of seabed characterization into level set estimation

2. An adaptive sampling policy to directly minimize the number of locations with unknown level set membership via a lookahead operation
3. Distance penalization using a nearest neighbors approach
4. Empirical results on synthetic and realistic seabed types data
5. Analysis on the lookahead operation trade-offs
6. Empirical results and analysis using continuous path sampling and two-step lookahead policies
7. Empirical results using a combination kernel composed of bathymetry and spatial similarities

2 Literature Review

2.1 Introduction

The following chapter begins by discussing modern advancements in seabed type identification from ambient acoustic noise using a binary hypothesis test between a completely known reference distribution and currently sampled ambient noise [13]. This motivates an adaptive sampling algorithm capable of assessing seabed similarities using two sample covariances, where similarities exceeding a threshold indicate two seabed types are equivalent with high probability. This transforms seabed identification into level set estimation. As a result, the section concludes by discussing modern adaptive sampling methods for level set estimation and explaining how it points towards a policy that uses a lookahead method to directly reduce the number of uncertain locations.

2.2 Seabed Characterization

Using the same AUV problem defined in Section 1.1, the goal is to assess if two separate locations contain seabed types with sufficiently different geoacoustic properties [13]. The Naval Oceanographic Office maintains a bottom sediment type database listing 23 seabed types provided in the High Frequency Environmental Acoustics (HFEVA) data set [24]. Seabeds are often composed of a predominant

background type, like clay, with patches of different sediment types scattered and distributed throughout [13], as Figure 2.1 demonstrates.

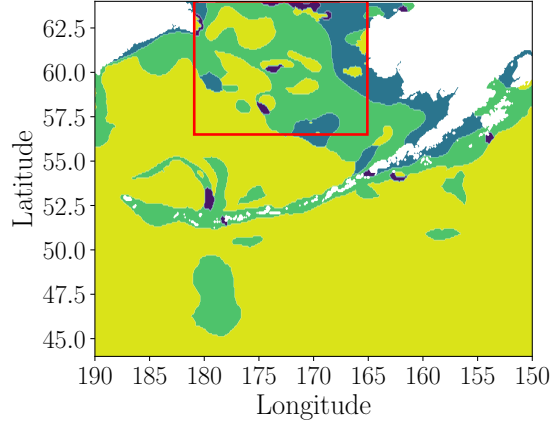


Figure 2.1: Seabed types in the northern Pacific Ocean. The seabed consists of a dominant material such as clay (yellow) or medium silt (green). The goal is to discover all locations where the sediment type differs significantly (light/dark blue). The red box denotes the subregion used for experiments in Chapter 4 .

Modern advancements in seabed identification from ambient acoustic snapshots use the Kullback-Leibler divergence (KLD) as the generalized likelihood ratio test comparing a sample covariance derived from snapshots against a completely known reference covariance [13]. Each snapshot, z_1, \dots, z_L , is assumed to be drawn from a circularly-symmetric complex Gaussian distribution with additive covariance

$$\Sigma_\theta = \mathbb{E}[zz^H] = \sigma_s^2 \Gamma_\theta + \sigma_n^2 I, \quad (2.1)$$

where σ_s^2 is the ambient noise power, σ_n^2 is the non-acoustic sensor noise variance, and Γ_θ is the signal covariance matrix characterized by non-random seabed parameters, θ [25, 26]. The probability density function for a circularly-symmetric

complex Gaussian distribution is known to be

$$f(x; \theta) = \det(\pi \Sigma_\theta)^{-1} \exp(-x^H \Sigma_\theta^{-1} x), \quad (2.2)$$

where x^H refers to the conjugate transpose of x .

With reference seabed parameters, $\theta_0 \in \Theta_0$, a binary hypothesis test assesses if current seabed parameters, $\theta_1 \in \Theta_1$, are the same, $H_0 : \theta_1 = \theta_0$, or sufficiently different, $H_1 : \theta_1 \neq \theta_0$. Specifically, Θ_0 and Θ_1 are mutually exclusive sets in \mathbb{R}^N and contain all possible parameter vectors. Using snapshots, a test statistic $\Lambda(S)$ decides when to reject H_0 , the null hypothesis, in favor for H_1 , the alternative hypothesis,

$$\Lambda(S) \underset{H_0}{\overset{H_1}{\geq}} \xi, \quad (2.3)$$

where ξ results from setting the false alarm rate, e.g., $P_{H_0}(\Lambda(S) > \xi) = 10^{-4}$.

The generalized likelihood ratio test (GLRT) is a suitable approach when the alternative hypothesis relies on unknown parameters [27]. The GLRT statistic, $\Lambda(S)$, using recently collected L snapshots is defined as

$$\Lambda(z_1, \dots, z_L) = 2 \log \frac{\sup_{\theta_1 \in \Theta_1} \prod_{i=1}^L f(z_i, \theta_1)}{\sup_{\theta_0 \in \Theta_0} \prod_{i=1}^L f(z_i, \theta_0)}. \quad (2.4)$$

Since the reference parameter is completely known, the denominator can be reduced to $\theta_0 = \Theta_0$, making the numerator a maximum likelihood estimate [13]. In the case of circularly-symmetric complex Gaussian distributions, the GLRT

simplifies to

$$\Lambda(x_1, \dots, x_L) = 2L \left(\log \left(\frac{\det(\Sigma_0)}{\det(\widehat{\Sigma})} \right) - M + \text{tr} \left(\Sigma_0^{-1} \widehat{\Sigma} \right) \right), \quad (2.5)$$

where Σ_0 is characterized by θ_0 and $\widehat{\Sigma} = \frac{1}{L} \sum_{i=1}^L z_i z_i^H$ is the sample covariance belonging to unknown seabed parameters, θ_1 .

Eq. (2.5) is identical to the Kullbeck-Leibler divergence between two circularly-symmetric complex Gaussians. The KLD (2.9) can be viewed as a comparison of entropy,

$$H(\theta) = - \int_{z \in \mathcal{Z}} f(z; \theta) \log(f(z; \theta)) dz, \quad (2.6)$$

with cross entropy,

$$H(\theta_1; \theta_0) = - \int_{z \in \mathcal{Z}} f(z; \theta_0) \log(f(z; \theta_1)) dz. \quad (2.7)$$

Both are non-negative measures of uncertainty [28]. One can expect (2.6) to diminish if θ fits $f(z; \theta)$ and increase if it does not. As a result, the KLD, a comparison of entropy and cross entropy, can be viewed as a logarithmic difference,

$$\begin{aligned} D_{KL}(\theta_0 \parallel \theta_1) &= H(\theta_1; \theta_0) - H(\theta_0) \\ &= - \int_{z \in \mathcal{Z}} f(z; \theta_0) \log(f(z; \theta_1)) dz + \int_{z \in \mathcal{Z}} f(z; \theta_0) \log(f(z; \theta_0)) dz \\ &= \int_{z \in \mathcal{Z}} f(z; \theta_0) \log \left(\frac{f(z; \theta_0)}{f(z; \theta_1)} \right) dz, \end{aligned} \quad (2.8)$$

which is an asymmetrical test. As seen above, the entropy and cross entropy are defined by the true probability density function, $f(z; \theta_0)$, with no assumptions made about $f(z; \theta_1)$. This allows the KLD to act as a measure of distance between $f(z; \theta_0)$ and $f(z; \theta_1)$.

Between a completely known reference θ_0 and completely known current location seabed parameter θ_1 , the KLD between two circularly-complex symmetric Gaussian distributions results in (2.5) exactly

$$D(\theta_0||\theta_1) = \text{tr}(K_1^{-1}K_0) - M + \log\left(\frac{\det(K_1)}{\det(K_0)}\right) \quad (2.9)$$

In the case of the current seabed parameters being unknown, the GLRT (2.5) can be replaced with the KLD, $D(\theta_0||\theta_1)$ (2.9), resulting in

$$\Gamma(z_1, \dots, z_L) = 2LD\left(\hat{\theta}||\theta_0\right), \quad (2.10)$$

where θ_0 is completely known and $\hat{\theta}$ are the parameters under (2.4) using the current sample covariance $\hat{\Sigma}_t$ [13]. The estimated error in the sample covariance implies that the KLD must have a discrepancy of at least c/\sqrt{L} for some $c > 0$ in order to accurately assess if the two distributions are separate [29].

Our goal becomes obtaining a subset of seabed from a finite region that is sufficiently different from a reference type using the KLD. These types of problems can be solved using level set estimation: the problem of approximating all points in an environment that exceed some threshold, τ , by leveraging noisy measurements [30], as defined in Eq. (1.2).

2.3 Adaptive Sampling for Level Set Estimation

Active learning is a subfield of machine learning used to generate a useful model by querying for data that was previously collected or needs to be collected [1]. Similarly, online learning focuses on making sequences of optimal decisions across many rounds as data becomes available [31]. Within a single round, an online learning algorithm will consider information gain and a function to optimize, recommend a location, sample the location, and incur a reward or loss. Previous active and online learning approaches to environmental monitoring assessed and updated small, local regions of the environment in a single iteration [2, 4, 20, 16] or globally [1, 19, 21, 32].

Local approaches often consider a subsection of possible options and update a small portion of the estimated environment at once [20, 16, 4, 2]. These approaches often focus on greedily accruing information, which results in failure to detect multiple boundaries between seabed types, previously seen in Figure 1.1. They sample in intervals along a straight line as distance penalization, and directly search for a change point in the domain.

In comparison, global approaches update the entire domain in a single iteration, allowing for multiple boundaries to be detected. These approaches are often implemented using the multi-armed bandit framework: a statistical game with the goal of accumulating the most money from slot machines, referred to as ‘arms’ due to their history of being called one-armed bandits [33].

Stochastic multi-armed bandits, one of the earliest multi-armed bandit prob-

lems, uses a learner to locate the highest paying arm, a^* , out of all possible arms, $\mathcal{A} = \{a_1, \dots, a_N\}$. During round t , a learner must select one arm to play, $a_t \in \mathcal{A}$, and observe reward, r_{a_t} with the goal of accumulating highest total reward, r^* . One of the earliest sampling policies for this game is called Explore-then-Commit (ETC), where a learner plays each arm at least once before committing to the estimated highest paying arm. While exploiting one winnable arm may yield larger rewards, it fails to explore the possibility of higher dividend arms.

To balance exploitation and exploration, the Upper Confidence Bound (UCB) sampling policy tells the learner to sample the arm with the greatest UCB [33]. In this approach, any arm, a , has a UCB composed of a mean, μ and a measure of uncertainty, σ ,

$$\text{UCB of } a = \begin{cases} \infty & t = 0 \\ \mu_t(a) + \sqrt{\sigma_t(a)} & t > 0 \end{cases} \quad (2.11)$$

where t is the round number, $\mu_t(a)$ is the average reward from arm a , and $\sigma_t(a) = \frac{\ln(\Delta t_a)}{n_a}$ with Δt_a counting the number of rounds since arm a has been sampled and n_a counts the number of times arm a has been sampled. This allows the UCB algorithm to exploit a few arms with higher means while occasionally exploring all other arms, a principle known as 'optimism in the face of uncertainty' [33]. There are many other sampling policies possible, such as sampling the arm with the maximal uncertainty, $\sigma_t(a)$. All, however, must form a balance between exploiting high rewards, $\mu_t(a)$, and exploring uncertainty, $\sigma_t(a)$.

Contextual bandits condition rewards on feature vectors associated with each

arm. By basing rewards on certain conditions, there is an embedded similarity structure between arms with similar feature vectors. In seabed characterization, spatial locations and bathymetry could be useful feature vectors as locations near each other and at the same depth are expected to have similar seabed characteristics. In linear contextual bandits, the expected reward is linearly related to the features, $c_a \in \mathbb{R}^d$, with an unknown parameter, ω^* , $\mathbb{E}[r_{a_t}|c_{a_t}] = c_{a_t}^T \omega^*$. Kernelizing contextual bandits allows for similarities between arms to be explicitly or implicitly computed using a kernel function to reduce the number of plays needed to identify high paying arms [34]. This can result in nonlinear relationships between $c_{a,t}$ and ω^* .

Kernel methods assume a mapping $\phi : \mathbb{R}^d \rightarrow \mathcal{H}$ that maps finite dimensional data $\mathcal{X} \subset \mathbb{R}^d$ to a Hilbert space \mathcal{H} where the inner product can be computed using the original feature vectors. Assuming such a Hilbert space exists, a positive definite kernel function is defined by the inner product associated with \mathcal{H}

$$k(x, y) = \phi(x)^T \phi(y), \forall x, y \in \mathbb{R}^d. \quad (2.12)$$

We view the kernel function as a measure of similarity between data $x, y \in \mathcal{X}$. As a result, the expected reward becomes linear with respect to the Hilbert space mapped data, $\mathbb{E}[r_{a,t}|x_{a,t}] = \phi(x_{a,t})^T \omega^*$, and non-linear in \mathbb{R}^d . The quadratic kernel is a clear example with its mapping defined by $\phi(x) = (x^2, \sqrt{2}x, 1)$ and a kernel

$$k(x, y) = (x^T y)^2 + 2x^T y + 1, \quad (2.13)$$

that is linear in the Hilbert space and polynomial in R^d .

Many researchers have found success in adaptive sampling using level set estimation and Gaussian processes to simulate the environment [1, 19, 21, 32, 35]. Level set estimation, the problem of approximating all points in an environment that exceed some threshold, τ , utilizes multi-armed bandits by viewing all possible sampling locations as arm where the reward is the measured specified phenomena. The process of automating sampling patterns to perform level set estimation is known as adaptive sampling. Gaussian processes are fully specified by a mean function, $\mu(x)$, and covariance function, $k(x, x')$ [28].

In these approaches, a learner queries the environment and iteratively samples for a phenomenon of interest. Upon receiving a measurement, it updates a current mean estimate (2.14) and confidence width (2.15) [34]. To form the mean and confidence width estimates, let $y_t = [r_1, \dots, r_t]$ contain the observed rewards, $k_{x,t} = [k(x, x_1), \dots, k(x, x_t)]$, and $K_t = \{k(x_i, x_j)\}_{i,j \leq t}$ then

$$\mu_t(x) = k_{x,t}^T (K_t + \gamma I)^{-1} y_t \quad (2.14)$$

and

$$\sigma_t(x) = \gamma^{-1/2} \sqrt{k(x, x) - k_{x,t}^T (K_t + \gamma I)^{-1} k_{x,t}}, \quad (2.15)$$

where $\gamma \geq 0$ is a regularization parameter. In performing level set estimation, it is common to maintain estimates of superlevel, sublevel, and uncertain sets. The superlevel and sublevel sets are each defined as locations with upper confidence bounds, $\mu_t(x) + \sqrt{\sigma_t(x)}$, and lower confidence bounds, $\mu_t(x) - \sqrt{\sigma_t(x)}$, on the

same side of the threshold, τ , indicating their true level set membership is likely known. The uncertain set maintains locations with unknown level set membership with upper and lower confidence bounds on opposing sides of τ . Therefore, the definitions for the superlevel, sublevel, and uncertain set are respectively,

$$H_t = \{x \in \mathcal{D} : \mu_t(x) - \eta\sigma_t(x) \geq \tau\} \quad (2.16)$$

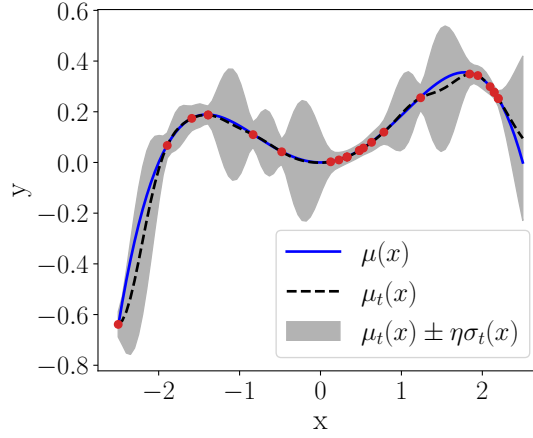
$$L_t = \{x \in \mathcal{D} : \mu_t(x) + \eta\sigma_t(x) \leq \tau\} \quad (2.17)$$

$$U_t = \{x \in \mathcal{D} : |\mu_t(x) - \tau| \leq \eta\sigma_t(x)\}, \quad (2.18)$$

where $\eta > 0$ is a parameter controlling confidence widths. For all points in the domain, the estimated mean and confidence width can be used to form a confidence interval,

$$C_t(x) = [\mu_{t-1}(x) \pm \eta_t\sigma_{t-1}(x)], \quad (2.19)$$

representing uncertainty about the true mean. Shown in Figure 2.2, the estimated mean and confidence widths approach the true mean when sampled. Uncertainty around sampled locations also reduces as they have associated feature vectors, which is their location on the x-axis.



(a)

Figure 2.2: Estimated mean (black) with confidence widths (shaded) over the true mean (blue). The true mean is shown in blue with the estimate mean (dashed) wavering about it. The measured locations (red) reduce the uncertainty (shaded) about the location of the true mean evident by the dips. Locations that are not near sampled locations are shown to hold more uncertainty.

The equations (2.14) and (2.15) are identical across kernel ridge regression and Gaussian processes. Kernel ridge regression is derived using regularization – specifically using the matrix inversion lemma to kernelize least squares. The Gaussian process view of (2.14) and (2.15) is one of a linear combination of multivariate Gaussian random variables, where the confidence interval (2.19) contains all reasonable prior functions. The empirical results are equivalent while the interpretations of the equations differ.

Gotovos et. al [1] introduce the use of Gaussian processes for level set estimation in an algorithm known as GP-LSE. This pivotal paper serves a baseline for global view level set estimation with mean and uncertainty updates coming from (2.14) and (2.15). Its policy – an empirical formula for recommending sampling

locations – selects current uncertain locations, $x \in U_t$, with maximal ambiguity

$$a_t(x) = \min\{\max(\dot{C}_t(x)) - \tau, \tau - \min(\dot{C}_t(x))\}, \forall x \in U_t \quad (2.20)$$

where $\dot{C}_t(x)$ is the intersection of previous confidence intervals, $\dot{C}_t(x) = \bigcap_{i=1}^t C_i(x)$. Ambiguity sampling minimizes the uncertain location with the largest confidence width from the threshold, encouraging uncertain locations to be assigned to the superlevel or sublevel set. It also proposes approaches for implicit thresholds, where the threshold is expressed as a percentile, $\tau = \alpha \max_{x \in \mathcal{D}} \mu(x)$, $\alpha \in [0, 1]$, batch sampling, which obtains measurements after sampling all B locations, and path planning, which uses a traveling salesperson solution to sample batches efficiently.

Various path planning approaches build upon GP-LSE, some which include selecting paths by information gain using dynamic programming [18, 36]. In these approaches, paths are quantified using a measure of information gain such as conditional mutual information or cross entropy. These approaches focus on exploration and implemented a receding horizon to reduce exploitation.

Other problems involve maintaining a set of potential maximizers for exploitation, M_t , [35, 19], which contains points with UCB greater than the largest lower bound. In [35], the problem requires sampling points deemed sufficiently larger than the threshold, S_t . As a result, there is a set of potential expanding locations that seem safe and could add more locations to S_{t+1} . This could be viewed as only sampling locations in the superlevel set, H_t , with some potentially safe points in U_t that could expand H_{t+1} . While this approach discovers safe regions, it favors higher

rewards over the pure exploration approach suitable for seabed characterization. In contrast, [19] aligns more with GP-LSE and considers sampling locations that maximally reduce variance using a lookahead operation. The lookahead operation involves forming $\sigma(x)_{t+1}$ conditioned on, \dot{x} , a location simulated for sampling. Because (2.15) does not require measurement values, it can be updated without a posterior mean.

Other approaches make paths intentionally exploring locations in the uncertain set while traveling to a selected next location [32, 21]. In [32], a graph with nodes containing all points in the uncertain set is made with a node's weight being assigned the location's ambiguity (2.20). The edge weight is the cost associated with traveling between nodes. An orienteering step calculates an affordable path by solving a traveling salesperson problem that maximizes cumulative ambiguity from visiting nodes. A separate approach selects a terminal location using (2.20) and travels through as many uncertain locations within a specified radius as it moves [21]. Both of these approaches utilize a process of explicitly sampling uncertain locations.

Outside of environmental monitoring, two multi-armed bandit solutions utilize directly exploring the decision boundary by sampling locations closest to the threshold [37, 22]. These approaches, which also perform level set estimation, directly focus on partitioning arms by discovering locations near the threshold. They travel minimally and often sample near previous locations, but fail to explore additional separated boundaries.

It stands to a reason that an algorithm focused on reducing the number of

uncertain locations, defined as locations in U_t (2.18), would be beneficial for LSE. Empirical results from [1, 19] and [21] indicate path sampling and variance reduction are also beneficial for environmental monitoring use LSE. To this end, we hypothesize that an algorithm capable of directly reducing the size of the uncertain set using a lookahead operation could perform exceptionally well in environmental monitoring. Unlike variance or ambiguity focused approaches, reducing the uncertain set size directly tackles LSE.

3 Lookahead Uncertain Set Reduction

The following chapter introduces the main contribution of this thesis: an adaptive sampling algorithm for seabed type identification using an autonomous underwater vehicle. It begins by connecting to previous environmental monitoring solutions before introducing uncertain set reduction and pessimistic sampling for a lookahead operation.

Lookahead uncertain set reduction (LUSR) is a distance penalized level set estimation algorithm that samples to directly reduce the cardinality of the uncertain set, defined as locations with upper and location confidence bounds straddling the threshold. defined by (2.18). Despite having the same goal of level set estimation, previous works sampled using surrogates, such as variance reduction [1, 19] and points near the decision boundary [37, 22], to prioritize correct level set assignment. In contrast, LUSR is designed to reduce the cardinality of the uncertain set as quickly as possible. Furthermore, it is restricted to sample from a fixed number of neighbors. This directly reduces the distance traveled while providing significant information gain.

LUSR sequentially updates its mean estimate through kernel ridge regression (KRR) (2.14) and its confidence interval through the kernelized Mahalanobis distance (2.15), which are equivalent to the update equations for Gaussian processes [28] and previous environmental monitoring solutions [1, 32, 21, 35, 18]. It is well

known that inverting matrix $K \in \mathbb{R}^{t \times t}$, which is used in both (2.14) and (2.15), has a computational complexity of $\mathcal{O}(t^3)$. As a result, an online update utilizing the Schur complement [38] is introduced [34] with a computational complexity of $\mathcal{O}((t-1)^2)$. Finally, while any regression algorithm with an uncertainty estimate can be used, there are significant benefits we leverage from kernel methods, particularly contextual bandits discussed in Section 2.3.

Briefly, kernel methods assume a mapping $\phi : \mathcal{X} \mapsto \mathcal{H}$ to place data $\mathcal{X} \subset \mathbb{R}^d$ in a Hilbert Space, \mathcal{H} . Assuming the associated Hilbert space exists, a positive definite kernel function is defined by the inner product of the mapping,

$$k(x, x') = \phi(x)^T \phi(x'). \quad (3.1)$$

where the mean estimate exists as an inner product with our high dimensional data, $\mu^*(x) = \phi(x)^T \omega^*$ [34]. Furthermore, it is shown in [34] that the mean estimate is bounded by

$$|\mu(x)_t - \phi(x)^T \omega^*| \leq \eta(1 + \|\omega^*\|) + \gamma^{(1/2)} \|\omega^*\| \sigma_t(x), \quad (3.2)$$

with high probability where γ is a regularization parameter from the confidence width update (2.15) and η is a tuning parameter controlling the weight of the confidence interval (2.19).

Consider a set of visited sample location x_1, \dots, x_t with corresponding similarity measurements s_1, \dots, s_t obtained from the exponentiated Jefferys divergence (1.3).

Given a kernel, $k(x, x^*) = \phi(x)^T \phi(x^*)$, define a matrix $K_t \in \mathbb{R}^{t \times t}$ to have entries

$$K_t = \{k(x_i, x_j)_{i,j \leq t}, x_i, x_j \in \mathcal{D}\}. \quad (3.3)$$

Denote $k_{x,t}$ to be

$$k_{x,t} = [k(x, x_1), \dots, k(x, x_t)]^T, \quad (3.4)$$

and form a collection of recorded samples,

$$y_t = [s_1, \dots, s_t]^T. \quad (3.5)$$

The KRR mean prediction and confidence widths satisfy (2.14) and (2.1) respectively.

It is known that inverting $A \in \mathbb{R}^{(t \times t)}$ is an $\mathcal{O}(t^3)$ operation using Gaussian elimination. Since K_t is symmetric and positive semi-definite, an online matrix inverse update utilizing the Schur complement is implemented to reduce the computational complexity, as shown in Algorithm 1 [38, 34]. This approach is $\mathcal{O}((t-1)^2)$ as it relies only on matrix multiplication.

LUSR utilizes a one-step lookahead to estimate the size of the future uncertain set after sampling location x_t . The confidence widths (2.15) do not depend on the measurement value and can be evaluated exactly prior to sampling a given point. The uncertain set (2.18) requires the posterior mean estimation (2.14), which depends on similarity measurement s_t . A lookahead estimate, therefore, can utilize the existing confidence interval (2.19) to obtain a false measurement for a

Algorithm 1 Online Matrix Inversion Update

Input: Matrices K_t and K_{t-1}^{-1} , Regularization parameter γ , kernel $k(\cdot, \cdot)$, Sampled location (x_1, \dots, x_t)

Output: K_t^{-1}

```

1: if  $K_t \in \mathbb{R}^{(1 \times 1)}$  then
2:    $K_t^{-1} \leftarrow \frac{1}{k(x_t, x_t) + \gamma}$ 
3: else
4:    $b \leftarrow (k(x_t, x_1), \dots, k(x_t, x_{t-1}))^T$ 
5:    $K_{22} \leftarrow (k(x_t, x_t) + \gamma - b^T K_{t-1}^{-1} b)^{-1}$ 
6:    $K_{11} = K_{t-1}^{-1} + K_{22} K_{t-1}^{-1} b b^T K_{t-1}^{-1}$ 
7:    $K_{12} \leftarrow -K_{22} K_{t-1}^{-1} b$ 
8:    $K_{21} \leftarrow -K_{22} b^T K_{t-1}^{-1}$ 
9:    $K_t^{-1} \leftarrow \begin{bmatrix} K_{11} & K_{12} \\ K_{12} & K_{22} \end{bmatrix}$ 
10: end if
11: return  $K_t^{-1}$ 

```

given location. In the Gaussian process view, the confidence interval acts as a bound on the true mean, allowing a fake measurement to be $\tilde{s} = \{x : |\mu_t(x) - \tau| \leq \eta\sigma_t(x)\}$ with high probability [34, 28]. In particular, we set

$$\tilde{s}_{t+1} = \begin{cases} \mu_t(x) - \eta\sigma_t(x), & \mu_t(x) > \tau \\ \mu_t(x) + \eta\sigma_t(x), & \mu_t(x) \leq \tau \end{cases} \quad (3.6)$$

then form estimate level set \tilde{L}_{t+1} , \tilde{H}_{t+1} , and \tilde{U}_{t+1} from an updated mean, $\tilde{\mu}_{t+1}$, and confidence width, σ_{t+1} . The false measurement (3.6) can be viewed as a pessimistic choice as the pseudo-sample removed the fewest points from the uncertain set. If $\mu_t(x) > \tau$, \tilde{s}_{t+1} is set to the lower bound, it pushes x and similar points closer to the level set boundary τ . Equation (3.6) was decided upon as it pushes more

locations into the lookahead uncertain set, \tilde{U}_{t+1} , by assuming a worse-case level set membership. Equation (3.6) was validated empirically, as seen in Figure 4.1. As a means of distance penalization, LUSR is restricted to the ρ nearest neighbors, denoted as $\mathcal{N}_\rho(x_t)$ where x_t is the AUV's current location

$$\mathcal{N}_\rho(x_t) = \arg \min_{U_{t,k} \in U_t, |U_{t,k}|=\rho} \sum_{x_i \in U_{t,k}} \|x_i - x_t\|. \quad (3.7)$$

LUSR samples the uncertain location that maximizes the estimated uncertain set reduction

$$x_{t+1} = \arg \max_{x \in \mathcal{N}_\rho(x_t)} |U_t| - |\tilde{U}_{t+1}|. \quad (3.8)$$

Distance penalization is an important decision that controls computational complexity, cost, and performance, as shown in Chapter 4. We show in Chapter 4 that the ρ nearest neighbors approach effectively balances exploring new regions and exploiting known boundaries. Pseudocode for LUSR can be found in Algorithm 2.

Extending (3.2), the error bound for a lookahead estimate can be derived by appending a pessimistic sample to $k_{x,t}$ and y_t . The results show that a lookahead estimate incurs error related to the similarity between the lookahead location, \tilde{x}_{t+1} , and any other point in the domain

$$|\tilde{\mu}_t(x) - \phi(x)^T \omega^*| \leq \bar{\eta} \sigma_t(x) + \frac{2\bar{\eta}}{\gamma} \sigma_t(x) |k(x, \tilde{x}_{t+1}) - k_{x,t+1}^T (K_{t+1} + \gamma I)^{-1} k_{\tilde{x}_{t+1}, t+1}| \quad (3.9)$$

where $\bar{\eta} = \eta(1 + \|\omega^*\|) + \gamma^{1/2} \|\omega^*\|$ with full derivation in Appendix A. This result shows a reliance on the previous confidence width, as shown in [34], with an

Algorithm 2 Lookahead Uncertain Set Reduction (LUSR)

Input: domain \mathcal{D} , initial location x_0 , kernel function k , threshold τ , tuning parameters γ, η , similarity parameter ℓ , number of neighbors ρ , number of samples T

Output: predicted sets L_T, H_T

```

1:  $L_0 \leftarrow \emptyset, H_0 \leftarrow \emptyset, U_0 \leftarrow \mathcal{D}$ 
2: for  $t = 0, \dots, T$  do
3:    $\mathcal{N}_\rho(x_t) \leftarrow \rho$  nearest neighbors of  $x_t$  in  $U_t$ 
4:    $x_{t+1} \leftarrow \arg \max_{x \in \mathcal{N}_\rho(x_t)} |U_t| - |\tilde{U}_{t+1}|$ 
5:   collect snapshots, form similarity  $s_{t+1} = \exp \left( -J(\hat{\Sigma}_0 \| \hat{\Sigma}_t) / \ell^2 \right)$ 
6:    $y_{t+1} \leftarrow [s_0, s_1, \dots, s_{t+1}]$ 
7:   update  $k_{x,t+1}$  by (3.4) and  $K_{t+1}^{-1}$  by Algorithm 1
8:    $\mu_{t+1}(x) \leftarrow k_{x,t+1}^T K_{t+1}^{-1} y_{t+1}, \forall x \in \mathcal{D}$ 
9:    $\sigma_{t+1}(x) \leftarrow \gamma^{-1/2} \sqrt{k(x, x) - k_{x,t+1}^T K_{t+1}^{-1} k_{x,t+1}}, \forall x \in \mathcal{D}$ 
10:   $H_{t+1} \leftarrow \{x \in \mathcal{D} : \mu_{t+1}(x) - \eta \sigma_{t+1}(x) \geq \tau\}$ 
11:   $L_{t+1} \leftarrow \{x \in \mathcal{D} : \mu_{t+1}(x) + \eta \sigma_{t+1}(x) \leq \tau\}$ 
12:   $U_{t+1} \leftarrow \{x \in \mathcal{D} : |\mu_{t+1}(x) - \tau| \leq \eta \sigma_{t+1}(x)\}$ 
13: end for

```

additional factor loosely representing (2.15) between x and \tilde{x}_{t+1} . In theory, η is a time-varying parameter. LUSR, however, sets it as a constant parameter to be optimized [39] as it proved to be difficult to set. While this comes with practical benefits, it lacks theoretical analysis [34].

4 Results

This section empirically evaluates the performance of LUSR on synthetic data and realistic seabeds from the HFEVA database. LUSR is compared to a few sample policies in the spirit of previous solutions. Specifically, LUSR is compared to random sampling, margin [22], max variance [1], max ambiguity [1], and a lookahead variance reduction [19]. Lookahead variance reduction (LA VAR) samples the location estimated to reduce the most variance via a lookahead operation and was made in the spirit of Truncated Variance Reduction (TruVar) [19]. While TruVar surveys the whole domain, LA VAR considers locations only in the uncertain set. All policies, which are defined in Table 4.1, use the radial basis function kernel for their mean (2.14) and confidence width (2.15) estimates

$$k(x, x') = \exp \left(-\frac{1}{2\delta^2} \|x - x'\|^2 \right), \quad (4.1)$$

and are restricted to the ρ nearest neighbors. Although not shown, the state-of-the-art straddle heuristic [40] performed similarly to margin while requiring an additional tuning parameter.

Class accuracy is used to quantify the performance of each algorithm

$$\mathcal{L}_t = \frac{1}{|\mathcal{D}|} |(H^* \cap H_t) \cup (L^* \cap L_t)|, \quad (4.2)$$

Table 4.1: Sampling Policies

Tested Sampling Policies	
<i>Identifier</i>	<i>Policy</i>
Margin	$x_{mar} = \arg \min_{x \in \mathcal{N}_\rho(x_t)} \mu_t(x) - \tau $
Variance Reduction	$x_{VAR} = \arg \max_{x \in \mathcal{N}_\rho(x_t)} \sigma_t(x)$
Ambiguity	$x_{ambig} = \arg \max_{x \in \mathcal{N}_\rho(x_t)} a_t(x) \text{ (2.20)}$
Lookahead Variance Reduction	$x_{LAVAR} = \arg \min_{x \in \mathcal{N}_\rho(x_t)} \sum_{x_i \in \mathcal{X}} \sigma_{t+1}(x_i)$

with H^* and L^* being the true superlevel and sublevel set respectively.

4.1 Lookahead Sampling

When introducing a lookahead sampling policy, we consider mean, optimistic, and pessimistic estimate for the false measurement, \tilde{s}_{t+1} . A mean estimate simply sets $\tilde{s}_{t+1} = \mu_t(x)$ while an optimistic estimate sets \tilde{s}_{t+1} maximally away from the threshold using its confidence bound

$$\tilde{s}_{t+1} = \begin{cases} \mu_t(x) + \eta\sigma_t(x), & \mu_t(x) > \tau \\ \mu_t(x) - \eta\sigma_t(x), & \mu_t(x) \leq \tau \end{cases}. \quad (4.3)$$

A pessimistic sampling policy, defined earlier in Chapter 3, sets the false measurement towards the threshold, encouraging more uncertain locations. Six trials over 12 synthetic boundaries, discussed in Section 4.2, were used as environments to compare the three false measurements. The pessimistic estimate achieved a higher accuracy with $\rho = [20, 50, 80]$ and $T = 100$ samples, as shown in Figure 4.1.

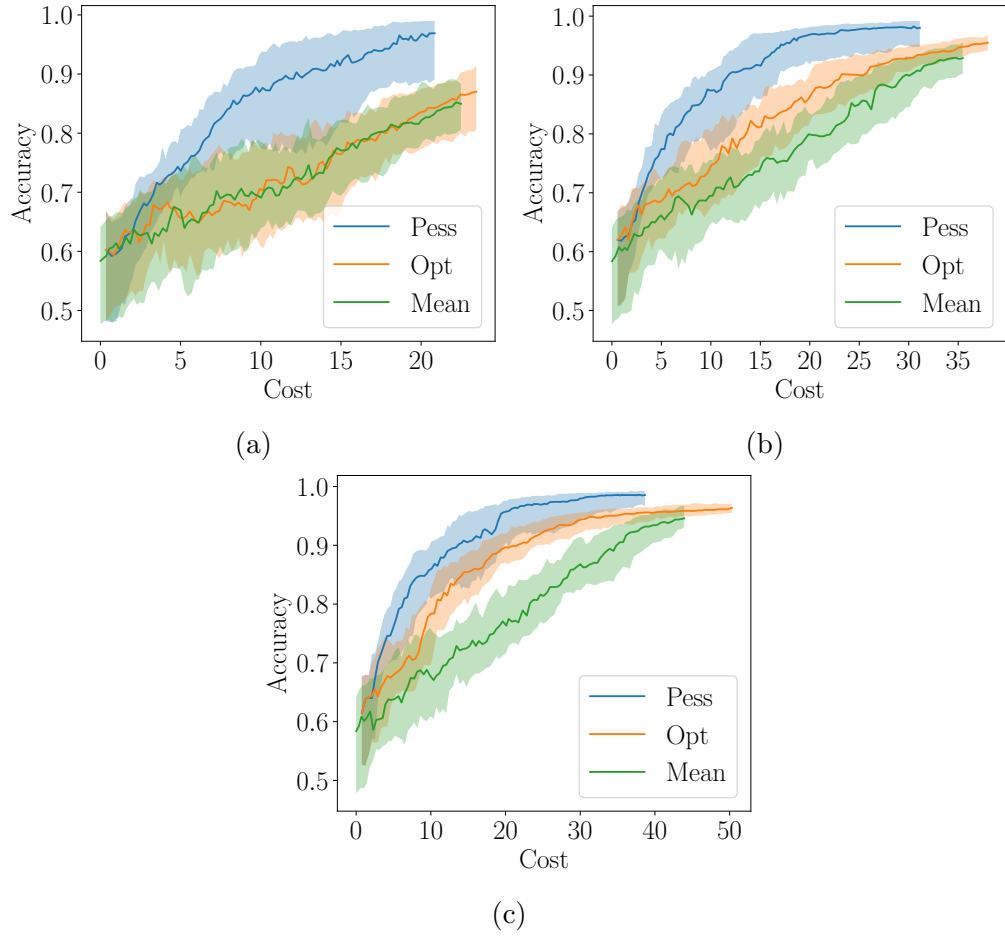


Figure 4.1: The solid lines reflect the median value over 6 trials for the first 12 randomly generalized boundaries discussed in Section 4.2. The interquartile range is shaded. In the low noise setting, 20 neighbors (a), 50 neighbors (b), and 80 neighbors (c) all show pessimistic sampling outperforming optimistic and mean sampling by reaching a higher accuracy with lower cost.



(a) Original Fields

(b) Additional Fields

Figure 4.2: Twenty randomly simulated fields used to evaluate each policy (a) and twenty additional fields, each with two boundaries (b). The reference location is the top left of each boundary with yellow indicating the superlevel set and purple indicating the sublevel set.

4.2 Synthetic Results

Twenty simulated boundaries of dimension 50×50 were randomly generated by mapping the coordinates to a fourth-order polynomial, generating a random weight vector in the high dimensional space, then labeling each location using a linear classifier defined by the weight vector. The results, shown in Figure 4.2, are composed of a superlevel set, H^* , in yellow and sublevel set, L^* , in purple. Each field underwent 32 trials with measurements drawn from a truncated normal distribution with means zero and one for within-class and across-class similarities. The low and high noise settings set $\sigma_{noise} = 0.15$ and $\sigma_{noise} = 0.45$ respectively.

The level set threshold was set at the median, $\tau = 0.5$. For all policies, $\gamma = 0.01$ and $\eta = 0.5$. The number of neighbors $\rho \in [1, 2, 5, 10, 15, 20, 40, 60, 80, 100]$ was

designed to minimize computational complexity and distance traveled. There is no cost to sample as ambient acoustic noise is captured continuously and freely. After each iteration, each policy was evaluated on its classification accuracy treating the level set membership as a binary label.

Figure 4.3 shows the median accuracy against the distance traveled in the original fields for $\rho = 100$ neighbors in low and high noise settings along with the interquartile range. It demonstrates that LUSR reaches a higher accuracy with a lower distance compared to the other competing algorithms. Of the two variance reducing algorithms, maximal variance reduction (VAR) is shown as it preformed similarly to the lookahead variance reduction algorithm (LA VAR) made in the spirit of TruVar [19] while being computationally cheaper. It was removed from Figure 4.3 for clarity. VAR can detect multiple boundaries quickly, but its strong preference for exploration increases its traveling distance. By focusing on reducing the uncertain set, LUSR indirectly sampled locations that are both near the boundary and have high uncertainty, obtaining an appropriate degree of exploration.

Margin sampling surpassed our expectations, which we assumed was due to the simplicity of the original fields. As a result, additional fields with two boundaries each were generated to mimic the realistic seabed data discussed in Section 4.3. The results, shown in Figure 4.4, show LUSR performing strongly in the low noise setting and similarly to VAR in the high noise setting. Random sampling, which consistently performs poorly, was replaced with ambiguity sampling as it uses information from both the mean and uncertainty estimates like LUSR.

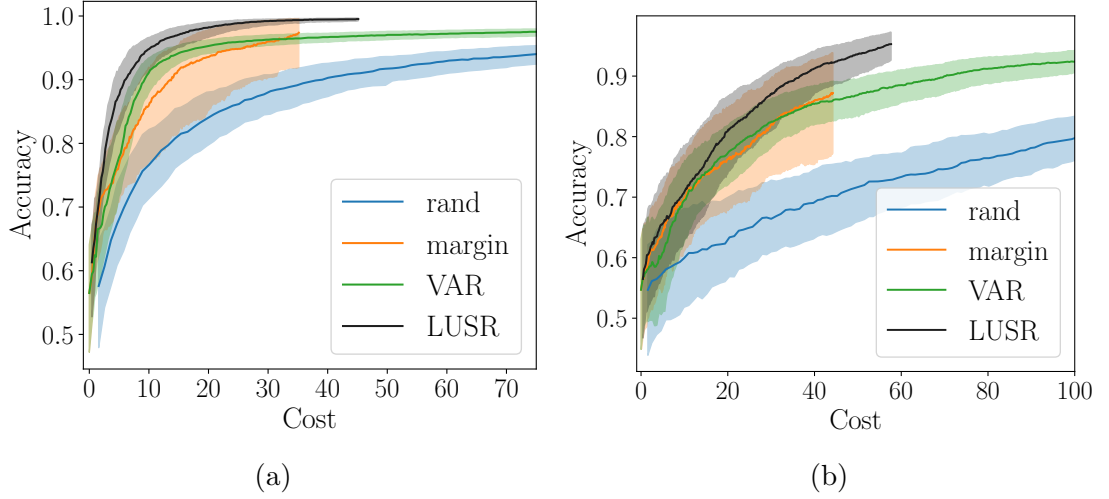


Figure 4.3: Accuracy compared to cost for the original synthetic fields. The solid lines reflect the median value over 32 trials for each of the boundaries. The interquartile range is shaded. In a low noise setting with 100 neighbors (a), LUSR achieves a high accuracy with very little cost compared to its competitors. In the high noise setting (b), LUSR moves a bit close to its competitors while still attaining a higher accuracy and lower cost.

Figure 4.5 shows the accuracy and distance traveled against the number of neighbors in the low noise setting after 100 samples on the original fields. LUSR’s median accuracy begins to outperform VAR’s with $\rho = 15$ neighbors. The plot shows margin achieving a worse accuracy across all ρ values while also traveling the least, a feature of prioritizing exploitation over exploration. VAR, in turn, performs similarly to LUSR while traveling significantly more as ρ increases due to over-exploration. LUSR reaches an middle ground between margin and VAR, which is hypothesized to be a product of prioritizing level set assignment. It samples both the boundary and locations of high variance.

Table 4.2 shows computation times for all algorithms using the synthetic boundaries. All algorithms ramp up within the first 10 samples before reaching a steady

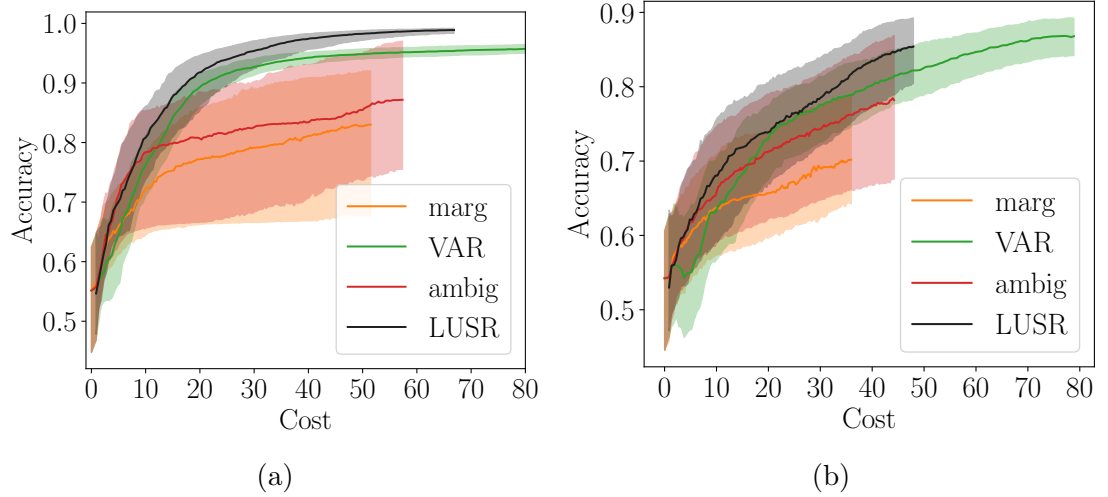


Figure 4.4: Accuracy compared to cost for the additional synthetic fields over 150 samples. In a low noise setting with 100 neighbors (a), LUSR still achieves a higher accuracy than its competitors. In the high noise setting (b), LUSR begins to struggle and performs similarly to VAR.

state computation time. Despite similar performances, LUSR requires more than 10^6 times longer than VAR. Both of these, however, are small enough to be usable. With limited computational resources, a small number of neighbors to consider, or if investigating a small environment, VAR may be preferred. LUSR seems to perform better with more computational resources, a higher number of neighbors, or when a larger environment is available. Similarly, the tuning parameters, η and γ were found to be difficult to set appropriately.

4.3 Seabed Results

The algorithms are compared on model data from the state-of-the-art Multi-Dimensional Ambient Noise Model [41] over a region in the the north Pacific Ocean,

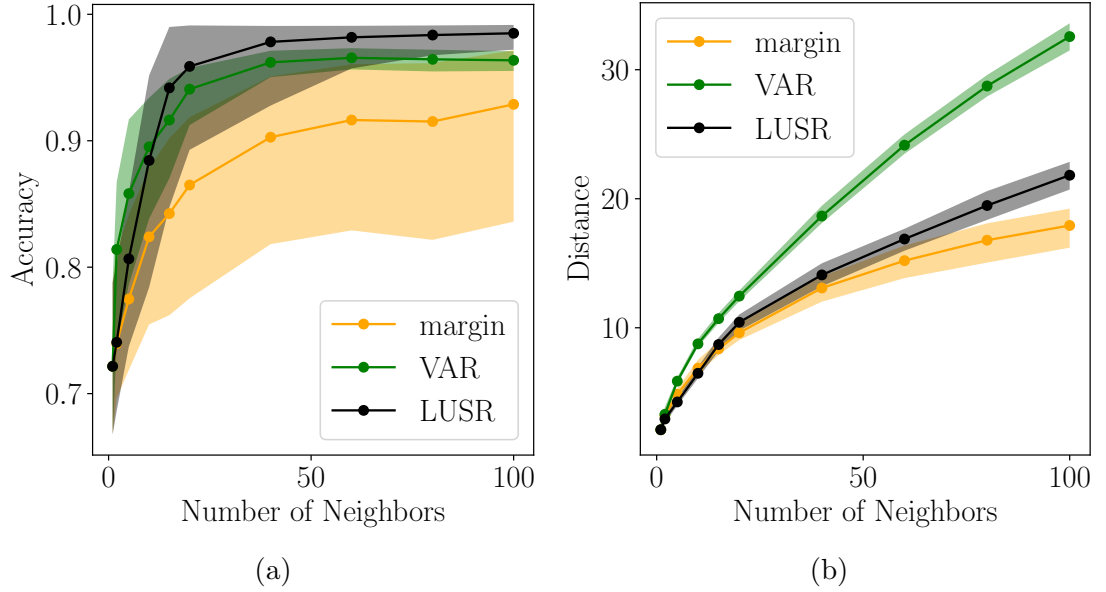


Figure 4.5: At the end of 100 iterations, LUSR’s median accuracy (a) outperforms the other algorithms with 15 neighbors. Similarly, it travels less than VAR and slightly more than margin. Margin’s favorability for the decision boundary helps keep a low traveling cost.

specifically the red box shown in Figure 2.1. The region contains HFEVA sediment types 2, 8, 16, 17, and 22, which are rock, gravelly muddy sand, gravelly mud or sandy silt, medium silt or sand-silt-clay, and silty clay respectively. Types 16-22 have very similar covariances and represent the predominant background sediment. The goal is to distinguish types 2 and 8, gravelly sediment types, from 16-22, silty sediment types. For each sediment type, the corresponding signal covariance from the multi-dimensional ambient noise model [41] is used to generate complex normal snapshots according to the covariances defined by (2.1) with a signal-to-noise ratio of 10. Each sampled location provides $L = 300$ snapshots from $M = 32$ receivers. The tuning parameter in the similarity measurements (1.3) is set to $\ell = 10$ with the

Table 4.2: Mean and median time (s) to compute the 100th sample with 40 neighbors

Computation Time (seconds) at Steady State		
Policy	Mean	Median
Margin	4.411×10^{-4}	3.757×10^{-4}
Variance Reduction	$0.0001.251 \times 10^{-4}$	1.191×10^{-4}
Ambiguity	1.428×10^{-4}	1.159×10^{-4}
Lookahead Variance Reduction	27.212	23.084
LUSR	16.292	13.455

level set threshold chosen to be $\tau = 0.3$ from domain knowledge. It is important to note that all policies are invariant to these parameters.

Figure 4.6 shows the performance metrics over the north Pacific Ocean region. This region contains multiple boundaries. As a result, margin sampling struggles to explore the new boundaries. VAR and LUSR operate similarly, which is justified as both attempt to reduce uncertainty. However, VAR over explores while LUSR remains committed to reducing the size of the uncertain set, evidenced by the 6000 km distance cost shown in Figure 4.6.

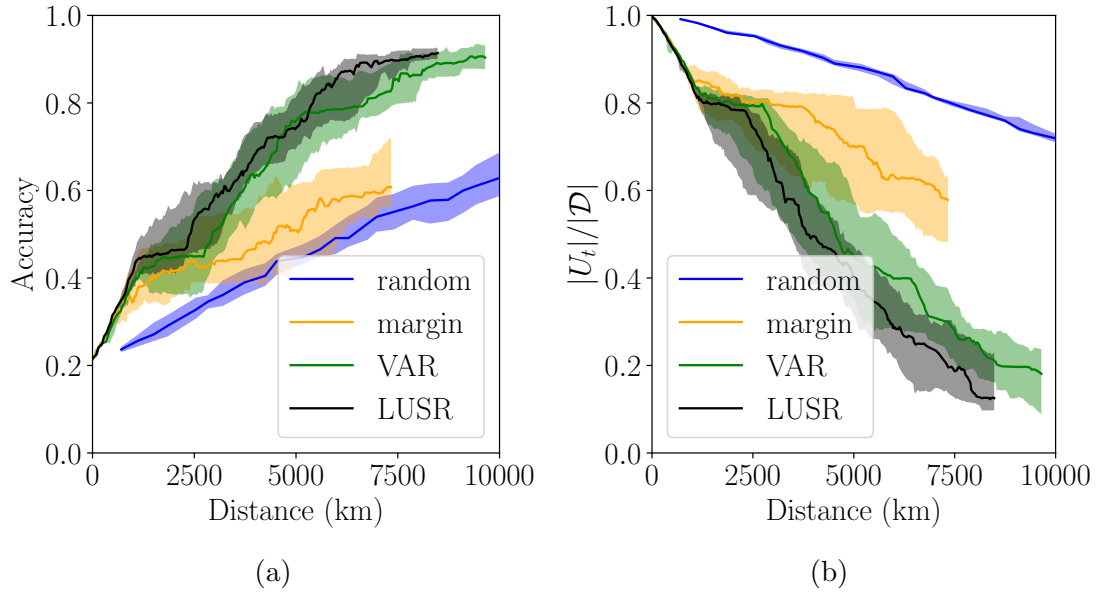


Figure 4.6: LUSR and VAR perform similarly (a), however, LUSR gets more of an edge over VAR between ,5000 - 7,500 km by finding an unexplored boundary earlier. This is shown in LUSR's uncertain set size (b).

5 Extensions

The following chapter discusses secondary LUSR iterations in addition to simulated path sampling. Since AUVs can sample freely, previous solutions implemented continuous sampling and path planning stages [21, 32, 42]. In regard to the lookahead estimate error derived in (3.9), it makes sense to ask how additional lookahead steps would alter performance in the view of a second LUSR iteration, known as LUSR-2. After an initial LUSR iteration, a second LUSR iteration performs LUSR again using \tilde{U}_{t+1} to predict the reduction in \tilde{U}_{t+2} , which forms similarities using the current mean estimate $\mu_t(x)$ (2.14) with the future confidence interval σ_{t+2} (2.15). The current mean estimate is used for the fake measurements as each lookahead mean estimate adds error, proven in (3.9). Similarly, LUSR Path – an algorithm that computes LUSR along a path – can be viewed as a n -step LUSR operation and is detailed in Section 5.2.

Additional kernel data would be beneficial since mean and confidence widths are reliant on kernel measurements. Bathymetry data covering a region of the northern Atlantic Ocean was combined with the radial basis function kernel in an attempt to encourage information gain.

5.1 Second-Order LUSR

A second-order LUSR could use a second lookahead operation to verify the first LUSR search and authenticate that the AUV is moving towards an appropriate area for exploration. LUSR-2, a second-order LUSR operation, completes an initial LUSR to identify sampling location, x_{t+1} . After forming the lookahead uncertain set, \tilde{U}_{t+1} , a secondary LUSR operation is computed to identify \tilde{x}_{t+2} , a hypothesized secondary future location to sample assuming the AUV travels to x_{t+1} . To compute a second-order LUSR, two similarity measurements are appended to (3.5) forming

$$\tilde{y}_{t+1} = [s_1, \dots, s_t, \tilde{s}_{t+1}, \tilde{s}_{t+2}],$$

where \tilde{s}_{t+1} and \tilde{s}_{t+2} come from (3.6) using the current mean $\mu_t(\tilde{x})$ while updating the confidence widths, $\sigma_t(\tilde{x}_{t+1})$ and $\sigma_{t+1}(\tilde{x}_{t+2})$ respectively. This process is repeated for all combinations of $x \in \mathcal{N}_\rho(x_t)$ and $\tilde{x}_{t+1} \in \tilde{\mathcal{N}}_\rho(x)$, where $\tilde{\mathcal{N}}_\rho(x)$ contains the nearest ρ uncertain locations after the secondary mean and confidence width update. Like LUSR, the sampling policy for LUSR-2 selects the next location upon the maximum cardinality reduction

$$x_{t+1} = \arg \max_{x \in \mathcal{N}_\rho(x_t)} |U_t| - \max_{\tilde{x} \in \tilde{\mathcal{N}}_\rho(x)} |\tilde{U}_{t+2}|, \quad (5.1)$$

where \tilde{U}_{t+2} is the uncertain set after the two-step lookahead that samples $x \in \mathcal{N}_\rho(x_t)$ then $\tilde{x} \in \tilde{\mathcal{N}}_\rho(x)$. It is important to note that both iterations of LUSR use the current mean estimate, μ_t , to form lookahead similarity measurements (1.3),

which was proven theoretically (3.2) and empirically, as shown in Figure 5.1.

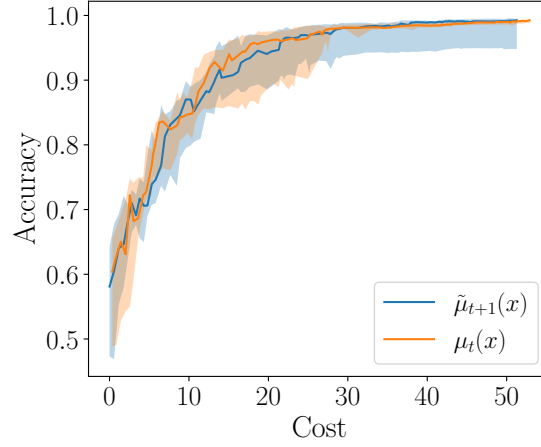


Figure 5.1: LUSR-2 is shown performing marginally better with the current mean estimate (orange) than one using a lookahead measurement (blue). The lookahead error bound (3.2) is worse than the current mean estimate, justifying the worse performance using the updated mean.

Excluding margin sampling, which performed significantly worse as ρ increased, LUSR-2 does achieve a higher accuracy than other algorithms while maintaining a worse cardinality, as shown in Figure 5.2.

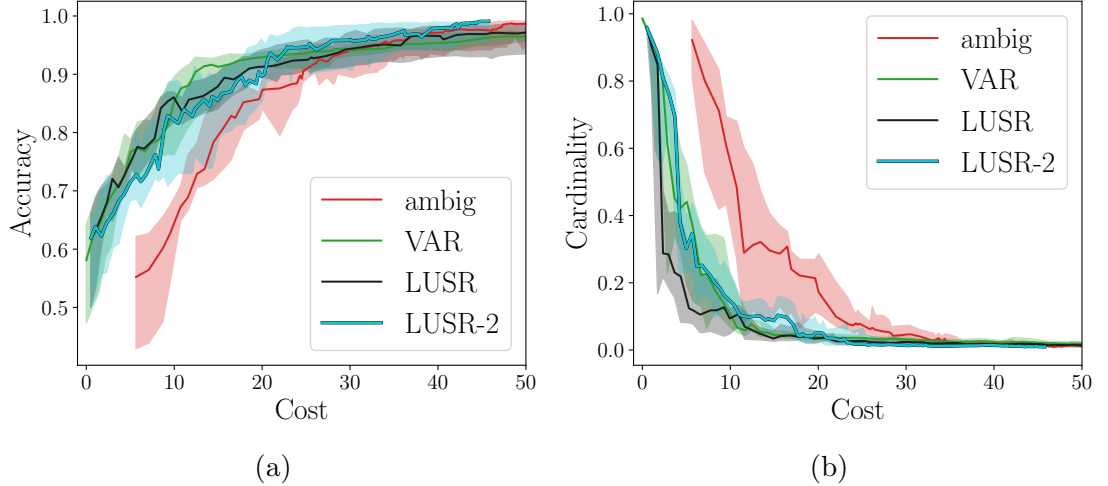


Figure 5.2: With 40 nearest neighbors, LUSR-2 reaches a near perfect accuracy with less cost than other algorithms (a). The cardinality, however, performed worse (b). Further testing is required if LUSR-2 maintains a higher accuracy with a lower cardinality or obtains a lower cardinality and the higher accuracy in (a) is due to simple boundaries.

5.2 Continuous Sampling

Since LUSR-2 achieved a higher accuracy than LUSR, it stands to reason that LUSR- n , an n -step LUSR algorithm may perform more optimally. Previous sampling methods benefit from continuous sampling as the AUV travels from one location to the next [21, 43, 32, 42]. Unlike original LUSR, which samples only at specified locations, continuous sampling locations collect data as they travel between locations. However, performing a lookahead process over a continuous path, where the estimated future predictions compound on each other, continues to add variance at each lookahead iteration. To counter this, the current mean estimate is used for all lookahead processes like LUSR-2. The path predicted to reduce the uncertain

set the most is executed using continuous sampling, shown in Figure 5.3.

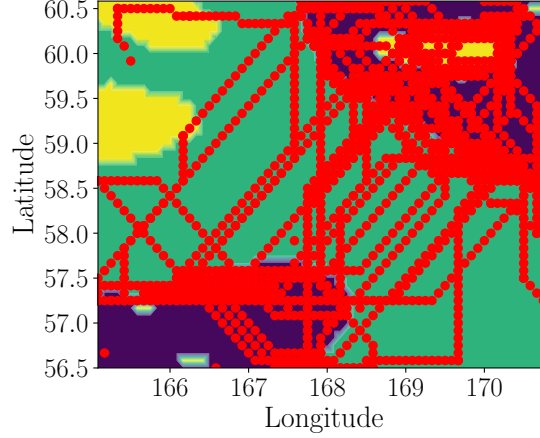


Figure 5.3: Executed paths computed using LUSR Path over a smaller region in the northern Pacific Ocean using $\rho = 20$.

In a given iteration, the original LUSR (3.8) estimates the reduction of the uncertain set given we synthetically sample a location $\tilde{x}_{t+1} \in \mathcal{N}_\rho(x_t)$. For each $\tilde{x}_{t+1} \in \mathcal{N}_\rho(x_t)$, the shortest path from x_t to \tilde{x}_{t+1} , \mathcal{P} is calculated. For each location $x \in \mathcal{P}$, a pessimistic mean and confidence width update is calculated using the current mean estimate μ_t .

Previously, all algorithms operated in rounds, where each round required one sample to be collected. LUSR Path, however, executes an entire path with varying length. In the beginning, paths are relatively short as the nearest neighbors are closer. In the later iterations, uncertain locations spread out, resulting in longer paths. Despite the significantly more sampling LUSR Path collects, it performs only slightly better than the other competing algorithms, as seen in Figure 5.4.

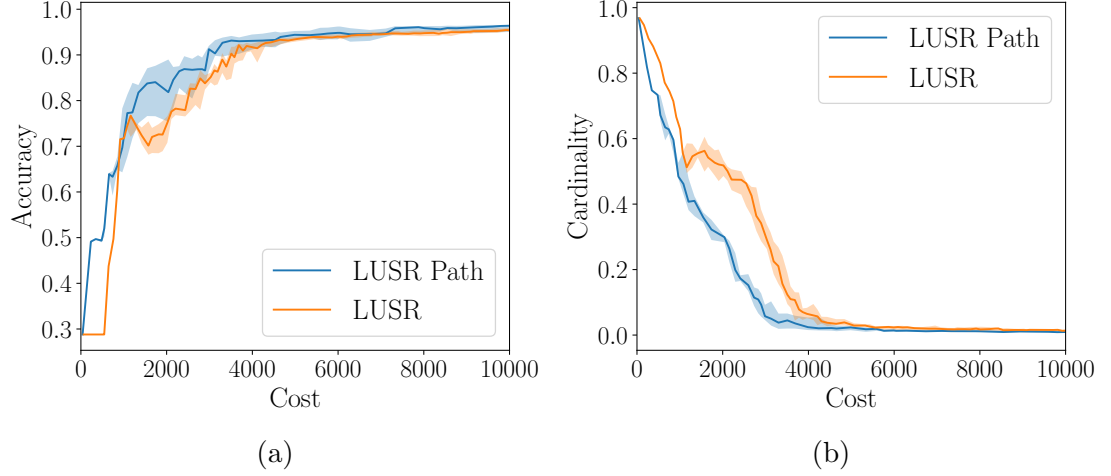


Figure 5.4: Due to the wide sampling range, LUSR Path avoids a dip that LUSR falls for (b) and (c). Despite this, they generally track each other while LUSR Path takes significantly more computation time – often taking days or weeks longer than LUSR.

LUSR Path successfully avoids a pitfall that LUSR does not avoid, shown in ???. The benefits on limited beyond that. In terms of computational cost, LUSR Path would take days to weeks longer than LUSR, stunting our ability to consider larger number of neighbors. It may be beneficial to restrict paths to finite number of samples taken at fixed intervals along the path, similar to some local approaches [20, 3, 16].

5.3 Bathymetry Data

When additional data is available, it may be beneficial to incorporate it into a Gaussian process model as an additional feature [28]. In regard to seabed identification, bathymetry – the depth of the ocean – data is readily available to be used as an additional feature. As a result, a combination kernel, $k_{combo}(\cdot, \cdot)$ is formed

using spatial and bathymetry similarities,

$$k_{combo}(x_1, x_2) = k_{RBF}(x_1, x_2) + c k_{bathymetry}(x_1, x_2), \quad c \in [0, 1], \quad (5.2)$$

where k_{RBF} is the radial basis function kernel described in Chapter 4. Adding kernels can be viewed as a boolean OR operation where $k_{combo}(x_1, x_2)$ will be high if either $k_{RBF}(x_1, x_2)$ or $k_{bathymetry}(x_1, x_2)$ is high. As a result, locations sharing similar bathymetry that are not physically close together will be viewed as similar under k_{combo} .

To do this, a region in the northern Atlantic Ocean with known bathymetry data is used, shown in Figure 5.5.

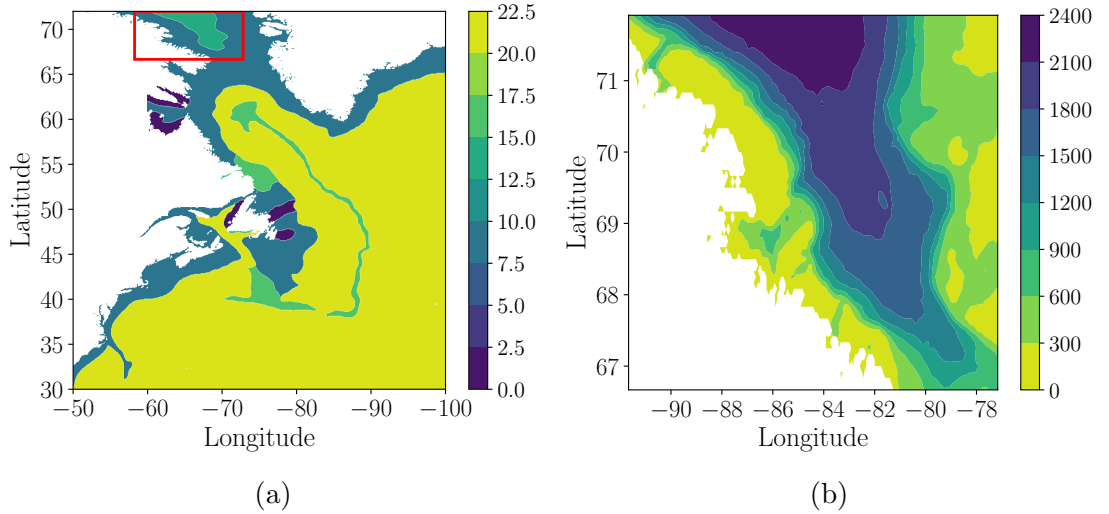


Figure 5.5: New region for LSE (a) with its corresponding bathymetry data (b) where white designates land. The seabed types shown in (a) is shown to be correlated to the depth of the ocean in (b).

The results, shown in Figure 5.6, reveal that the bathymetry data quickly produces accurate field estimates. However, it is difficult to find the appropriate

weight for the combined kernels. Using only bathymetry data, it is difficult to surpass 93% accuracy.

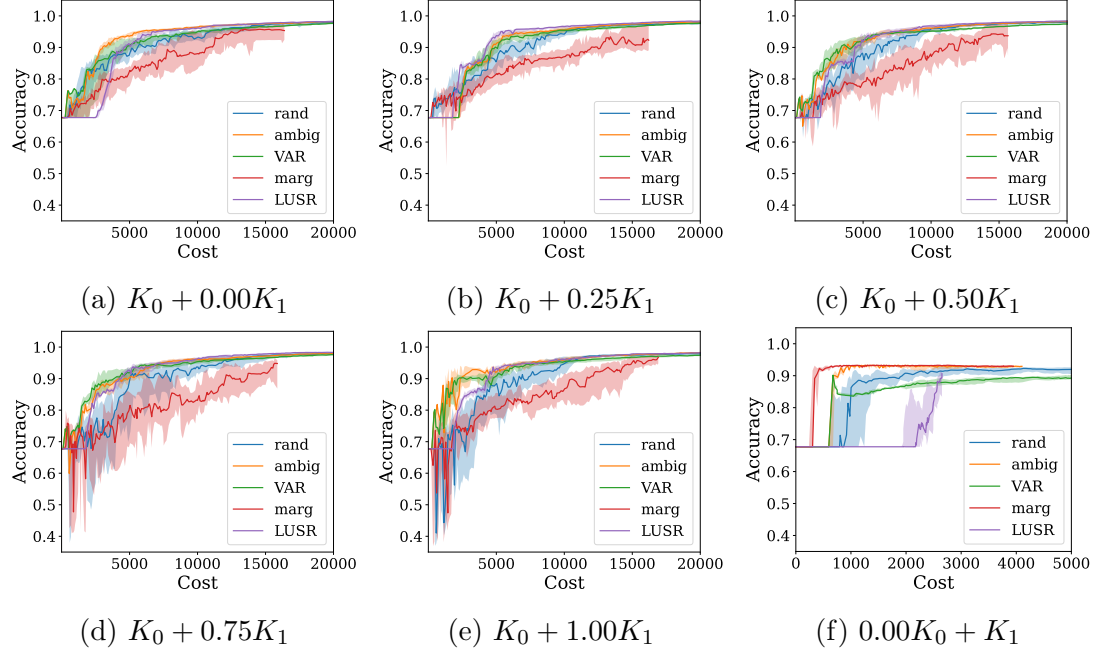


Figure 5.6: The results generally track when adding in the new kernel. The new kernel bumps accuracy to .9 quickly (f), and the adaptive sampling policies push past that at the same rate.

It is important to note that a kernel using only bathymetry data quickly terminates sampling as all locations have been assigned to H_t or L_t . Appropriately weighting uncertainty through tuning parameter η has proven to be difficult.

Table 5.1: Mean and median time (s) to compute the 100th sample with 40 neighbors

Computation Time (seconds) at Steady State		
Policy	Mean	Median
Margin	4.411×10^{-4}	3.757×10^{-4}
Variance Reduction	$0.0001.251 \times 10^{-4}$	1.191×10^{-4}
Ambiguity	1.428×10^{-4}	1.159×10^{-4}
Lookahead Variance Reduction	27.212	23.084
LUSR	16.292	13.455
LUSR-2	487.351	456.891
LUSR Path*	Terminated	Terminated

* Two attempts were terminated after stalling after 4 weeks

6 Conclusion

Approaching seabed characterization through a binary hypothesis test using ambient acoustic noise captured by an autonomous underwater vehicle (AUV) motivates an online adaptive sampling algorithm. We introduce an algorithm capable of using the test statistic from the generalized likelihood ratio test to identify locations that exceed a user-supplied threshold, transforming the problem of seabed characterization into level set estimation. During this process, the algorithm maintains three sets for locations exceeding the threshold, falling short of the threshold, and with uncertain placement with respect the threshold. Our algorithm, named LUSR, directly seeks to reduce the uncertain set using a lookahead operation. Furthermore, we implement distance penalization by restricting sampling to the nearest neighbors. While our algorithm excels at reducing the number of uncertain locations, it is sensitive to the coefficient scaling confidence widths, and it is unclear on how to set these appropriately. While we provide a bound on this coefficient, it can be difficult to tune in practice.

Empirical results on synthetic boundaries and realistic seabed data proved our algorithm outperforms or tracks competing algorithms. In particular, LUSR excels in low noise environments. Three extensions of LUSR are introduced: two-step LUSR, continuous sampling LUSR, and a combination kernel. LUSR-2, an algorithm with two lookahead operations, promoted high accuracy but fails to reduce

the cardinality of the uncertain set. More clarity is needed around whether LUSR-2 performs better across various boundaries or only in the single boundary synthetic fields it was simulated on. LUSR Path, a natural n^{th} order extension of LUSR-2, seems to avoid pitfalls LUSR fell into. However, performed slightly better with 20 nearest neighbors while consuming significantly more computational resources. Both attempts to simulate LUSR Path with the same setup as LUSR were terminated after stalling for weeks. The combination kernel appears to be an appealing addition which would improve information gain. Bathymetry data for a region in the Atlantic Ocean is introduced and quickly obtains a high accuracy. However, it is difficult to improve performance after the initial increase.

Future work could consider the estimated reduction in the uncertain set by using a neural network. The effects of realistic AUV movement, especially underwater, is another area to consider as we assume the AUV is capable of moving in straight lines. Additionally, determining how to set the correct bounds for the coefficient could provide significant advantages to set assignment and sampling protocol.

7 Bibliography

- [1] Alkis Gotovos et al. “Active learning for level set estimation”. In: *IJCAI*. 2013, pp. 1344–1350.
- [2] Phillip Kearns, John Lipor, and Bruno Jedynak. “Optimal Adaptive Sampling for Boundary Estimation with Mobile Sensors”. In: *2019 53rd Asilomar Conference on Signals, Systems, and Computers*. IEEE. 2019, pp. 1621–1625.
- [3] John Lipor et al. “Distance-Penalized Active Learning Using Quantile Search”. In: *IEEE Transactions on Signal Processing* 65.20 (2017), pp. 5453–5465.
- [4] Sean Stalley and John Lipor. “A Graph-Based Approach to Boundary Tracking with Mobile Sensors”. In: *IEEE Robotics and Automation Letters* (2021). accepted for publication.
- [5] John Gebbie and Martin Siderius. “Optimal environmental estimation with ocean ambient noise”. In: *The Journal of the Acoustical Society of America* 149.2 (2021), pp. 825–834.
- [6] Martin Siderius and John Gebbie. “Environmental information content of ocean ambient noise”. In: *The Journal of the Acoustical Society of America* 146.3 (2019), pp. 1824–1833.

- [7] MK Prior, CH Harrison, and SG Healy. “Assessment of the impact of uncertainty in seabed geoacoustic parameters on predicted sonar performance”. In: *Impact of Littoral Environmental Variability of Acoustic Predictions and Sonar Performance* (2002), pp. 531–538.
- [8] DEL BALZO D. R. “Critical angle and seabed scattering issues for active-sonar performance predictions in shallow water”. In: *High Frequency Acoustics in Shallow Water, Conference Proceedings, 1997* (1997).
- [9] Peter Chu, M. Cornelius, and M. Wegstaff. “Effect of suspended sediment on acoustic detection using the Navy’s CASS-GRAB model”. In: vol. 2005. Feb. 2005, pp. 1–7. ISBN: 0-933957-34-3. DOI: 10.1109/OCEANS.2005.1639866.
- [10] Jacob Piccolo, George Haramuniz, and Zoi-Heleni Michalopoulou. “Geoacoustic inversion with generalized additive models”. In: *The Journal of the Acoustical Society of America* 145.6 (June 2019), EL463–EL468. DOI: 10.1121/1.5110244.
- [11] Christina Frederick, Soledad Villar, and Zoi-Heleni Michalopoulou. “Seabed classification using physics-based modeling and machine learning”. In: *The Journal of the Acoustical Society of America* 148.2 (2020), pp. 859–872.
- [12] David F. Van Komen et al. “Seabed and range estimation of impulsive time series using a convolutional neural network”. In: *The Journal of the Acoustical Society of America* 147.5 (May 2020), EL403–EL408. ISSN: 0001-4966. DOI: 10.1121/10.0001216.

- [13] John Lipor, John Gebbie, and Martin Siderius. “On the limits of distinguishing seabed types via ambient acoustic sound”. In: *The Journal of the Acoustical Society of America* 154.5 (Nov. 2023), pp. 2892–2903. ISSN: 0001-4966. DOI: 10.1121/10.0022331.
- [14] Burr Settles. “Active Learning Literature Survey”. In: (July 2010).
- [15] Stephanie Petillo et al. “Autonomous & adaptive oceanographic front tracking on board autonomous underwater vehicles”. In: *OCEANS 2015-Genova*. IEEE. 2015, pp. 1–10.
- [16] John Lipor and Gautam Dasarathy. “Quantile Search with Time-Varying Search Parameter”. In: *2018 52nd Asilomar Conference on Signals, Systems, and Computers*. IEEE. 2018, pp. 1016–1018.
- [17] Dingyu Wang, Gautam Dasarathy, and John Lipor. “DISTANCE-PENALIZED ACTIVE LEARNING VIA MARKOV DECISION PROCESSES”. In: *Proc. IEEE Data Science Workshop*. 2019.
- [18] Gregory Hitz et al. “Fully autonomous focused exploration for robotic environmental monitoring”. In: *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE. 2014, pp. 2658–2664.
- [19] Ilija Bogunovic et al. “Truncated variance reduction: A unified approach to bayesian optimization and level-set estimation”. In: *Advances in neural information processing systems*. 2016, pp. 1507–1515.

- [20] John Lipor and Laura Balzano. “Quantile Search: A Distance-Penalized Active Learning Algorithm for Spatial Sampling”. In: *Proc. Allerton Conf. on Communication, Control, and Computing*. 2015.
- [21] Lorenzo Bottarelli et al. “Path efficient level set estimation for mobile sensors”. In: *Proceedings of the Symposium on Applied Computing*. 2017, pp. 262–267.
- [22] Daniel LeJeune, Gautam Dasarathy, and Richard Baraniuk. “Thresholding graph bandits with graph”. In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2020, pp. 2476–2485.
- [23] Parth Thaker et al. “Maximizing and Satisficing in Multi-armed Bandits with Graph Information”. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 2019–2032.
- [24] Naval Oceanographic Office Acoustics Division. *Database description for bottom sediment type (U)*. 2003.
- [25] W. A. Kuperman and F. Ingenito. “Spatial correlation of surface generated noise in a stratified ocean”. In: *The Journal of the Acoustical Society of America* 67.6 (June 1980), pp. 1988–1996. ISSN: 0001-4966. DOI: 10.1121/1.384439.
- [26] C. H. Harrison. “Formulas for ambient noise level and coherence”. In: *The Journal of the Acoustical Society of America* 99.4 (Apr. 1996), pp. 2055–2066. ISSN: 0001-4966. DOI: 10.1121/1.415392.

- [27] S.M. Kay. *Fundamentals of Statistical Signal Processing: Detection theory*. Fundamentals of Statistical Si. Prentice-Hall PTR, 1998. ISBN: 9780133457117.
- [28] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning series. MIT Press, 2005. ISBN: 9780262182539.
- [29] R. Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2018. ISBN: 9781108415194.
- [30] Blake Mason et al. “Nearly Optimal Algorithms for Level Set Estimation”. In: *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*. Ed. by Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera. Vol. 151. Proceedings of Machine Learning Research. PMLR, 28–30 Mar 2022, pp. 7625–7658.
- [31] S. Shalev-Shwartz. *Online Learning and Online Convex Optimization*. Foundations and Trends® in Machine Learning Series. Now Publishers, 2012. ISBN: 9781601985460.
- [32] Lorenzo Bottarelli et al. “Skeleton-Based Orienteering for Level Set Estimation”. In: (Sept. 2016). DOI: 10.3233/978-1-61499-672-9-1256.
- [33] T. Lattimore and C. Szepesvári. *Bandit Algorithms*. Cambridge University Press, 2020. ISBN: 9781108486828.
- [34] Michal Valko et al. “Finite-Time Analysis of Kernelised Contextual Bandits”. In: *Uncertainty in Artificial Intelligence*. 2013.

- [35] Yanan Sui et al. “Safe Exploration for Optimization with Gaussian Processes”. In: *Proceedings of the 32nd International Conference on Machine Learning*. Ed. by Francis Bach and David Blei. Vol. 37. Proceedings of Machine Learning Research. Lille, France: PMLR, July 2015, pp. 997–1005.
- [36] Paul Stankiewicz, Yew Teck Tan, and Marin Kobilarov. “Adaptive sampling with an autonomous underwater vehicle in static marine environments”. In: *Journal of Field Robotics* 38 (Dec. 2020). DOI: 10.1002/rob.22005.
- [37] Andrea Locatelli, Maurilio Gutzeit, and Alexandra Carpentier. “An optimal algorithm for the Thresholding Bandit Problem”. In: *Proceedings of The 33rd International Conference on Machine Learning*. Ed. by Maria Florina Balcan and Kilian Q. Weinberger. Vol. 48. Proceedings of Machine Learning Research. New York, New York, USA: PMLR, 20–22 Jun 2016, pp. 1690–1698.
- [38] F. Zhang. *The Schur Complement and Its Applications*. Numerical Methods and Algorithms. Springer, 2005. ISBN: 9780387242712.
- [39] Lihong Li et al. “A contextual-bandit approach to personalized news article recommendation”. In: *Proceedings of the 19th International Conference on World Wide Web*. WWW ’10. Raleigh, North Carolina, USA: Association for Computing Machinery, 2010, pp. 661–670. ISBN: 9781605587998. DOI: 10.1145/1772690.1772758.
- [40] Brent Bryan et al. “Active learning for identifying function threshold boundaries”. In: *Advances in neural information processing systems* 18 (2005).

- [41] Naval Oceanographic Office Acoustics Division. *Software design description and software test description for the multi-dimensional ambient noise model (MDANM) version 1.01 (U)*. 2022.
- [42] Jonathan Binney, Andreas Krause, and Gaurav S Sukhatme. “Informative path planning for an autonomous underwater vehicle”. In: *2010 IEEE International Conference on Robotics and Automation*. IEEE. 2010, pp. 4791–4796.
- [43] Pierre FJ Lermusiaux et al. “Science of autonomy: Time-optimal path planning and adaptive sampling for swarms of ocean vehicles”. In: *Springer Handbook of Ocean Engineering*. Springer, 2016, pp. 481–498.

8 Appendix: Lookahead Error Bound

Valko et. *al* provides

$$|\mu_{a,t}^{(s)} - \phi(x_{a,t})^T \omega^*| \leq \bar{\eta} \sigma_{a,t}^{(s)}, \quad (8.1)$$

where $\bar{\eta} = \eta(1 + \|\omega^*\|) + \gamma^{1/2}\|\omega^*\|$ and each group $\{\Psi^{(s)}\}_s$ is mutually exclusive. This allows the Azuma-Hoeffding inequality to be applied. We will eliminate the index sets to simplify superscripts and subscripts resulting in

$$|\mu_t(x) - \phi(x)^T \omega^*| \leq \bar{\eta} \sigma_t(x), \quad \forall x \in \mathcal{D}. \quad (8.2)$$

The difference decomposes into

$$\mu_t(x) - \phi(x)^T \omega^* = \phi(x)^T (C_t^\gamma)^{-1} \Phi_t^T (y_t - \Phi_t \omega^*) - \gamma \phi(x)^T (C_t^\gamma)^{-1} \omega^*, \quad \forall x \in \mathcal{D}. \quad (8.3)$$

Looking at Eq. 8.3, a lookahead operation forms an estimated \tilde{y}_{t+1} by appending synthetic reward \tilde{r}_{t+1} to y_t . While our work practically benefits from a pessimistic sampling protocol, we are only considering a mean estimate fake sample in this proof producing

$$\tilde{\mu}_{t+1}(x) - \phi(x)^T \omega^* = \phi(x)^T (C_{t+1}^\gamma)^{-1} \Phi_{t+1}^T (\tilde{y}_{t+1} - \Phi_{t+1} \omega^*) - \gamma \phi(x)^T (C_{t+1}^\gamma)^{-1} \omega^*, \quad (8.4)$$

and

$$\phi(x)^T C_{t+1}^{-1} \Phi_t^T (\tilde{y}_{t+1} - \Phi_{t+1} \omega^*) = \phi(x)^T C_{t+1}^{-1} \Phi_{t+1}^T \begin{bmatrix} r_1 - \phi(x_1)^T \omega^* \\ \vdots \\ r_t - \phi(x_t)^T \omega^* \\ \tilde{r}_{t+1} - \phi(\tilde{x}_{t+1})^T \omega^* \end{bmatrix}, \quad (8.5)$$

where $\tilde{y}_{t+1} = [y_t, \tilde{r}_{t+1}] = [r_1, r_2, \dots, r_t, \tilde{r}_{t+1}]$. We can split this up into

$$\begin{aligned} \tilde{\mu}_{t+1}(x) - \phi(x) \omega^* &= \phi(x)^T C_t^{-1} \Phi_t^T (y_t - \Phi_t \omega^*) + \\ &\quad [\phi(x)^T C_{t+1}^{-1} \Phi_{t+1}^T]_{t+1} (\mu_t(\tilde{x}_t) - \phi(\tilde{x}_t)^T \omega^*) - \gamma \phi(x)^T (C_{t+1}^\gamma)^{-1} \omega^* \end{aligned} \quad (8.6)$$

which we can bound using techniques from the original proof. Specifically, we can use the Cauchy-Schwarz inequality to find that

$$\begin{aligned} |\phi(x)^T C_{t+1}^{-1} \Phi_t^T| &\leq \|\omega^*\| \sqrt{\phi(x)^T C_{t+1}^{-1} \gamma^{-1} \gamma I C_{t+1}^{-1} \phi(x)} \\ &\leq \gamma^{-1/2} \|\omega^*\| \sqrt{\phi(x)^T C_{t+1}^{-1} C_{t+1} C_{t+1}^{-1} \phi(x)} \\ &\leq \gamma^{-1/2} \|\omega^*\| \sigma_{t+1}(x) \\ &\leq \gamma^{-1/2} \|\omega^*\| \sigma_t(x). \end{aligned} \quad (8.7)$$

Additionally, we can bound the first term using the Azuma-Hoeffding inequality such that

$$\mathbb{P}(|\phi(x)^T C_t^{-1} \Phi_t^T (y_t - \Phi_t \omega^*)| > (1 + \|\omega^*\|) \eta \sigma_t(x)) \leq 2e^{-\eta^2/2}. \quad (8.8)$$

Now, Eq. (8.6) looks like

$$\begin{aligned}
\tilde{\mu}_{t+1}(x) - \phi(x)\omega^* &\leq (1 + \|\omega^*\|)\eta\sigma_t(x) + \gamma^{-1/2}\|\omega^*\|\sigma_{t+1}(x) + \\
&\quad [\phi(x)^T C_{t+1}^{-1} \Phi_{t+1}^T]_{t+1} (\mu_t(\tilde{x}_t) - \phi(\tilde{x}_t)^T \omega^*) \\
&\leq \bar{\eta}\sigma_t(x) + [\phi(x)^T C_{t+1}^{-1} \Phi_{t+1}^T]_{t+1} (\mu_t(\tilde{x}_t) - \phi(\tilde{x}_t)^T \omega^*)
\end{aligned} \tag{8.9}$$

where we only need to bound the middle term.

For a mean-estimate lookahead, $\tilde{r}_{t+1} = \mu_t(\tilde{x}_t)$, where $\mu_t(\tilde{x}_t)$ is the most recently mean estimate at lookahead sampling location, \tilde{x}_t . To begin bounding Eq. (8.2), we start with

$$\begin{aligned}
\tilde{r}_t - \phi(\tilde{x}_t)^T \omega^* &= \tilde{r}_t - r_t + r_t - \phi(\tilde{x}_t)^T \omega^* \\
&= \mu_t(\tilde{x}_t) - r_t + r_t - \phi(\tilde{x}_t)^T \omega^*.
\end{aligned} \tag{8.10}$$

We can bound $r_t - \phi(x)^T \omega^*$ using Hoeffding's inequality

$$\mathbb{P}(|r_t - \phi(x)\omega^*| \geq \epsilon) \leq 2 \exp\left(-\frac{\epsilon^2}{2R}\right), \tag{8.11}$$

where \mathcal{R} is from the resulting \mathcal{R} -subGaussian. This forms

$$|r_t - \phi(x)\omega^*| \leq \bar{\eta}\sigma_t(x) \text{ w.p. } \geq 1 - 2 \exp\left(-\frac{\bar{\eta}^2 \sigma_t^2(x)}{2R}\right), \tag{8.12}$$

where r_t comes from sampling x . Returning to Eq. (8.10), we can bound using

Eq. (8.2) and Eq. (8.12)

$$\begin{aligned}
|\mu_t(x) - \phi(x)\omega^*| &\leq \bar{\eta}\sigma_t(x) \\
|\mu_t(x) - r_t + r_t - \phi(x)\omega^*| &\leq \bar{\eta}\sigma_t(x) \\
|\mu_t(x) - r_t| \pm \bar{\eta}\sigma_t(x) &\leq \bar{\eta}\sigma_t(x) \\
|\mu_t(x) - r_t| &\leq 2\bar{\eta}\sigma_t(x)
\end{aligned} \tag{8.13}$$

Reproducing Eq. (8.9) with lookahead values \tilde{r}_{t+1} and \tilde{x}_{t+1} , the most updated bound looks as follows

$$|\tilde{\mu}_t(x) - \phi(x)^T \omega^*| \leq \bar{\eta}\sigma_t(x) + 2\bar{\eta}\sigma_t(x) [\phi(x)^T C_{t+1}^{-1} \Phi_{t+1}^T]_{t+1} \tag{8.14}$$

where $\tilde{\mu}(\cdot)$ is the lookahead mean estimate having sampled location \tilde{x}_t and received measurement $\mu_t(\tilde{x}_t)$. Now, we want to bound $[\phi(x)^T C_t^{-1} \Phi_t^T]_{t+1}$.

$$\begin{aligned}
[\phi(x)^T C_{t+1}^{-1} \Phi_{t+1}^T]_{t+1} &= [\phi(x)^T C_t^{-1} [\phi(x_1), \phi(x_2), \dots, \phi(\tilde{x}_{t+1})]]_{t+1} \\
&= [\phi(x)^T C_{t+1}^{-1} \phi(x_1), \dots, \phi(x)^T C_{t+1}^{-1} \phi(\tilde{x}_{t+1})]_{t+1} \\
&= \phi(x)^T C_{t+1}^{-1} \phi(\tilde{x}_{t+1})
\end{aligned} \tag{8.15}$$

To bound $\phi(x)^T C_{t+1}^{-1} \phi(\tilde{x}_{t+1})$, we use the following lines from Valko

$$\begin{aligned}
(\Phi_t^T \Phi_t + \gamma I) \Phi_t^T &= \Phi_t^T (\Phi_t \Phi_t^T + \gamma I) \\
\Phi_t^T (\Phi_t \Phi_t^T + \gamma I)^{-1} &= (\Phi_t^T \Phi_t + \gamma I)^{-1} \Phi_t^T
\end{aligned} \tag{8.16}$$

and $\phi(x) = \Phi_t^T(\Phi_t\Phi_t^T + \gamma I)^{-1}k_{x,t} + \gamma(\Phi_t^T\Phi_t + \gamma I)^{-1}\phi(x)$. Therefore

$$\begin{aligned}
\phi(\tilde{x}_{t+1}) &= \Phi_{t+1}^T(\Phi_{t+1}\Phi_{t+1}^T + \gamma I)^{-1}k_{\tilde{x}_{t+1},t+1} + \gamma(\Phi_{t+1}^T\Phi_{t+1} + \gamma I)^{-1}\phi(\tilde{x}_{t+1}) \\
\phi(x)^T\phi(\tilde{x}_{t+1}) &= \phi(x)^T(\Phi_{t+1}^T(\Phi_{t+1}\Phi_{t+1}^T + \gamma I)^{-1}k_{\tilde{x}_{t+1},t+1} + \\
&\quad \gamma(\Phi_{t+1}^T\Phi_{t+1} + \gamma I)^{-1}\phi(\tilde{x}_{t+1})) \\
&= \phi(x)^T\Phi_{t+1}^T(\Phi_{t+1}\Phi_{t+1}^T + \gamma I)^{-1}k_{\tilde{x}_{t+1},t+1} + \\
&\quad \gamma\phi(x)^T(\Phi_{t+1}^T\Phi_{t+1} + \gamma I)^{-1}\phi(\tilde{x}_{t+1})
\end{aligned}$$

We can swap sides of the equation for convenience,

$$\begin{aligned}
&\phi(x)^T\Phi_{t+1}^T(\Phi_{t+1}\Phi_{t+1}^T + \gamma I)^{-1}k_{\tilde{x}_{t+1},t+1} + \\
&\quad \gamma\phi(x)^T(\Phi_{t+1}^T\Phi_{t+1} + \gamma I)^{-1}\phi(\tilde{x}_{t+1}) = \phi(x)^T\phi(\tilde{x}_{t+1}) \\
&\gamma\phi(x)^T(\Phi_{t+1}^T\Phi_{t+1} + \gamma I)^{-1}\phi(\tilde{x}_{t+1}) = \\
&\quad \phi(x)^T\phi(\tilde{x}_{t+1}) - \phi(x)^T\Phi_{t+1}^T(\Phi_{t+1}\Phi_{t+1}^T + \gamma I)^{-1}k_{\tilde{x}_{t+1},t+1} \\
&\phi(x)^T(\Phi_{t+1}^T\Phi_{t+1} + \gamma I)^{-1}\phi(\tilde{x}_{t+1}) = \\
&\quad \frac{1}{\gamma}(\phi(x)^T\phi(\tilde{x}_{t+1}) - \phi(x)^T\Phi_{t+1}^T(\Phi_{t+1}\Phi_{t+1}^T + \gamma I)^{-1}k_{\tilde{x}_{t+1},t+1}) \\
&\phi(x)^TC_{t+1}^{-1}\phi(\tilde{x}_{t+1}) = \\
&\quad \frac{1}{\gamma}(\phi(x)^T\phi(\tilde{x}_{t+1}) - \phi(x)^T\Phi_{t+1}^T(\Phi_{t+1}\Phi_{t+1}^T + \gamma I)^{-1}k_{\tilde{x}_{t+1},t+1}) \\
&\phi(x)^TC_{t+1}^{-1}\phi(\tilde{x}_{t+1}) = \frac{1}{\gamma}(k(x, \tilde{x}_{t+1}) - k_{x,t+1}^T(K_{t+1} + \gamma I)^{-1}k_{\tilde{x}_{t+1},t+1}) \quad (8.17)
\end{aligned}$$

Our final bound is complete

$$|\tilde{\mu}_t(x) - \phi(x)^T \omega^*| \leq \bar{\eta} \sigma_t(x) + \frac{2\bar{\eta}}{\gamma} \sigma_t(x) |k(x, \tilde{x}_{t+1}) - k_{x,t+1}^T (K_{t+1} + \gamma I)^{-1} k_{\tilde{x}_{t+1},t+1}| \quad (8.18)$$

which proves we do as poorly as the original bound plus additional variance related to the lookahead measurement location. This makes sense as we are not actually sampling a location but using an estimate for our measurement.