Enabling Human-Multi-Robot Collaborative Visual Exploration in Underwater Environments

by

Stewart Christopher Jamieson

B.A.Sc. in Engineering Science, University of Toronto, 2018
S.M. in Aeronautics and Astronautics, MIT, 2020

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY IN AUTONOMOUS SYSTEMS

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY and
WOODS HOLE OCEANOGRAPHIC INSTITUTION

May 2024

Authored by:	Stewart Christopher Jamieson Joint Program in Oceanography/Applied Ocean Science and Engineering March 29, 2024
Certified by:	Jonathan P. How R. C. Maclaurin Professor of Aeronautics and Astronautics, MIT Thesis Chair
Certified by:	Yogesh Girdhar Associate Scientist in Applied Ocean Physics and Engineering, WHOI Thesis Supervisor
Accepted by:	Jonathan P. How R. C. Maclaurin Professor in Aeronautics and Astronautics, MIT Chair, Graduate Program Committee
Accepted by:	Alexandra H. Techet Professor of Ocean & Mechanical Engineering, MIT Chair, Joint Committee for Applied Ocean Science & Engineering

Enabling Human-Multi-Robot Collaborative Visual Exploration in Underwater Environments

by

Stewart Christopher Jamieson

Submitted to the Department of Aeronautics and Astronautics
on March 29, 2024 in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY IN AUTONOMOUS SYSTEMS

ABSTRACT

This thesis presents novel approaches to vision-based autonomous exploration in underwater environments using human-multi-robot systems, enabling robots to adapt to evolving mission priorities learned via a human supervisor's responses to images collected *in situ*. The robots model the spatial distribution of various habitats and terrain types in the environment using semantic classes learned online, and send image queries to the supervisor to learn which of these classes are associated with the highest concentration of targets of interest. The robots do not require prior examples of these targets, and learn these concentration parameters online. This approach is suitable for exploration in unfamiliar environments where unexpected phenomena are frequently discovered, such as coral reefs. A novel risk-based online learning algorithm identifies the concentration parameters using the fewest possible number of queries, enabling the robots to adapt quickly and reducing the operational burden on the supervisor.

I introduce four primary contributions to address prevalent challenges in underwater exploration. Firstly, a multi-robot semantic representation matching algorithm enables inter-robot sharing of semantic maps, generating consistent global maps with 20-60% higher quality scores than those produced by other methods. Next, we present DeepSeeColor, a novel real-time algorithm for correcting underwater image color distortions, which achieves up to 60 Hz processing speeds, thereby enabling improved semantic mapping and target recognition accuracy online. Thirdly, an efficient risk-based online learning algorithm ensures effective communication between robots and human supervisors, and, while remaining computationally tractable, overcomes the myopia which would cause previous algorithms to underestimate a query's value. Lastly, we propose a new reward model and planning algorithm tailored for autonomous exploration, together enabling a 25-75% increase in the number of targets of interest located when compared to baseline surveys. These experiments were conducted with simulated robots exploring real coral reef maps and with real, ecologically meaningful targets of interest.

Collectively, these contributions overcome key barriers to vision-based autonomous underwater exploration, and enhance the capability of autonomous underwater vehicles to adapt

to new and evolving mission objectives *in situ*. Beyond marine exploration, these contributions have value in broader applications, such as space exploration, ecosystem monitoring, and other online learning problems.

Thesis Chair: Jonathan P. How
Title: R. C. Maclaurin Professor of Aeronautics and Astronautics, MIT

Thesis Supervisor: Yogesh Girdhar
Title: Associate Scientist in Applied Ocean Physics and Engineering, WHOI

Acknowledgments

This thesis was made possible by the bountiful support and guidance that I am so grateful to have received over the last several years. To begin, I'd never have embarked on this journey if not for the infectious enthusiasm of my research advisor, Dr. Yogi Girdhar, who convinced me to dive headfirst into marine robotics. I thank him for always bringing positive energy to our meetings and constantly introducing me to new and exciting opportunities and ideas.

I am also deeply grateful to Professor Jonathan How, for giving me a home in the Aerospace Controls Lab and for demonstrating incredible generosity with his time and guidance over the years as my co-advisor. His feedback has always steered my work in the right direction while his mentorship has always directed me towards success.

I sincerely thank Professor Julie Shah and Dr. John Fisher for their support and feedback as committee members. They have each provided insightful commentary on my work and encouraged me to consider broader applications and research parallels. I extend this thanks to my thesis readers, Professor Florian Shkurti and Dr. Seth McCammon, who have generously offered their time and feedback on my work.

My peers and colleagues in the WARPLab and ACL consistently inspire me and have all been wonderful to work with. I couldn't have asked for better mentors than Seth, Kaveh, and Kasra, and I'm deeply grateful to Vv, John, Nathan, Jess, Dan, Dong-Ki, Genevieve, Kevin and Victoria P. for the many fascinating discussions we've had over the years across nearly all areas of robotics research. I can't thank Nathan, Vv, and Patrick enough for their roles in developing and maintaining the AUVs which collected nearly all of the real-world data used in this thesis, and my gratitude extends to many more of WHOI's incredible staff engineers. Lastly, my USVI fieldwork collaborators were similarly essential in supporting real-world data collection and system testing, and made our trips to St. John some of the best parts of my graduate student experience.

My friends were the bedrock that kept me grounded throughout the course of this work. My heartfelt thanks goes out to Jordan & the Sea Trek crew, Seth's MLS watch parties, Theo & Matt, Soumya & Andres, Jane & Aaron, and Ian, for giving me an abundance of joyful memories with which to remember my time in graduate school. I am also deeply grateful for the wonderful MIT-WHOI Joint Program community in its entirety.

I have a very special thanks for my family. Victoria Davis has been at my side through the best and the worst of times, and always inspires me with her unparalleled dedication and passion for research; I owe so much of my happiness and success to her endless love and

support. My brother Chris and his wife Victoria have made Boston an even more fun place to be since moving out here, and the opportunities to spend more time with them these past couple years have been lovely. My mom and my dad have been my greatest supporters from the start, and I couldn't be more thankful for their constant love and encouragement. I also feel so very fortunate for the incredible love and support of Suzy, Rob, and my entire extended family: the Jamiesons, the Barnetts, the Davises, the Owens and the Martyns.

Lastly, I'd like to thank some key figures from my academic career for their positive influences and foundational mentorship: Prof. Jonathan Kelly, Prof. Dr. Angela Schoellig, and Prof. Gabriele D'Eleuterio; Dr. Raymond Phan and Dr. Richard Rzeszutek; and Denise Grightmire. My thanks are also due to the National Science Foundation for funding much of this research, and to the WHOI Academic Programs Office and the department admins and admin assistants who have supported me in countless ways throughout my time in the MIT-WHOI Joint Program (Kris, Lea, Tricia N., Heather, Erinn and Bryt especially!).

Funding: This work was partially supported by the National Science Foundation (NSF-NRI Awards 1734400 and 2133029), NOAA Ocean Exploration Award NA19OAR4320072, the NVIDIA Academic Hardware Grant Program, as well as by the Woods Hole Oceanographic Institution (WHOI) Investment in Science Fund. Chapter 3 includes work with coauthors partially supported by ARL DCIST under Cooperative Agreement Number W911NF-17-2-0181, and by ONR under BRC Award N000141712072.

Contents

List of Figures

List of Tables

Glossary

Term	**Definition**
AUV	Autonomous Underwater Vehicle
ROV	Remotely Operated Vehicle
Semantic Mapping	The process of creating a map that uses different labels for visually distinct regions (e.g. habitats or terrain types), such that it captures the high-level structure of an environment.
Online (or "*in situ*")	Refers to things that happen during a robot deployment/mission.
Online Learning	A machine learning approach where a model is continuously updated online as new data is collected/received, rather than being trained in advance with a fixed dataset.
Risk-Based Learning	An online learning strategy where decisions on what labelled data to acquire next are made based on the potential impact of the new information to reduce *risk*.
Regret	Measures how much better a robot could have performed in a task (i.e. how much its plan could have been).
Risk	Measures the amount of regret expected to be incurred by the robot in a task, based on its current plan and knowledge of the world.
Utility	Measures how much some specific information helps to reduce risk.
Plan	A sequence of actions to be taken; often, the planned trajectory of a robot.
Policy	A function that produces a plan given the robot's current state and understanding of the world.
Open-Loop	Refers a robot following a specific plan while ignoring any further observations.
Closed-Loop	Refers to a robot following a policy that adapts the plan online in response to new information that changes the robot's understanding of the world.
Spatiotemporal	Relating to or involving both space and time.
Semantic Class	Categories used in semantic mapping to differentiate between types of terrain, objects, or other relevant features.

Chapter 1

Introduction

Exploration plays a key role in many applications ranging from protected ecosystem monitoring to seeking out new life or other phenomena in extreme environments; it is an activity carried out with the expectation of making new *discoveries* in these places. Sites targeted for exploration are generally remote, unfamiliar, and/or dynamic, qualities which make novel or unexpected discoveries more likely. For example, unusual topographical map features may inspire a search for hydrothermal vents in unexpected parts of the deep sea [1],[1] or multispectral satellite imagery may suggest the presence of coral reefs along remote coasts [2], motivating higher-resolution *in situ* exploration to characterize possible new species and reef compositions [3]. Meanwhile, the routine monitoring of underwater assets or ecosystems is often motivated by the potential discovery of emergent threats, such as invasive species, diseases, or biofouling, which often demand immediate follow-up investigations or interventions. An exploration mission begins with only an educated guess of what might be found; one cannot be certain about what the most important priorities will be until after the mission has started and new discoveries have been made.[2]

In the practices of underwater exploration, humans and robots have traditionally occupied distinct roles. Human explorers, equipped with intuition and experience, recognize immediately when new and interesting phenomena are discovered and react by directing their search towards them. Conversely, autonomous underwater vehicles (AUVs) typically operate under rigid, pre-defined instructions without the capacity to react to new discoveries. This difference highlights a significant gap: the ability to set and adapt to changing mission priorities. The challenge lies in enabling AUVs to recognize when novel or unexpected phenomena merit more directed investigation. While there has been substantial recent progress

[1] Large-area topographical seafloor maps can be collected at moderate resolutions with multibeam sonar arrays, or at much lower resolution by radio mapping from satellites.

[2] For example, after discovering disease or biofouling, a new objective may be to determine its extent.

in adaptive AUV exploration of scalar fields, with applications that include locating high chemical concentrations [4] or temperature anomalies [5], there has been minimal progress in adaptive AUV exploration based on *vision* or other high-dimensional sensing modalities. The work described in this thesis has been largely motivated by the question: **can we enable AUVs to effectively learn and adapt to dynamic mission priorities, like a human would, in vision-based underwater exploration?**

The focus on visual exploration is driven by the role of vision as one of the most widespread and versatile sensing modalities. Visual observations in the form of images are, for the majority of individuals, the most familiar and information-rich format for the perception and classification of objects, organisms, and other physical phenomena. This information richness is possible due to the high dimensionality of images relative to most other data representations, such as audio or text.[3] While specialized sensors can be better suited for some specific tasks (e.g., DNA sequencers for species identification), enabling AUVs to perceive and adapt to visual observations helps to broadly bridge the gap between human and robot exploration behaviors across many applications. Furthermore, while many of the contributions to be presented in this thesis are similarly applicable to enabling adaptive mission behaviors with respect to other high-dimensional sensing modalities, vision serves as an intuitive and familiar modality to use as an example.

Despite significant advancements in computer vision, vision-based autonomous robots typically require example images in order to recognize and focus upon targets of interest. The intrinsic value of new discoveries often stems from their unpredictability and uniqueness, qualities that defy anticipation and make such discoveries challenging for robots to identify and assess the value of autonomously. This limitation undermines the use of autonomous robots for uncovering novel and valuable findings in uncharted environments, and has led to the dominance of two *non-autonomous* robotic exploration paradigms. The first is the use of *remotely operated vehicles* (ROVs), which are piloted by a human operator with a live video feed. This approach enables the vehicle to act as an avatar of a human explorer, but is limited to situations where high-bandwidth communications can be consistently maintained with the vehicle; for underwater exploration, this generally requires the use of a tether, which drastically complicates operational complexity and reduces feasible deployment areas.[4]

The second paradigm is autonomous *surveys*, where the robot follows a pre-planned trajectory; these are often box surveys, also known as "lawnmower" patterns. Key advantages of this paradigm are its simplicity and the ease at which it can be scaled to a multi-robot fleet

[3]Other high-dimensional sensing modalities (e.g. multibeam sonar) often have their observations converted to and presented as color images, especially for the purposes of human analysis.

[4]This complexity is further compounded by simultaneous deployment of multiple vehicles.

deployment to cover larger areas. The only decision to make is the structure of the survey; it can be dense, covering a small area thoroughly, or sparse, spanning a wider area but with observational gaps. Sparse surveys are superior at discovering as many new phenomena as possible across an unknown region in a fixed amount of time because they maximize the number of distinct *domains* (habitats or terrain types) surveyed. Domains such as coral reefs, seagrass meadows, and sand bars can be tens to hundreds of meters long, while individual corals, fish, and other potential phenomena of interest are generally found in some domains and not others. By covering a more diverse range of domains, sparse surveys are more likely to discover *a priori* unknown phenomena of interest. However, this approach also increases the risk of merely grazing past these interesting phenomena without detailed examination, as the AUV spends minimal time in any specific domain; in essence, **optimizing a pre-planned survey to discover *any* interesting phenomena inherently reduces the chances of extensively finding *many* such phenomena**. As such, sparse surveys must often be followed by more targeted dense surveys the domains found to contain the most phenomena of interest; this is inefficient and expensive, if not infeasible,[5] and may fail if the phenomena are transient or if the fleet cannot easily relocate the original site after recovery and redeployment.

I propose a more effective behavioral paradigm for autonomous robotic explorers, which mirrors strategies employed by human operators. The robot should begin with a sparse survey, to cover a broad area. However, upon encountering an unfamiliar or unexpected phenomenon, it should assess whether it is "interesting" enough to merit more thorough investigation. If so, it should transition to a denser survey around where this phenomenon was detected, potentially using other onboard sensors or samplers for more in-depth analysis,[6] before carrying on with the sparse survey to look for more phenomena of interest. Importantly, the robot should infer correlations between these interesting phenomena and broader features that may be used to locate more of them. For example, if an invasive species is consistently discovered in particular habitat type, such as the rocky outcroppings of a coral reef, the robot should more thoroughly survey such habitats if the goal is to locate more of them in order to better measure their extent.

As discussed previously, the main challenge in enabling this behavior is preparing robots to visually recognize any "interesting" phenomenon when we cannot provide a comprehensive set of examples of everything that would be interesting. Until the day that we can build robots with comparable knowledge and intuition to human experts, the visual determina-

[5] As it often is, as an example, for large-ship operations with scheduled waypoints.

[6] For example, AUVs are often equipped to collect a finite number of water samples around interesting phenomena, while the Mars Curiosity rover Sample Analysis Tool performs detailed analysis of geologic materials using lasers [6]. These tools have costs that motivate limiting their to only high-interest targets.

tion of targets of interest will require some degree of human-robot communication; that is, a *human-in-the-loop* solution. Unfortunately, communication bandwidth tends to be very limited in places that need to be explored; this is especially true for ocean exploration. Long-range ($>$ 100 m) underwater communications devices only support data transfer rates up to around 10 kilobits per second, which is about one or two 50 kB images per minute. State of the art underwater exploration relies on the AUV sending back infrequent, low-resolution images, which are a very sparse subset of the complete set of observations collected. If an image is determined, by a human supervisor, to contain a phenomenon of interest, the initial, pre-programmed sparse survey can be modified remotely by the supervisor in an attempt to better target this phenomenon (or, to target the habitats in which it is most likely to be found).

This brings us to the cornerstone of this thesis: **we propose that the most effective way to handle new and evolving mission objectives in vision-based exploration is to model the co-occurrence of targets of interest with visually distinct domains (e.g., terrain or habitat types).** Learning such co-occurrence relationships online enables targeted exploration that exploits the spatial structure of these domains to efficiently discover more targets of interest *in situ*, without the need for prior examples or training. Fortunately, relatively recent developments in *unsupervised spatiotemporal semantic modeling* enable robots to autonomously categorize visually distinct and *a priori* unknown terrain and habitat types without human supervision [7]–[10], as well as create *semantic maps* which model the spatial distribution of these domains [11], [12]. These semantic maps tend to be continuous and highly spatially auto-correlated, enabling some degree of interpolation and extrapolation into regions where the robot has yet to visit [11], [12]. Henceforth, we will use terms "semantic classes", "domains", and "terrain/habitat types" interchangeably.

My own journey into vision-based robotic exploration began by developing an online learning algorithm to enable an AUV to identify which of the images it has collected would be most *useful* to send back to its supervisor, assuming that the supervisor will reply back with corresponding "interest labels" [13]. The robot uses these labels to learn a *reward model* which guides it towards places where it expects to find more targets of interest. Specifically, the robot learns which domains are most correlated with positive labels, and plans a path that goes wherever it expects to find more of these domains (based on the extrapolated semantic map). The *utility* (equivalently, *risk*) of an image query measures the sensitivity of the robot's plan to the label assigned to that query. It is large if either a positive or negative label would cause the robot to re-evaluate some alternative plan as being much more effective than its current plan, and small (or zero) if either label would have minimal impact on the robot's plan. Intuitively, learning that a phenomenon of interest is correlated

(or anti-correlated) with a particular domain is useless if the robot doesn't know where to find more of that domain, or if the only places it would still find that domain are too far away. Instead, the robot should send image queries that could produce *actionable* results, such that the supervisor's label guides the robot towards or away from one or more domains.

This is an example of *risk-based* online learning, and in basic simulations it enabled a robot explorer to locate far more targets of interest than when sending the supervisor images selected at random, or other baseline strategies [13]. Furthermore, by directly quantifying the utility (risk-reduction) of each query, the robot is able to decide whether it is even worth asking. If no new label from the supervisor would cause the robot to change its plan,[7] the robot can skip making a query to mitigate the cognitive and operational burden on the supervisor, enabling them to focus on other tasks. This is particularly beneficial for multi-robot exploration of large areas, as it enables prioritizing query requests from multiple robots (based on the estimated utility).

Equipped with unsupervised spatiotemporal semantic mapping systems and a risk-based online learning algorithm to efficiently learn (over low-bandwidth) which of these classes are most correlated with targets of interest, it would seem the pieces were in place to enable large-scale autonomous vision-based exploration of the earth's oceans. Unfortunately, things are rarely so simple in the field; aside from the usual challenges of AUV operations,[8] we encountered four practical barriers to the integration of unsupervised spatiotemporal semantic mapping with online reward learning in real-world marine exploration. We present them below, followed by the contributions made by this thesis to overcome them.

1.1 Problem Statements

As discussed above, we propose that autonomous vision-guided robotic explorers make use of *unsupervised spatiotemporal semantic mapping* and *online learning* to effectively learn and address new and evolving mission priorities. However, four key problems were identified in the pursuit of using these technologies to enable large-scale, human-in-the-loop autonomous exploration of the Earth's oceans. We enumerate them below.

1. **The Egocentricity of Learned Semantic Representations Prevents Multi-Robot Collaboration.**

   Unsupervised semantic models learn, over time, a set of *semantic representations* which describe the various terrain and habitat types (domains) that the robot has encoun-

[7] This may happen if, for example, the robot has only found a small number of distinct domains and already asked several queries about each of them.

[8] These challenges include underwater SLAM, which continues to be an active research area [14].

tered. These representations define the semantic classes learned by the model, and evolve over time to reflect what the robot has seen. Accordingly, they are *egocentric* (unique to the robot which produced them), and so different robots may learn similar, but not exact, semantic representations for the same domains.[9] Without knowing these correspondences, robots cannot interpret each other's semantic maps, and each robot needs to learn its own co-occurrence model between its own semantic classes and targets of interest. The extra labelling required by the human supervisor for each additional robot makes larger fleets proportionally harder to manage, while the inability to share semantic maps prevents collaborating for greater exploration efficiency.

2. **Underwater Image Distortions Degrade Learned Semantic Maps/Models.**

   Light attenuation and backscattering effects are much stronger underwater than in air. This means that an object farther from an underwater camera tends to look darker, more blue, and more "hazy" than if it were closer. As color and intensity (brightness) features in images are important for automated recognition of different terrain and habitat types, this can result in inconsistent domain classification depending on whether a scene is viewed from close up or far away. With unsupervised semantic models, this manifests as multiple semantic classes being used to represent the same domain (even by the same robot), thereby creating inconsistencies in the semantic map; this makes the semantic maps harder to extrapolate and generally degrades their utility for adaptive exploration. Furthermore, these image distortions can make it more difficult for human supervisors to recognize the image contents, or to recognize the same type of phenomenon in multiple images, and can thereby result in erroneous interest labels.

3. **Risk-Based Online Learning Algorithms Are Either Myopic or Intractable.**

   The risk-based image query selection heuristic presented in [13] bears much in common with previous Bayes' risk criteria (e.g., [15], [16]) for POMDPs as well as knowledge-gradient algorithms for online learning [17]. However, the one-step lookahead strategy it uses to determine which query has the highest utility (risk reduction) is known to be overly *myopic* [18], [19]. This means that it tends to underestimate the utility of a single query, as the information gained from any single new label may be insufficient to change the robot's plan, even if some combination of *multiple* new labels could have a significant impact. Unfortunately, the complexity of identifying the value of multiple queries is exponential in the number of queries considered, and there are no known

[9] For example, one robot, operating in a part of reef where fire coral is relatively abundant, may use a semantic representation for "reef habitats" which reflects the features of fire coral more so than the representation used by another robot operating where fire coral is more sparse.

risk-based online learning algorithms which can efficiently approximate this value. As robotic explorers have relatively limited computing power, they are therefore restricted to these myopic heuristics which can select poor queries.

4. **Existing Reward Models and Planners Are Ill-Suited For Risk-Based Online Learning.**

   The core of risk-based online learning is considering *counterfactuals*: these describe what the robot would think and do if it were given some specific new information (for our purposes, a newly labelled query). The robot maintains a distribution over the possible reward model parameters as part of its *belief state*. By generating a random sample of responses that may be obtained for a given query, based on a (prior) belief state, we can generate counterfactual (posterior) belief states with refined reward model parameter distributions. The utility of a query is based on how inferior the robot's current plan would be when compared to the updated plan for the counterfactual belief state. While this risk-based approach provides the many advantages outlined previously, the computational cost of generating these counterfactuals beliefs and plans can be significant, as updating the reward model and plan often requires complex operations. For example, popular sampling-based path planners like MCTS [20] and RRT [21], [22] can require substantial computational resources to perform well, while updating even a simple probabilistic reward model can require expensive inference algorithms. As evaluating each candidate query generally requires considering multiple counterfactuals, we need reward models and planners that enable efficient computation (or approximation) of these counterfactual belief states.

1.2 Contributions

This thesis proposes a novel approach to vision-based exploration: by leveraging the strengths of unsupervised semantic mapping and online learning, we aim to bridge the gap between static mission programming and the adaptive, intuitive decision-making that characterizes human exploration. This research addresses the fundamental challenge of enabling robots to autonomously identify and respond to new and evolving mission priorities in unpredictable environments, which we accomplish by keeping a human supervisor in the loop and using efficient, strategic with them to guide the behavior of the robot(s). Specifically, the following contributions are solutions to the challenges presented in the previous subsection and, together, enable human-multi-robot collaborative vision-based exploration in marine environments.

1. **An Algorithm For Efficient Matching of Learned Semantic Representations**

   We present a system that enables each robot to use a multiway matching algorithm to efficiently identify consistent sets of matches between learned semantic classes belonging to different robots [23]. This enables fusing the semantic maps produced by multiple robots to produce *global* maps that use a unified set of semantic classes to label every location visited by any robot. Compared to the previous state of the art [24], the proposed solution produced 20-60% higher quality global maps; the largest improvements were observed when fusing many maps from many robots. The algorithm is computationally efficient, so each robot can feasibly integrate the semantic maps it receives by running the algorithm onboard. This system also enables the fleet to share reward labels and coordinate queries or any other behaviors that depend on the semantic classes or map.

   Impact: This enables a shared semantic map for use across the exploration fleet, significantly enhancing the scalability and efficiency of multi-robot coordinated exploration efforts.

2. **DeepSeeColor: Realtime Adaptive Underwater Color Correction**

   DeepSeeColor is a novel algorithm that combines state-of-the-art underwater image color correction techniques with the computational efficiency of deep learning frameworks [25]. In experiments, we show that DeepSeeColor offers comparable color reconstruction performance to leading methods while being able to process images much more rapidly, at up to 60Hz. This makes it suitable for use onboard AUVs as a preprocessing step to enable more robust vision-based behaviors in varying underwater conditions. In particular, this facilitates more accurate semantic mapping as well as human recognition and labeling of visual targets of interest.

   Impact: Solving the challenge of underwater image distortions is key to improving the accuracy and reliability of semantic mapping. This directly impacts a robot's ability to interpret and respond to its surroundings, making exploration efforts more efficient.

3. **An Efficient & Non-Myopic Risk-Based Online Learning Algorithm**

   We developed a general framework for risk-minimization across a wide range of problems, and incorporated an *asymptotic* utility (value of information) function that provides a computationally efficient solution to overcoming the myopia of previous risk-based online learning algorithms. We first established a POMDP formulation for online learning and used it to determine when risk-estimation is feasible. In particular, we identified that whenever there is an efficient algorithm to produce an optimal *open-loop*

plan,[10] this algorithm can be used to efficiently estimate both the one-step and *asymptotic* utility of a query. The (myopic) one-step utility is equivalent to the approach in [13], and measures the immediate value of information, while the (non-myopic) asymptotic term guides the robot towards queries that may be useful in aggregate.

Impact: This efficient and non-myopic algorithm avoids the potential pitfalls of previous strategies, while maintaining the advantages of risk-based online learning in guiding strategic communication.

4. **An Approximate-Probabilistic Reward Model and Fast Hotspot Planner**

   We present a novel "beta-rate" reward model which learns a distribution over the *concentrations* at which targets of interest are found in the various learned semantic classes. These concentrations are modeled as draws from a multivariate beta distribution, for which we developed an analytical approximate update rule that enables highly efficient construction of counterfactual posterior distributions. We further present a simple but effective hotspot-based greedy planner which identifies the regions of the map with the highest expected concentrations of reward, and efficiently plans dense coverage paths over these hotspots. These models are ideal for autonomous exploration, where plan quality is primarily limited by the available information, and so the cost of using inexact posteriors and sub-optimal greedy planners is outweighed by the improved information gathering (query selection) enabled by these models.

   Impact: This novel reward model and planner are well suited to enable vision-based autonomous exploration with risk-based online learning on even severely computationally constrained robotic platforms.

1.2.1 Publications

The contributions presented in this thesis appear in multiple published works, cited below.

- The contents of Chapter 3 are adapted from S. Jamieson, K. Fathian, K. Khosoussi, J. P. How, and Y. Girdhar, "Multi-robot distributed semantic mapping in unfamiliar environments through online matching of learned representations," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, ISSN: 2577-087X, Xi'an, China: IEEE, May 30, 2021, pp. 8587–8593. DOI: 10.1109/ICRA48506.2021.9561934. The majority of the text, figures, code and ideas presented in this chapter were my own, but formed and refined under the guidance of my co-authors Prof. Kaveh Fathian and

[10] That is, a plan which is optimal with respect to the robot's current reward model, assuming that no new information is later obtained/received.

Dr. Kasra Khosoussi. Their most active role in the text was in describing the related works and specifics of the relevant CLEAR algorithm (presented in [26]).

- The contents of Chapter 4 are adapted from S. Jamieson, J. P. How, and Y. Girdhar, "DeepSeeColor: Realtime adaptive color correction for autonomous underwater vehicles via deep learning methods," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, London, UK: IEEE, May 27, 2023, pp. 3095–3101. DOI: 10.1109/ICRA48891.2023.10160477. arXiv: 2303.04025[cs].

- The contents of Chapter 5, as well as some of the contents of Chapter 2, are adapted from S. Jamieson, J. P. How, and Y. Girdhar, "Finding the optimal exploration-exploitation trade-off online through bayesian risk estimation and minimization," *Artificial Intelligence*, p. 104 096, Feb. 21, 2024, ISSN: 0004-3702. DOI: 10.1016/j.artint.2024.104096.

- The contents of Chapter 6 are being prepared for submission to IEEE Transactions on Robotics in 2024 in a work jointly authored by S. Jamieson, J. P. How, and Y. Girdhar.

1.3 Overview

The remainder of the thesis is structured as follows. Chapter 2 presents background material on autonomous robotic exploration, semantic mapping, and online learning. Chapter 3 presents an approach for efficiently matching learned semantic representations online. Chapter 4 presents the DeepSeeColor algorithm for realtime color correction of underwater imagery. Chapter 5 presents a novel online learning approach, "Endogenous Bayesian Risk Minimization" (EBRM), and specifically the non-myopic risk-based online learning algorithm AsympGreedy-EBRM. Chapter 6 presents the novel beta-rate reward model and fast hotspot planner, as well as experiments demonstrating how these tools, in combination with AsympGreedy-EBRM, enable autonomous explorers to find up to 75% more targets of interest than a pre-planned survey in simulation experiments using real coral reef maps. Finally, Chapter 7 will review the main findings and contributions of this thesis, and discuss avenues of future work.

Chapter 2

Background

In this chapter we present some core content and concepts that are routinely referred throughout the thesis. As the contributions made in this thesis are quite diverse, later chapters will supplement this with their own specific background and related works sections as required.

2.1 Motivating Multi-Robot Marine Exploration

Autonomous robotic explorers are the best solution for the exploration and sustained monitoring of the vast expanse of Earth's oceans. The exploration and monitoring of marine environments is crucial for conservation efforts and for understanding key ecosystems under threat; for example, coral reefs are perhaps the most important marine ecosystem, yet are degrading at an alarming rate due to climate change and ocean acidification [28]–[31]. The inaccessibility of many reefs has historically prevented extensive study, while the global nature of such anthropogenic threats has now reached even the most remote atolls [3]; as these regions offer a rare view of relatively pristine reef ecosystems (for now) there is an urgent need for more thorough study of them. Furthermore, exploration can lead to groundbreaking discoveries, from species with unique medicinal properties [32]–[34] to rare mineral resources, which has provided extensive motivation for both sea and space exploration [35]–[40]. Other discoveries may have less immediate utility, yet still hold valuable, scientific, economic, or cultural significance; such discoveries include unique ecosystems, unknown species, or artifacts of historical importance [41]–[48], which each contribute to our understanding of Earth's diverse environments and history.

The vast scale of marine environments calls for the use of multi-robot fleets for exploration, which can cover far greater areas and at higher spatiotemporal resolutions than a single robot. Large fleets are particularly important in applications like widespread continuous environmental monitoring [49], [50]. Advancements in inter-robot collaboration can help

with these large-scale applications by enabling new capabilities and super-linear scaling of task performance [51], however to our knowledge there is no prior work on AUV collaboration for vision-based exploration tasks.

2.2 Spatiotemporal Semantic Mapping

Semantic mapping is a relatively young field that was initially motivated by giving robots a spatial awareness of nearby terrains, objects, and activities [52]. Semantic maps describe the world using a set of classes, and have been used with great effectiveness in solving many field robotics problems such as mission summarization [10], object-based SLAM [53]–[55], and context-aware planning [13], [56]. Great progress has been made in improving the accuracy of semantic mapping systems by leveraging deep learning models trained on large datasets [57]–[61]. However, most of these systems do not support *a priori* unknown classes, which are an essential part of autonomous exploration.

In autonomous exploration, semantic maps serve two key purposes; the first is spatial prediction. To collect more reward than a simple coverage path, a robot must predict what it will observe in new locations, and plan a trajectory along which it expects to find the most rewarding phenomena. Semantic fields have high spatial autocorrelation [11], making this spatial prediction problem far more tractable than for images. The second purpose is dimensionality reduction. In any particular area that a robot is deployed to explore, it will observe images from only a small subset of the full observation space; the robot need only approximate the supervisor's reward function over this subdomain to direct its exploration towards the most rewarding phenomena. Unsupervised semantic models use generally use a small number of dimensions to represent the variety of collected inputs, so the sample complexity of learning a function approximation over these semantic representations is far smaller it would over the space of images; this makes it more tractable to learn with very few examples.

Spatiotemporal topic models (STMs) [62]–[64] are a class of unsupervised learning algorithms that has been specifically augmented for realtime semantic mapping in unfamiliar environments [8], [65]. BNP-ROST is an STM that adaptively develops semantic labels online as new phenomena are observed [9], and has been used for 3D semantic mapping in bandwidth-limited environments without any pre-training [11]. STMs can create high quality semantic maps in realtime by leveraging the spatial and temporal correlations between observations and using efficient sampling algorithms to discover good semantic representations online [11], [66]. While there has been progress in other unsupervised semantic image segmentation approaches [67], including deep-learning based methods [68], by not leveraging

these correlations they produce lower quality maps of large environments, and are less likely to lead to accurate semantic map extrapolation.

In each of the following chapters, we use the STM from [11] for unsupervised semantic modelling. In brief, this approach extracts from each image many visual *words* $\{w_i\}$, which are elements of a *vocabulary* of size $V \in \mathbb{N}$. Through the use of corresponding depth images (produced by a stereo camera, or structure from motion), along with the robot's estimated SE(3) pose and the camera intrinsics, the image coordinates of each word are projected into world coordinates $\boldsymbol{r}_i \in \mathbb{R}^3$. Nearby words, in terms of world coordinates, are then grouped into the same *cell*. Each cell is modelled to have some latent distribution $\boldsymbol{\theta}$ over discrete "topics", where topic $k \in \mathbb{N}_{>0}$ is described by a probability distribution over the word vocabulary, $\boldsymbol{\phi}_k \in \Delta^V$. Each $\boldsymbol{\theta}$ is assumed to be generated from a spatiotemporally correlated Dirichlet process prior, while $k \sim \text{Categorical}(\boldsymbol{\theta})$ and $w_i \sim \text{Categorical}(\boldsymbol{\phi}_k)$. For every word w_i, we sample a topic k from the posterior distribution $P(k \mid w_i)$ as the semantic label for the word coordinates $\boldsymbol{r}_i$. The Dirichlet process prior permits an unbounded number of topics to be learned over time; the number of unique topics (semantic classes) grows logarithmically over time [13]. For more details on the parameter priors and inference procedures used, please refer to [8], [9].[1]

2.3 Online Learning

The field of online learning has been strongly motivated by practical applications since the seminal work of Thompson in the 1930s [70]. The original motivating idea of exploring the efficacy of a discrete set of medical treatments while simultaneously exploiting the leading treatments to save lives continues to be a research interest [71] and is an exemplar of the ubiquitous stochastic multi-armed bandit (MAB) problem [72]. In this terminology, the "arms" of the bandit represent, for example, different treatments. In stochastic multi-armed bandits, an observation is generated each round from an unknown distribution specific to the arm played that round, and the reward is the sum of the observations. Multi-armed bandits are a good structure with which to approximate many real-world problems, and there are a variety of successful online learning algorithms designed for them. Variations on this structure include infinite-time and time-discounted bandits, as well as best-arm identification.

Partial monitoring is a generalization of multi-armed bandits that allows for more general relationships between observations and rewards [73]. Partial monitoring problems are

[1] An efficient implementation of this model for the Robot Operating System (ROS) [69] is publicly available at https://gitlab.com/warplab/ros/sunshine.

characterized by different levels of observability, which bound how well any algorithm can perform [74], [75]. For example, the dynamic pricing problem models how adjusting the price of a product changes profits when different customers are willing to pay different prices [76]. The lack of local observability in dynamic pricing makes some instances of this problem fundamentally harder than bandit problems [77].

However, most real-world problems do not fit perfectly into archetypal online learning problem (OLP) structures or the assumptions of common online learning algorithms. For example, the objective of identifying the best treatment for an ailment from a range of options is well modelled as best-arm identification, but this approach is detrimental to the objective of exploiting effective dosages for the sample population [78]. Conversely, algorithms which focus only on effectively treating patients within a sample population may fail to reach sufficient statistical power to achieve the clinical trial's goal of supporting the superiority of any one specific treatment [71]. Precisely controlling the balance between such competing objectives is not a feature of previous online learning heuristics.

2.3.1 Online Learning Algorithms and Design Principles

Decades of research into online learning problems have generated a variety of widely adopted algorithm design principles. For example, "optimism in the face of uncertainty" [79], [80] is the driving principle behind the popular and well-studied family of Upper Confidence Bound algorithms, which achieve optimal asymptotic performance bounds in many OLPs [81]–[83]. Other design principles, such as "maximize the expected improvement" [84], [85] and "follow the knowledge gradient" [17] have similarly led to the development of online learning algorithms with strong guarantees and empirical results. In particular, many of these algorithms are "no-regret", in that the regret between of the policy grows sublinearly, so the average regret over some horizon decays asymptotically to zero. Information-directed sampling (IDS) is perhaps the best of the state-of-the-art strategies for long-/infinite-horizon multi-armed bandits, and introduces novel information-theoretic techniques with strong finite-time regret bounds [19].

The most similar online learning algorithm to the proposed *Endogenous Bayesian Risk Minimization* (EBRM) strategy presented in Chapter 5 is likely the Knowledge Gradient (KG) algorithm, which greedily chooses informative actions in order to increase the expected performance of a simple "stop-learning" policy [17]. As such, the Greedy-EBRM approach can be viewed as a generalization of the KG approach to incorporate additional problem complexities and constraints. Furthermore, we extend KG principles beyond multi-armed bandits, and produce similar results to that of knowledge-gradient optimality for monotone

submodular value-of-information functions [86], but for broader classes of OLPs. Strategies for best-arm identification include Top-Two Thompson Sampling [87], and the original KG exploration algorithm [88], [89]. The current state-of-the-art is the Top-Two Expected Improvement algorithm [85], which builds upon the ideas in these strategies and provides improved performance and regret bounds.

2.3.2 Markov Decision Processes and Partial Observability

Markov Decision Processes (MDPs) [90] are a highly flexible framework that can model a much broader range of decision problems than online learning problems. In particular, *partially observable* Markov decision processes (POMDPs) [91] can describe any problem in which there is some hidden *state* (set of parameters), which may change, and the consequences of actions taken by an agent depend on the value of that state. The agent's goal is to collect *reward* (a performance metric), but the amount of reward collected depends on the state; like in OLPs, the agent must generally trade-off between taking actions that help to reveal the hidden state (explore), or actions that collect reward (exploit) [92], [93]. The solution of a (PO)MDP is a *policy* (strategy for choosing actions), often found with *reinforcement learning* [94], [95].

The "Markovian" property of POMDPs require that the effects of an action depend only on the current state, but the model is otherwise able to capture many kinds of complexities, such as competing reward objectives, action costs or constraints, and complex state transition dynamics. POMDP models have been used to develop solutions for problems as diverse as railroad maintenance planning [96], unmanned aerial vehicle contingency management [97], recommendation systems [98], and treatment planning for sepsis patients [99]. However, many practitioners instead use online learning heuristic algorithms that ignore these complexities. A major factor in this decision is that specifying the components of a POMDP and computing its optimal solution tends to be tedious and computationally expensive, if not entirely infeasible [72].

Bayes-Adaptive MDPs (BAMDPs) [100] are a subclass of POMDPs which describe problems in which a fully observable state is separate from a stationary (fixed) hidden state. In this work, we consider all problems which can be represented by BAMDPs, with the additional constraints that the state transition following an action is deterministic given only the *observable* state, while the (possibly stochastic) reward of an action depends only on the *hidden* state. We formulate stochastic online learning as a fully-observable belief-space MDP (BMDP) [91], where "belief states" represent the BAMDP observable state combined with a probability distribution over possible hidden states. As the hidden state is stationary,

an agent navigating a BMDP begins with a belief state which broadly distributes probability across many possible values of the hidden parameters, but moves towards one which concentrates probability on the most likely values.

2.3.3 Risk Quantification, Decomposition, and Minimization

The *risk* of a policy describes the degree to which the performance of that policy may exceeded by some other policy; even if a policy may produce good results *in expectation*, it may still carry substantial risk. Most autonomous agents use risk-neutral decision making, which only considers the expected performance of a policy. However, this can lead to undesirable behaviours, particularly in social or safety-critical contexts. Risk quantification is essential to developing risk-*aware* and risk-*averse* (or risk-sensitive) agents [101]–[103]. A risk-aware agent is capable of, for example, reporting to the user when it is in a high-risk state (i.e., a situation for which there is no low risk policy) [104]. A risk-averse agent goes further by actively avoiding high risk states even if they would, in expectation, lead to better outcomes [102], [103].

It is often useful to separate a policy's risk into *aleatoric* and *epistemic* risks [105]–[107]. The aleatoric component describes how much a strategy's results depend on random processes in the environment, while the epistemic component describes how much risk could be eliminated through better knowledge of the environment [107], [108].[2] Actions which reveal information on the hidden state of the POMDP reduce epistemic risk. Conversely, aleatoric risk cannot be avoided in OLPs, as it is constant for a fixed hidden state.

In this thesis, we consider the Bayesian online learning setting where the hidden state is modelled as being drawn from a "prior" distribution; the *Bayes' risk* of a policy is the expected risk for a random hidden state distributed according to this prior. This setting is explored in Bayesian reinforcement learning [94], [109], [110], and techniques have been developed to learn risk-aware and risk-averse policies for general BAMDPs [93], [111]. However, such approaches require training a policy by running many episodic simulations of the problem. We propose EBRM as a risk-aware solution which, like the online learning heuristics discussed in Subsection 2.3.1, does not require any training, thus making it far more convenient to use.

[2] Intuitively, aleatoric risk is due to the inherent unpredictability of "dice rolls", while epistemic risk is due to uncertainty in whether the "dice" is loaded (and how).

2.3.4 Online Reward Learning in Autonomous Exploration

In our formulation of autonomous exploration, the "hidden state" presented in the previous section represents the true concentration of targets of interest within each semantic class in the area of operations. We assume that these targets are distributed at random throughout the environment with likelihood proportional to the these concentrations; these locations are called the "hidden outcomes". As the semantic maps produced by semantic mapping systems tend to assign each location to a distribution over semantic classes, we will assume that the probability of a target of interest being found at a particular location is a linear mixture of the concentrations parameters based on this semantic class distribution.

The robot's belief state contains a distribution over these possible concentration values, as well as distributions over all other variables relevant to planning, such as its own location. We call the belief distribution over the these concentration the robot's "reward model", so the online learning process is called *online reward learning*.

The agent's "policy" refers to a robotic explorer's *planner*, which produces a trajectory that the robot expects will collect as much reward as possible. Epistemic risk then describes the sub-optimality of the robot's planned trajectories that is due to incomplete knowledge of these concentration values (the reward model parameters, or "hidden state"). Conversely, the aleatoric risk measures how the random distribution of target *locations* could cause the robot to fail to find many targets along its trajectory, even if it were constructed with perfect knowledge of the true concentration parameters.

Chapter 3

Online Matching of Learned Semantic Representations

As discussed in Chapter 2, an important aspect of scientific exploration is that it often involves large and unknown environments which would be very time consuming to cover with a single robot. Furthermore, many marine environments, such as coral reefs and hydrothermal vent chains, are highly dynamic and thus require both broad and frequent exploration to monitor for changes. These issues necessitate using a team of robots working in parallel for *distributed* mapping and exploration.

When learning semantic representations online using unsupervised algorithms, the learned models are egocentric. Fig. 3.1 demonstrates how this can be an issue: one robot has learned a different permutation of the same semantic labels (represented with colors) learned by the other robot. In addition to the unknown permutation between corresponding labels, some labels learned by one robot may not correspond to any label observed by another robot, or a single label may represent the union of multiple labels learned by another robot.

Creating unified semantic maps and models is essential for fleet scalability. In particular, unifying the semantic models of individual robots enables sharing both semantic map information as well as reward models, which map the concentrations of targets of interest to semantic classes. Therefore, multi-robot distributed exploration with learned semantic models requires correctly estimating the total number of distinct semantic classes across all robots, and associating and fusing those that correspond to the same real-world domains (habitats or terrain types).

This chapter presents a novel system for multi-robot distributed semantic mapping that addresses the previously described issues. Each robot uses an online semantic 3D mapping system to model its own observations and create a high quality semantic map. The robots explore the target environment in parallel, sharing their learned semantic maps and models

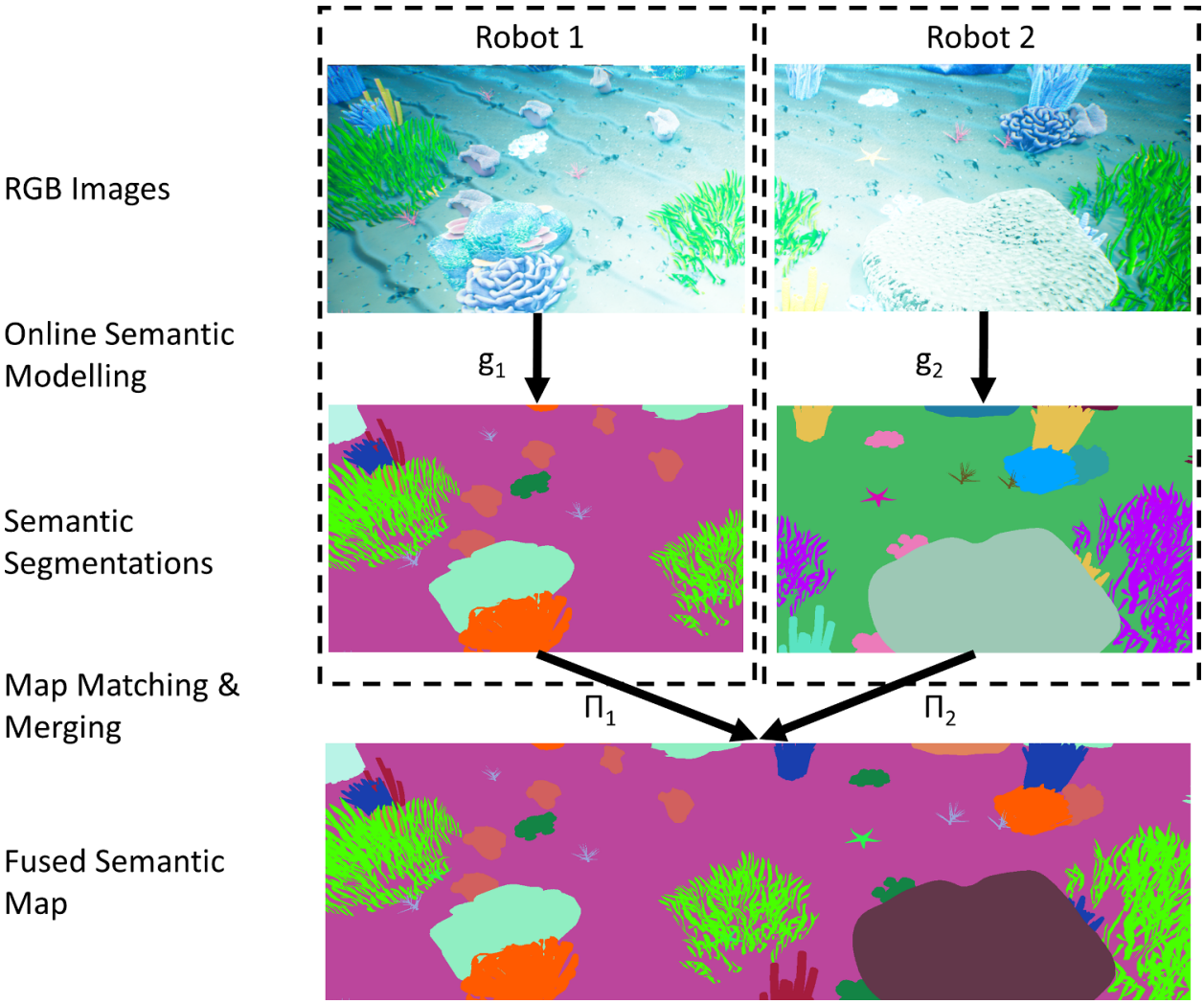


Figure 3.1: When robots develop different individual semantic models g_i, they solve the correspondences Π_i into a global (shared) semantic language in order to fuse their results. Images are from environment #2 (see Fig. 3.4).

with each other and with the human operator whenever communication constraints permit. Finally, a multiway matching algorithm, which can run on any robot, estimates the total number of unique phenomena observed across the robot team, finds matches between the same phenomena labeled differently by various robots, and fuses the local maps to obtain a consistent global map across all robots.

3.1 Related Works

Doherty et al. [24] showed that repeatedly matching the topic models of two robots with the Hungarian algorithm [112], merging them together, and distributing the merged model would result in both robots converging to a single set of good semantic labels, even if this was done at a low frequency. To our knowledge, [24] is the only prior work that has explored multi-robot distributed semantic mapping with representations learned online. The present work improves upon [24] with a novel solution for matching many ($N \gg 2$) semantic maps that is more robust to major variations across what each robot sees, and does not rely on distributing the merged topic model throughout the entire robot team. Eliminating the need to distribute the merged model halves communication bandwidth usage, and makes the approach more robust to transient connection failures.

Multiway matching algorithms are a class of data association techniques that leverage the transitivity property (cycle consistency) to rectify wrong correspondences and construct a unified representation from the partial and noisy observations of multiple agents. Multiway matching leads to superior accuracy compared to classical pairwise approaches (e.g., [112]), however it has combinatorial complexity. State-of-the-art methods consider approximations of this problem via convex relaxations [113]–[116], spectral relaxations [117], [118], or graph clustering [119] to obtain a solution in polynomial time. In this work we use CLEAR [26], a spectral clustering based approach, to match topic models because it is one of the most efficient multiway matching algorithms with leading performance in both precision and recall.

3.2 Problem Setup

Let us denote the environment to be mapped as $E \subset \mathbb{R}^3$. We partition E into a grid of disjoint cells (boxes) $\mathcal{B} = \{b_i\}$ such that $E = \cup_i b_i$. An oracle (e.g., the human operator) could assign to each box a distribution over human-defined semantic labels $\mathcal{Z}^H$ that represent its semantic contents. We assume that if these boxes are sufficiently small, each box can be effectively represented by a single dominant label. We thus define the ground truth semantic

segmentation $f : \mathcal{B} \to \mathcal{Z}^H$, where $f(b) \in \mathcal{Z}^H$ represents the human label for $b \in \mathcal{B}$. The model f and set $\mathcal{Z}^H$ cannot be provided to the robots in advance because the operator cannot predict everything that will be found during the mission (i.e., the set $\mathcal{Z}^H$ is unknown *a priori*), and communication bandwidth limitations prevent the operator from seeing most of the observations until the mission is over and the robots are recovered. The goal of the robot team is to construct a fused semantic map $g : \mathcal{B} \to \mathcal{Z}^G$, where $\mathcal{Z}^G$ is a shared set of learned semantic labels, such that $g(b)$ is "similar" to $f(b)$. A metric to evaluate this similarity will be presented in Section 3.4.

We assume that the team consists of N autonomous robots. By timestep t, the n^{th} robot has collected its own set of localized image observations and used them to build, in an unsupervised manner, a local semantic map $g_{n,t} : \mathcal{B} \to \mathcal{Z}_t^n$, where $\mathcal{Z}_t^n$ is the set of semantic labels the robot has developed to describe its own observations.[1] Due to the egocentricity of unsupervised semantic mapping, robots which observe different phenomena, or the same but in a different order, will almost certainly develop disparate semantic models, as in Fig. 3.1. In order to construct a fused semantic map, these N unique semantic models must first be fused to use a common set of labels. Therefore, the team must construct a set of global semantic labels $\mathcal{Z}_t^G$ and a set of correspondences $\Pi_t = \{\Pi_{n,t} : \mathcal{Z}_t^n \to \mathcal{Z}_t^G\}_{n=1}^N$ that translate individual robots' labels $\mathcal{Z}_t^n$ into $\mathcal{Z}_t^G$. Given Π_t, the individual semantic maps can be fused into a single global map $g_t : \mathcal{B} \to \mathcal{Z}_t^G$ for time t (see Fig. 3.1). Each label can be computed as $g_t(b) = \Pi_{n_b^\star,t}(g_{n_b^\star,t}(b))$ where $n_b^\star$ is the index of the robot that most recently visited and observed the cell $b \in \mathcal{B}$.

3.3 Consistent Online Topic Matching

We present an algorithm to produce a fused semantic map from semantic maps built by robots with disparate semantic models. *Consistency* ensures that the fused model created by any agent is the same across all agents, and *online* indicates that our algorithm will be run during the mission whenever a new matching is required. Typically, we expect a human operator would run our algorithm at a central node whenever they require an updated global map, however it is computationally lightweight enough that it may be run by any robot that has collected other robots' semantic maps.

The stages of the proposed system are presented graphically in Fig. 3.2 and in detail by the following subsections.

[1] In practice, we assume $g_{n,t}$ is learned from RGB-D image observations $\{o_{n,\tau}\}_{\tau=1}^t$ paired with estimated camera poses $\{x_{n,\tau} \in \mathtt{SE}(3)\}_{\tau=1}^t$.

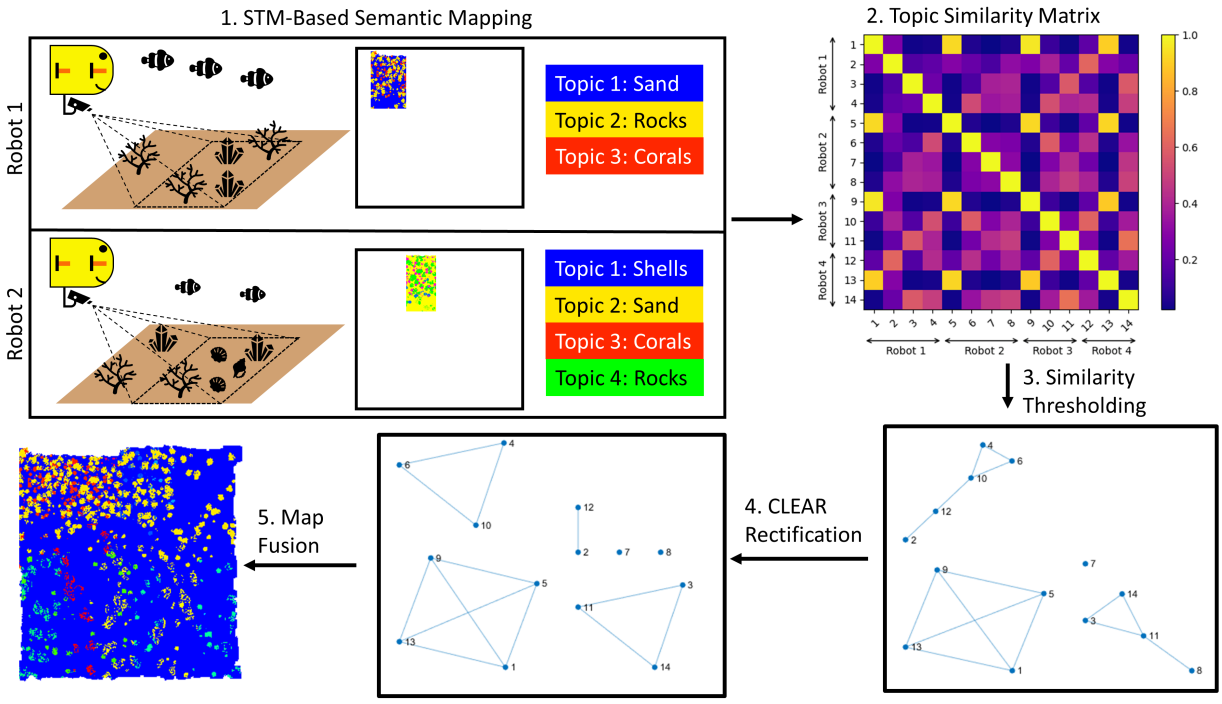


Figure 3.2: Our proposed system is composed of five stages. Each robot first learns an individual semantic model online which it uses to describe its own observations. When a fused map is required, a topic similarity matrix is constructed by using a similarity metric for topic descriptors to compute all pairwise similarity scores. A noisy association graph is produced by treating each topic as a vertex and using similarities above a specified threshold as edges. The noisy graph is rectified using the CLEAR multiway matching algorithm to produce a consistent topic matching, which has the form of a cluster graph. These topic matches are used to fuse the individual maps into one consistently labelled global map.

3.3.1 Online STM-Based Semantic Mapping

The proposed approach has each robot construct a spatiotemporal topic model online as the basis for its individual semantic map, as in [11]. Nonetheless, it would be straightforward to adapt this system to use any similar unsupervised online semantic mapping module.

While an exhaustive description of the BNP-ROST model used is left to [9], a few details are relevant here. First, the model requires no pre-training, although it can be bootstrapped with topics for common phenomena. Second, the model is tuned by varying the feature vocabulary, the spatiotemporal grid cell size, and three scalar hyperparameters. Typically, the vocabulary is domain-specific while the cell size and the scalar hyperparameters are tuned for a specific mission.[2] All robots on the same mission use the same hyperparameters. Finally, each robot's topic model uses a stochastic process to develop an ever-evolving set of

[2]While they may be tuned with a dataset representative of the target environment, these hyperparameters encode only abstract information about the domain and thus tend to generalize well to novel environments.

topics online based on its own visual observations. At time t the n^{th} robot has $K_{n,t}$ topics, so $\mathcal{Z}_t^n = \{1, \ldots, K_{n,t}\}$. Each topic $z_k \in \mathcal{Z}_t^n$ is characterized by a semantic "descriptor" $\phi_k \in \Delta^V$, a distribution over "words" in the predefined vocabulary of size V [65], where $\Delta^V = \{p \in \mathbb{R}_+^V : \|p\|_1 = 1\}$. Each grid cell b_i of the environment E is labelled with a single maximum likelihood topic, which may change as the model evolves; thus, we will start using the terms "topic" and "label" interchangeably.

3.3.2 Computing Topic Similarity

We require a similarity metric that measures how similar two topics are to each other in order to identify when multiple robots have developed any semantically equivalent topics. Since each descriptor ϕ_k represents a probability mass function, it is natural to consider similarity metrics that operate on discrete probability distributions. Total Variation Distance (TVD) [120] measures the largest possible difference in probability that two topics assign to the same set of words. Thus, the Topic Overlap (TO),

$$\text{TO}(\phi_1, \phi_2) = 1 - \text{TVD}(\phi_1, \phi_2), \tag{3.3.1}$$

is a similarity metric that represents the total probability mass which both ϕ_1 and ϕ_2 assign similarly. It can be computed using the identity $\text{TVD}(\phi_1, \phi_2) = \frac{1}{2}\|\phi_1 - \phi_2\|_1$.

Another metric commonly used for comparing topic descriptors is Cosine Similarity (CS) [121], which computes the cosine of the angle between two descriptors as

$$\text{CS}(\phi_1, \phi_2) = \frac{\phi_1 \cdot \phi_2}{\|\phi_1\|\|\phi_2\|}. \tag{3.3.2}$$

The CS metric is about as efficient to compute as TO, but assigns a higher score when ϕ_1 and ϕ_2 are very similar and a lower score when they are very dissimilar. This may be preferable because, as the stochastic nature of topic models means that the descriptors $\{\phi_k\}$ fluctuate constantly, two topics with the same semantic meaning are likely to have slightly different descriptors at any given time.

Each similarity metric presented above is symmetric and bounded by $[0, 1]$, where a score of 0.0 indicates two topics have no words in common, and 1.0 indicates that two topics are exactly the same. We use the chosen similarity metric s to construct the *pairwise similarity graph*, a weighted and undirected graph in which vertices are topics and edge weights are the similarity of the adjoining vertices; it is represented in Fig. 3.2 by the topic similarity matrix.

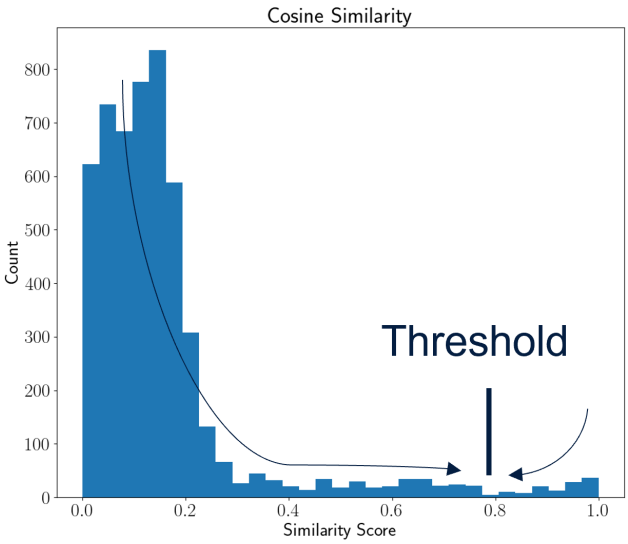


Figure 3.3: This histogram shows the pairwise similarity scores across the 72 topics learned by a fleet of 12 robots, computed using cosine similarity. The annotations highlighting the shape of the histogram suggest an automated method of determining the similarity threshold, which is discussed further in the main text.

3.3.3 Constructing the Noisy Association Graph

In practice, similarity metrics are "noisy" in that topics which a human would judge to have the same semantic meaning may not have a similarity score of 1.0, and topics with very different semantic meanings may not have a similarity of 0.0. The pairwise similarity graph is simplified by removing edges with weights below some $\sigma \in (0, 1)$ that represents a sufficient level of similarity, and setting weights above σ to 1, resulting in the unweighted *noisy association graph*.

In general, a good threshold σ for considering two topics to be "sufficiently similar" will depend on factors including the similarity metric s, the topic model hyperparameters, and the subjective opinion of the human operator. It is difficult to choose σ analytically because the expected topic growth rate and average inter-topic similarity are complicated functions of the topic model hyperparameters. A simple solution for choosing σ is to collect a validation set of topics developed by robots in past missions, for which the human operator can infer their semantic meanings, and then tune σ low enough that the algorithm merges as many topics with the equivalent meanings as possible but high enough that it does not match distinct topics. In the training dataset used to choose the topic model hyperparameters,[3] this method was used to establish a similarity threshold of $\sigma = 0.75$, which was found to be suitable for use with both topic similarity metrics.

[3]This dataset and process will be described in Section 3.4.

While straightforward, the method for choosing σ presented above requires human analysis and is subject to personal biases (the inferred semantic meaning of various topics). Figure 3.3 depicts a histogram of the similarity scores across a large set of matching and non-matching topics, and suggests an automated method of determining a sufficient similarity threshold. We see in the figure that the similarity scores tend to be bimodal, with peaks near 0 (for highly dissimilar topics) and 1 (for matching topics). By modeling the distributions of similarity scores for matching and non-matching topics, respectively, using two unimodal distributions (e.g., beta distributions) with corresponding peaks, σ could be computed as the value at which it is equally likely the score was generated from either distribution. In this example, it appears as though the non-matching distribution has support between 0 and just below 0.8, while the matching distribution has support from 0.8 and up, so a suitable threshold would be just below 0.8; this agrees well with the manually determined threshold of 0.75.

Regardless of the method used to determine the threshold σ, it may not be obvious from the resulting noisy association graph how many unique topics should be used in the final map, or which sets of topics should be matched. This is because noisy associations can violate *transitivity*: if we match topics 1 and 2, as well as 2 and 3, then we expect topic 1 should have the same semantic meaning as topic 3, but for most similarity functions f (including topic overlap and cosine similarity), $f(\phi_1, \phi_2) \geq \sigma \wedge f(\phi_2, \phi_3) \geq \sigma \implies f(\phi_1, \phi_3) \geq \sigma$. This situation raises the question of whether topic 2 should be assigned the same label as topic 1, topic 3, both, or neither; different answers can further imply different numbers of labels in the global map. The next subsection will address these concerns by finding the "closest" association graph which contains only consistent matches.

3.3.4 Rectifying the Noisy Association Graph

A consistent association graph, without any noise, should have the structure of a *cluster graph*. A cluster graph is a set of disjoint fully-connected components, such that any two nodes (topics) in the same component are matched and no nodes (topics) are matched between components. The number of distinct topic labels developed by the entire robot team should then match the number of disjoint components.

CLEAR [26] is a spectral clustering algorithm that estimates the *closest* cluster graph to a noisy association graph (see Fig. 3.2), in terms of the number of edges added or removed. A key reason for choosing CLEAR is that it is one of the fastest algorithms to perform multiway matching with high accuracy. The Laplacian L of the noisy association graph is a matrix defined in terms of the graph adjacency matrix A and degree matrix D as $L = D - A$,

where

$$[A]_{ij} = \begin{cases} 1, & s(\phi_i, \phi_j) \geq \sigma \\ 0, & \text{otherwise} \end{cases} \tag{3.3.3}$$

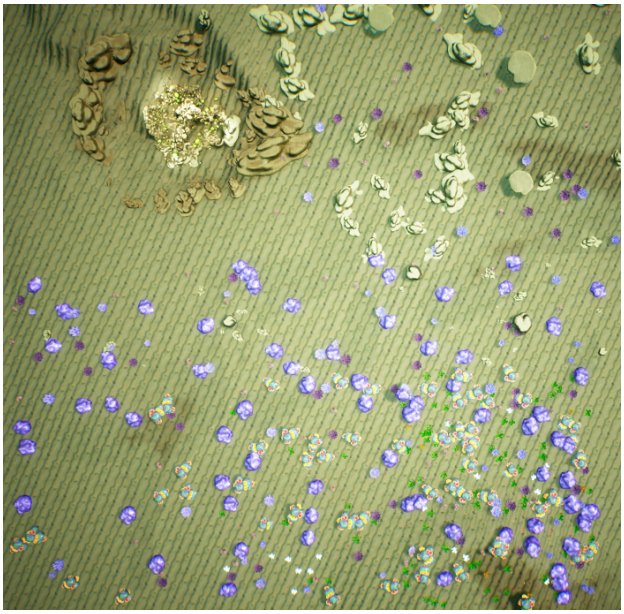
$$[D]_{ij} = \begin{cases} \sum_{k=1}^{N} [A]_{ik}, & i = j \\ 0, & \text{otherwise} \end{cases} \tag{3.3.4}$$

CLEAR uses a special normalization of the Laplacian based on the degree matrix plus identity, denoted by L_{nrm}, to identify clusters of semantic labels in the noisy association graph with high pairwise similarity. The number of eigenvalues of L_{nrm} less than 0.5 is a robust estimate of the number of global labels, $|\mathcal{Z}_t^G|$. CLEAR then uses the eigenvectors of L_{nrm} to find a consistent set of label correspondences Π_t.

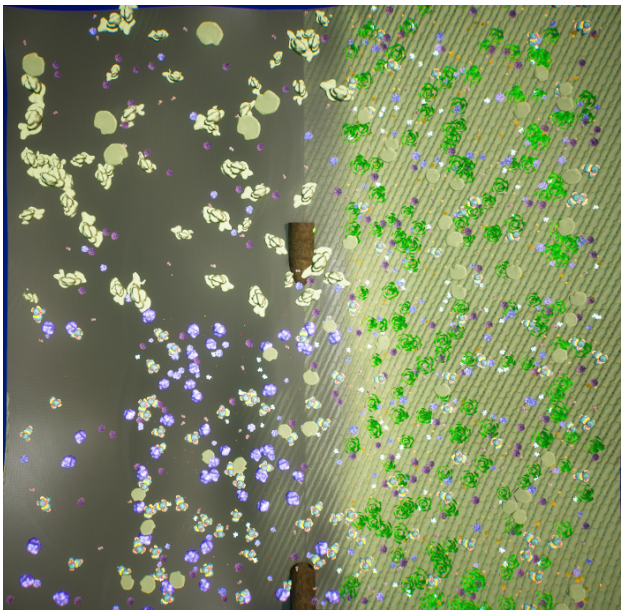
3.4 Experimental Methodology

The proposed consistent online topic matching system was evaluated using semantic mapping experiments in two unique high-resolution 3D simulated coral reef environments produced in the Unreal Engine [122] using the Automatic Coral Generator package [123]. A top-down view of each simulated environment is presented in Fig. 3.4. In each experiment, a team of 12 simulated robots traversed one of the two environments and each collected 250 RGB-D observations using the AirSim plugin [124]. AirSim also provided a ground truth semantic segmentation for each image. Each robot was given noiseless localization information and ran a new release of the BNP-ROST [9] spatiotemporal topic model called "Sunshine", which is conceptually identical to the system presented in [11] but redesigned for ease of use and with optimized code to produce higher quality maps with less processing power. Throughout the experiment, sets of 1 to 12 robots' local maps were randomly chosen and fused together using the approach described in Section 3.3. Each experiment was repeated 24 times to control for between-run variation in the topic models produced by each robot.

The topic model hyperparameters were set using a Bayesian Optimization algorithm [125] to find the values that resulted in the highest map quality, evaluated using a third simulated reef environment which was similar to Environment #1. The topic model vocabulary was the same one used in [11]. All code, instructions, hyperparameters, and datasets required to reproduce these experiments, as well as instructions to generate similar test environments, are available in the Sunshine repository.

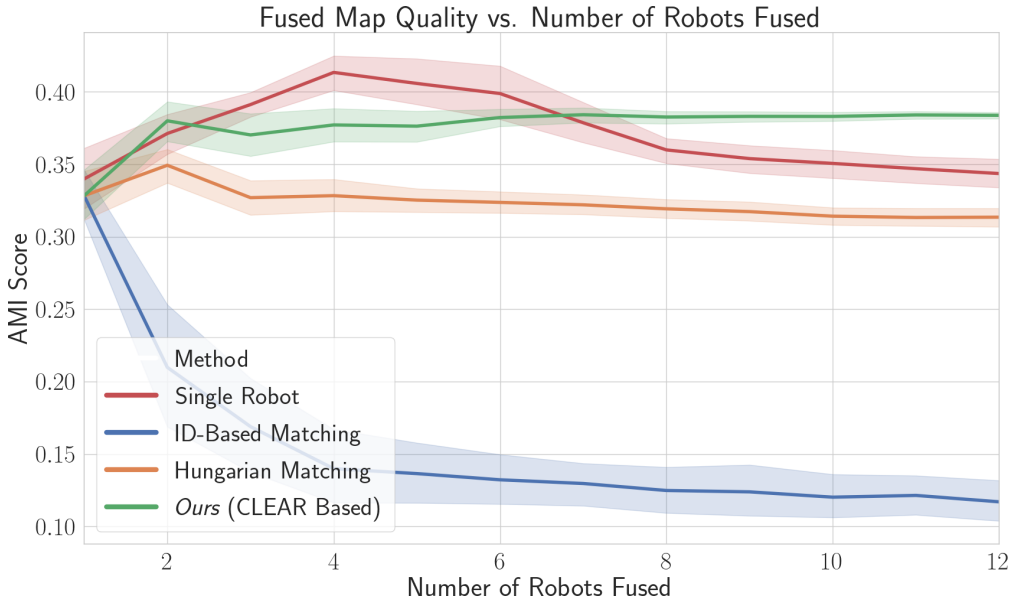


(a) Environment # 1.

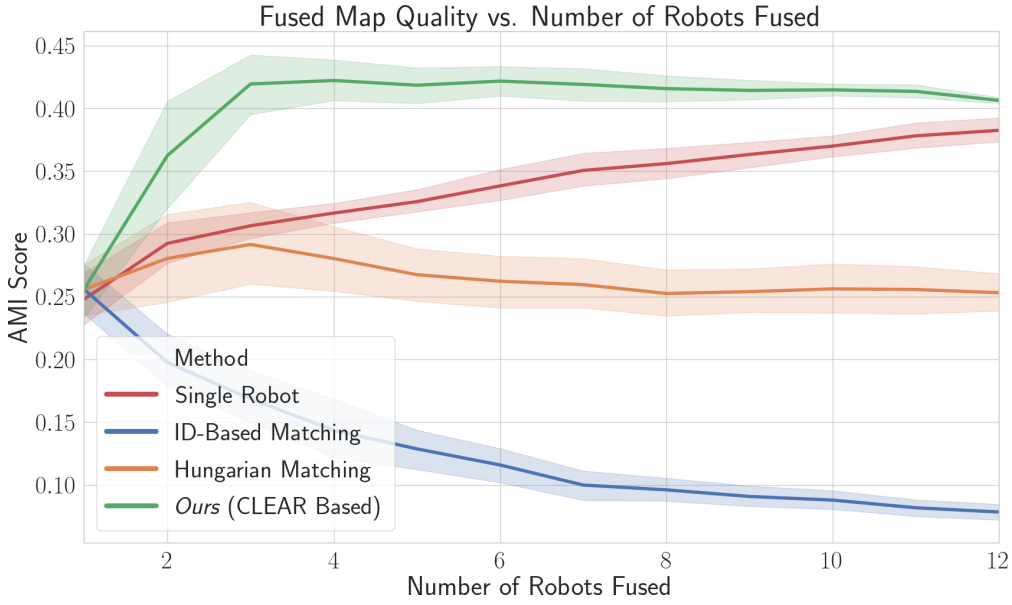


(b) Environment # 2.

Figure 3.4: Top-down views of the two simulated test environments used in the experiments. Each map is approximately 250m×250m, and contains a rich variety of coral species, seaweed, and rocks.



(a) In environment #1, sparser and less varied phenomena (coral species) were spread throughout a uniformly sandy reef. The prevalence of sand in both shaded and well lit conditions tended to cause the single robot to develop two sand topics, reducing its performance.



(b) In environment #2, more variation between robots in the phenomena (coral species) and terrains observed meant that the Hungarian algorithm's assumptions were violated, leading to reduced performance compared to the proposed CLEAR-based approach.

Figure 3.5: The AMI scores between the fused maps and corresponding ground truth maps extracted from the simulator demonstrate that the multi-robot distributed mapping system with CLEAR-based label matching outperforms all other multi-robot approaches. Error bars represent the 95% confidence interval of the mean score. The score of a single-robot that explored the same total area as the corresponding fused maps is shown in red, for comparison.

3.4.1 Evaluating Semantic Map Quality

A useful fused semantic map $g(x)$ is one that is, at every location x, a *good predictor of* the ground truth label $f(x)$ defined by the human operator. We measure this predictive strength using Adjusted Mutual Information (AMI) [126], a normalized variant of Mutual Information (MI). MI represents the number of bits of information contained in one random variable that describe another, and is defined for discrete random variables $U \in \mathcal{U}, V \in \mathcal{V}$ as

$$\mathrm{MI}(U,V) = \sum_{u\in\mathcal{U}}\sum_{v\in\mathcal{V}} p(u,v)\log\left(\frac{p(u)p(v)}{p(u,v)}\right). \tag{3.4.1}$$

Given the semantic maps f and g and a finite set of cells $\mathcal{B}$, we define the random variable B as a cell chosen uniformly at random from $\mathcal{B}$, and thus define the random variables $Y_f = f(B)$ and $Y_g = g(B)$ as the robot team's semantic label and the ground truth semantic label for B, respectively. The joint probability of these random variables is:

$$p(y_f, y_g) = \frac{1}{|\mathcal{B}|}\sum_{b\in\mathcal{B}} \mathbb{1}_{f(b)=y_f \wedge g(b)=y_g}, \tag{3.4.2}$$

which gives the probabilities $p(y_f), p(y_g)$ through marginalization. Denoting the entropy of random variables Y_f and Y_g as $H(Y_f)$ and $H(Y_g)$, the AMI is computed as

$$\mathrm{AMI}(Y_f, Y_g) = \frac{\mathrm{MI}(Y_f, Y_g) - \mathbb{E}[\mathrm{MI}(Y_f, Y_g)]}{\max\{H(Y_f), H(Y_g)\} - \mathbb{E}[\mathrm{MI}(Y_f, Y_g)]}. \tag{3.4.3}$$

A perfect AMI score of 1 indicates that the robot team's semantic map contains all information required to reproduce the ground truth semantic map of the same area. Conversely, a score of 0 indicates that the team's semantic map contains no more information about the corresponding part of the ground truth map than a randomly generated map is expected to. This is because the normalization subtracts out the expected mutual information $\mathbb{E}[\mathrm{MI}(Y_f, Y_g)]$ between the labelings Y_f and Y_g, computed according to [126]. If a fused semantic map has a high AMI score with respect to the ground truth, then there is a consistent correspondence between each semantic label used by the robot team and each label used to produce the ground truth map. Thus, for high AMI scores, the human operator only needs to look at a few example images for any label in the fused map in order to determine that label's human-interpretable meaning.

3.4.2 Baseline Comparisons

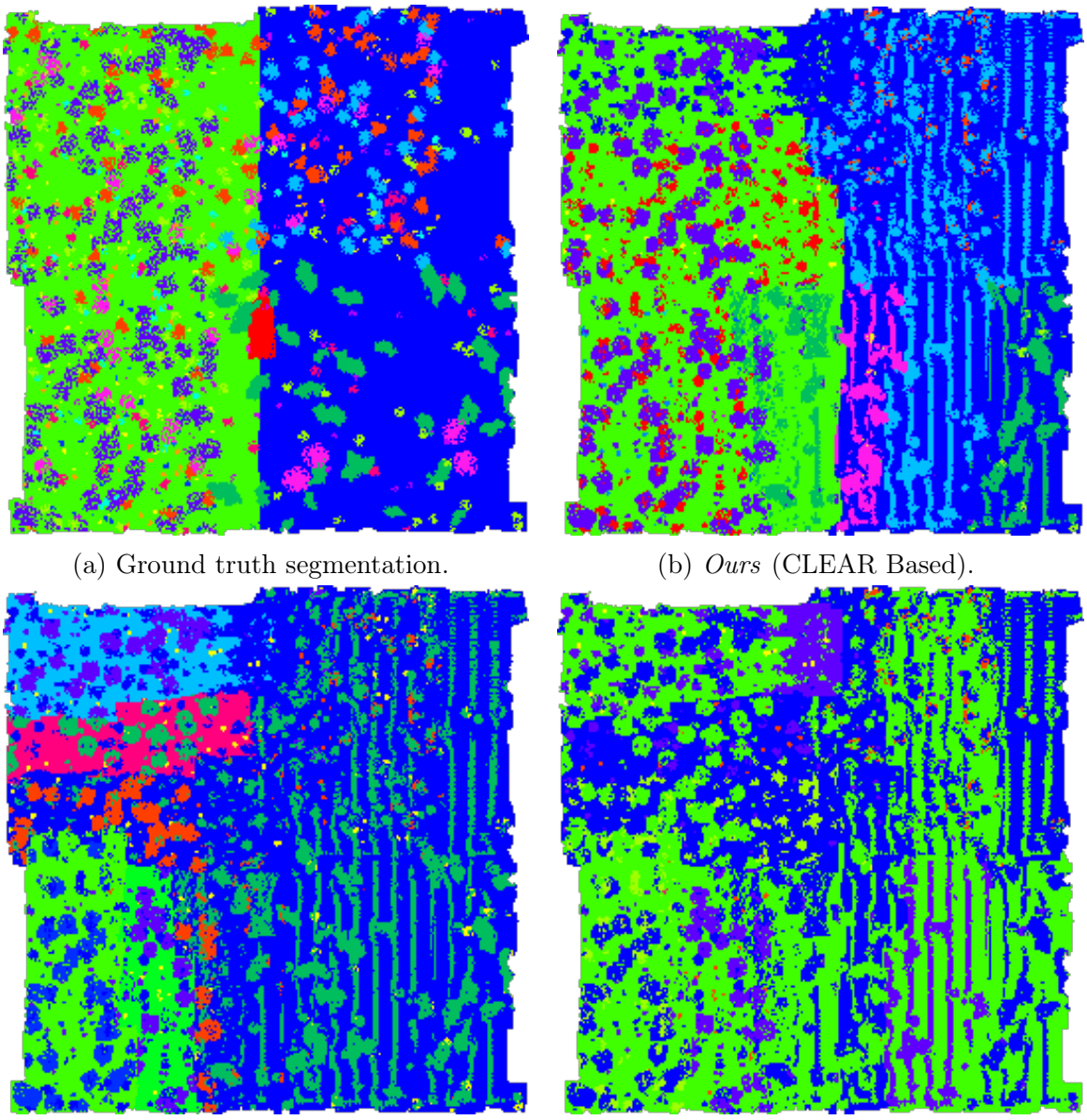
The local maps were also fused using the ID-based matching and Hungarian matching approaches described in [24] to get baseline performance metrics. ID-based matching assumes that every robot observed the same phenomena in the same order, and so the first topic learned by one robot corresponds to the first topic learned by every other robot, and likewise for the remaining topics. The Hungarian approach only assumes that every robot observed the same phenomena, and finds the maximum similarity permutation between the first robot's topics and each additional robot's. This is a *sequential*, not multiway, matching approach because it compares topics belonging to a pair of robots at a time instead of considering all of the topics together. We are not aware of any previous baselines that used a multiway matching algorithm or did not assume that every robot observed the same phenomena.

Separate from the multi-robot experiments, a single robot was used to explore the same environments and independently build its own semantic map. It used the same topic model hyperparameters as the robots in the multi-agent experiment, but did not require any topic matching or map fusion. The single robot required Nt seconds to explore the same area that N robots explored in t seconds; the quality of the map it produced after exploring the same area as the N fused robots is reported in the results as "Single Robot".

3.5 Results & Discussion

Figure 3.5 shows the performance of the proposed system using CLEAR, compared to using baseline matching solutions and to using a single robot. The performance was measured in terms of AMI of the fused map with the ground truth semantic map produced by AirSim. While the performance of the other matching algorithms declines as more robots are fused, the performance of the proposed matching solution *increases* or stays steady. This happens because CLEAR leverages redundant edges in the noisy association graph, added by additional robots, to help compensate for incorrect edges. The number of incorrect edges at each vertex grows slower than the number of correct edges as topics are added, so our system is able to find a better solution when more robots' maps are fused.

In the first test environment, Fig. 3.5a, the fused map quality of the proposed approach goes from about 10% lower than the semantic map produced by a single robot to about 10% higher as more robots are fused. This is excellent performance considering that the robot team was able to map the entire environment in $1/12^{\text{th}}$ the amount of time. Compared to the Hungarian matching approach, the proposed system achieves 23% higher AMI scores on



(a) Ground truth segmentation.

(b) *Ours* (CLEAR Based).

(c) Hungarian matching.

(d) ID-Based matching.

Figure 3.6: Sample fused maps from each multi-robot matching approach with all 12 robots, alongside the ground truth segmentation, for Environment #2. Note that each map has been manually colored (1 color per label) with the same palette to ease comparison. The proposed approach most accurately captures the variation in terrain and coral species present in each quadrant.

Table 3.1: Semantic Mapping Performance with 12 Robots.

Matching Alg.	Metric	Mean AMI Score (Std. Dev.) Env. #1	Env. #2
ID-Based	N/A	0.117 (0.035)	0.078 (0.016)
Hungarian	L1 Distance	0.297 (0.006)	0.216 (0.004)
	L2 Distance	0.313 (0.016)	0.253 (0.039)
	Cosine Distance	0.304 (0.011)	0.203 (0.015)
CLEAR	TO Similarity	0.250 (0.002)	0.341 (0.003)
	Cosine Similarity	**0.384** (0.006)	**0.406** (0.007)
Single Robot (No Matching)		0.344 (0.026)	0.382 (0.024)

average; as shown in Fig. 3.6, this is primarily because CLEAR is better suited to recognize when different robots have observed distinct phenomena. In the second test environment, Fig. 3.5b, this difference was magnified as there was very little in common between what any pair of robots observed. Table 3.1 summarizes numerical results for the map quality after fusing all 12 local maps together with each matching algorithm and for various similarity and distance metrics. The other figures shown used Cosine similarity for CLEAR matching and Euclidean (L2) distance as the Hungarian cost metric.

As seen in Fig. 3.7, as the team explores environment #2 the fused map quality is mostly constant after each robot has collected 125 images, i.e., covered half of its assigned area. This suggests that environment #2 would be most efficiently explored using 24 robots; in general, the optimal number will depend on the size and complexity of the environment.

The results presented are expected to generalize well to real world environments. Previous versions of BNP-ROST have demonstrated the ability to produce useful semantic maps of real-world environments while running on real robot hardware [11]. Furthermore, the compressed semantic maps and topic descriptors shared between robots are very small (typically around 10 to 100 kB) so they can be transmitted between robots in even severely bandwidth-constrained environments, like the deep sea [11], [127].

3.6 Summary

In this chapter we presented a novel multi-robot distributed semantic mapping system that produces accurate semantic maps even when fusing maps from *many* robots and when each robot is building its unsupervised semantic model online with *no pre-training*. The proposed topic matching approach results in 20-60% higher map quality than pairwise Hungarian matching, with the largest gains in mapping complex and diverse environments, while also

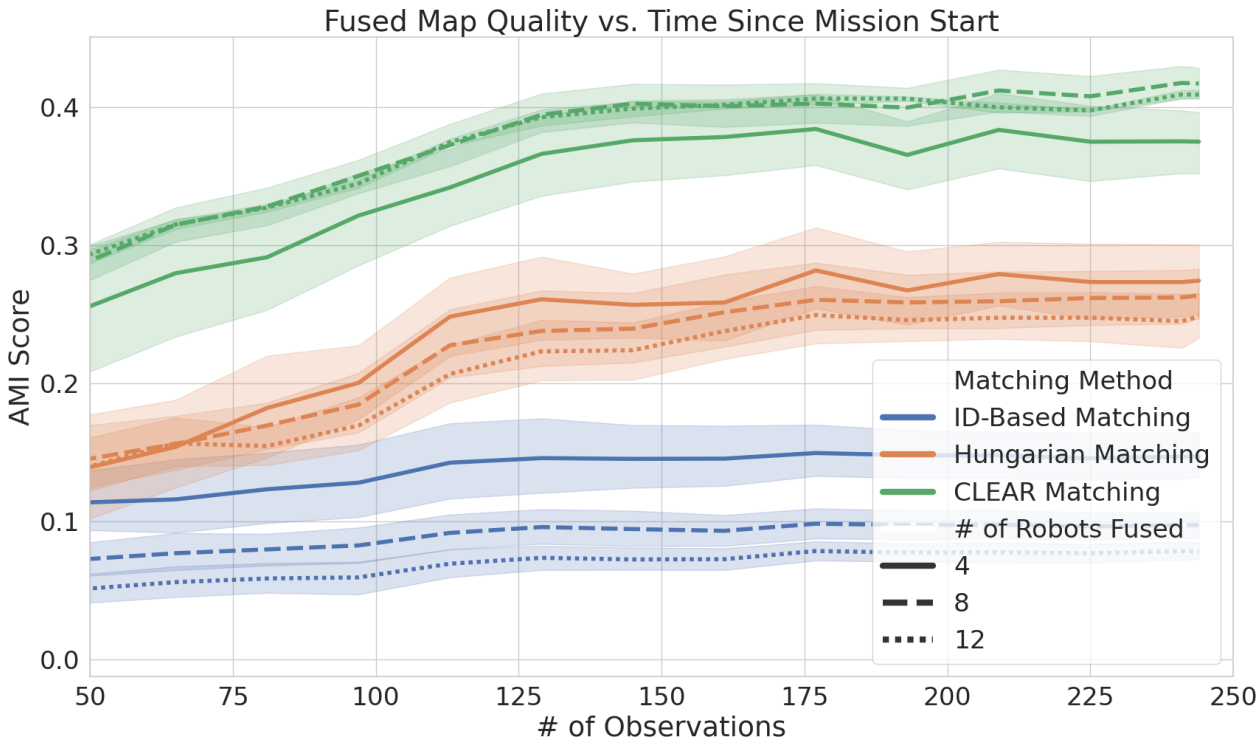


Figure 3.7: The fused map quality varies throughout each experiment; shown here is how the fused map quality (AMI) changes as the robots explore environment #2. The performance increase across all methods is caused by each robot's topic model improving over time.

using less communication bandwidth than the previous state-of-the-art [24]. The fused maps are suitable for the human operator to use for mission summarization and informative path planning. We find that the fused maps approximate the quality of the best single-robot maps, hence further performance increases will likely come from improving the STM-based online semantic mapping component. The presented system for accurate topic matching over low-bandwidth enables novel multi-robot distributed autonomous exploration capabilities, such as cooperative-adaptive path planning and distributed reward learning, which should be explored in future work.

Chapter 4

Realtime Adaptive Underwater Color Correction

The development of autonomous underwater vehicles (AUVs) has trended towards more complex and adaptive vision-based behaviours, enabled by the increasing performance and energy efficiency of onboard micro-computing resources over time. Many such behaviours, including visual target tracking [128]–[131], adaptive exploration of benthic environments [9], [13], [132]–[136], autonomous docking [137]–[139], and diver following [140] and assistance [141]–[144], strongly depend on an AUV's capability for visual perception and understanding. Unfortunately, underwater image quality is generally lower and more variable than the quality of similar images taken in air, making vision-based autonomous behaviours far more challenging to perform underwater than in other domains.

The most prominent image quality issues result from spectrally-selective light attenuation and backscattering [145]. These phenomena cause images taken underwater to be lacking in certain colors, typically red, and to have an excess of others, such as blue and green [145], as seen in Figure 4.1. The magnitude of these effects is correlated with the camera's water depth and range to the target; objects in the image that are shallow and close to the camera will have more of their natural color than objects that are deeper or further away. However, the specific color distortions in an image depend also on the presence of other light sources (e.g., the sun), and on the optical properties of the camera and the body of water in which the image was captured.

These issues are especially problematic for autonomous exploration with learned semantic models, as the semantic classes learned by the robot reflect the visual features it observes in the environment. When these features are corrupted by light attenuation and backscattering, a robot may assign different semantic classes to the same domain based on factors such as the time of day, the depth of the robot, and the current water conditions. This produces

inaccurate semantic maps that are more poorly correlated with targets of interest.

More broadly, the performance of computer vision algorithms for image classification and 3D reconstruction is severely degraded by color distortions, regardless of whether those algorithms are based on deep neural networks or on more traditional feature detection and matching methods [146], [147]. Even learning-based systems trained to be robust to targets within some range of distances, lighting conditions, and water masses will generally not be robust to real-world targets outside of the training distribution. This makes realtime, adaptive, *in situ* color correction a necessary prerequisite for successful vision-based AUV tasks and behaviours, especially in potentially safety-critical applications such as diver assistance. This is a gap in the underwater vision field; while previous works have focused on high-quality offline color reconstruction [148]–[153], to our knowledge no methods have been developed specifically for realtime applications.

This chapter presents *DeepSeeColor*, a novel method for realtime and adaptive color correction of images onboard an AUV. This realtime preprocessing enables more robust execution of *in situ* autonomous behaviours such as object recognition and tracking, semantic mapping, and mission summarization. *DeepSeeColor* improves upon the successful "Sea-Thru" algorithm [153] in terms of computational efficiency. It achieves this improvement through using highly efficient gradient-based optimization methods to learn the weights of two simple convolutional neural networks, where these weights correspond to a physics-based image formation model's unknown parameters. This novel approach replaces far more computationally expensive image processing operations, while leveraging GPU acceleration and other built-in performance optimizations provided by popular deep learning frameworks. We further present validation of DeepSeeColor on the dataset provided with [153], as well as on a new dataset of stereo-imagery collected from AUV deployments at coral reef sites in the US Virgin Islands.

4.1 Specific Background & Related Works

4.1.1 Underwater Image Formation

A digital camera system creates images by measuring the intensity of incoming light reflected from various targets in its lens' field of view. Assuming the light originates from a broadband source similar to the sun, the wavelengths of the reflected light accurately represent the colors of the targets. A color filter array on the camera's image sensor enables measuring

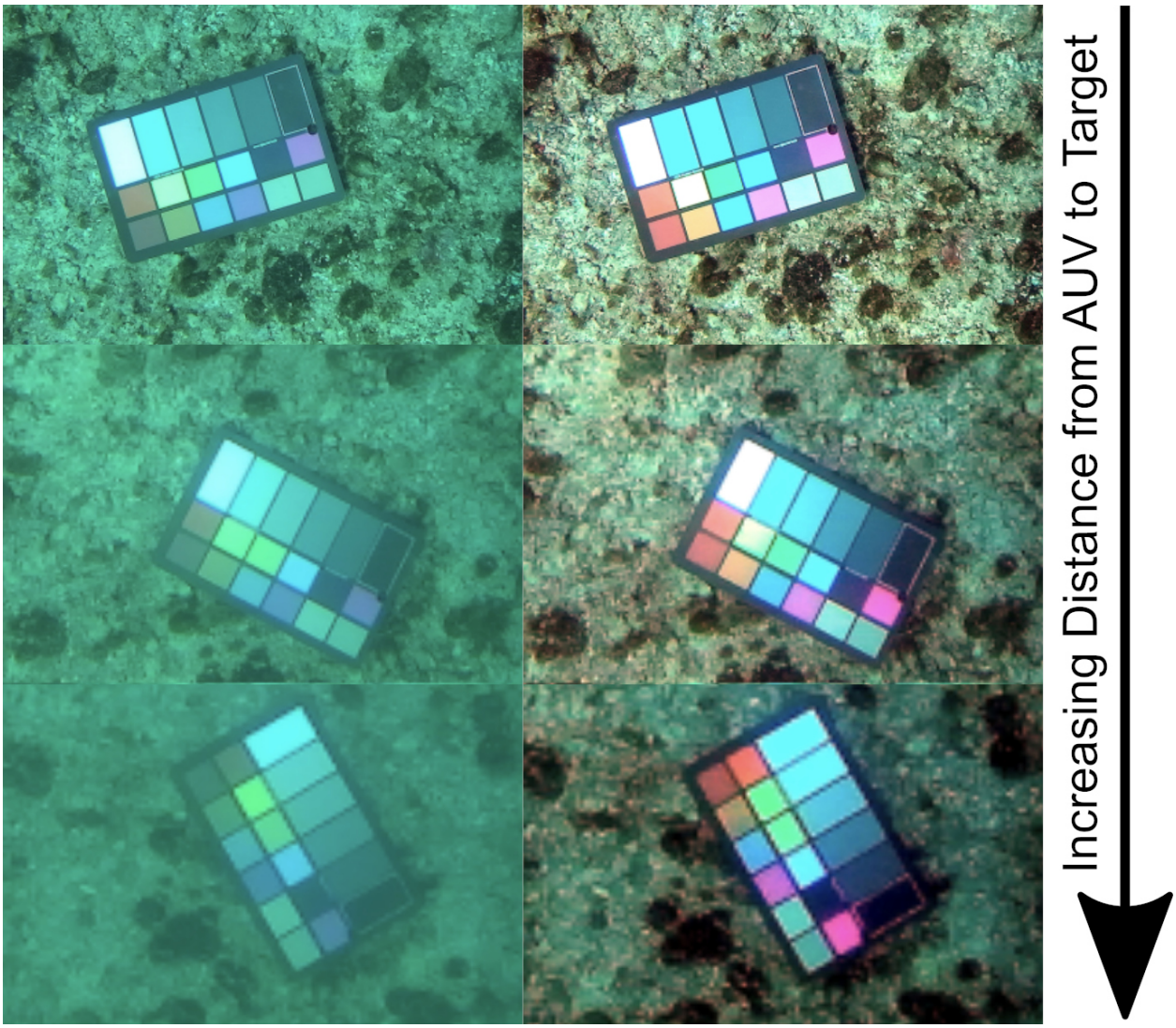


Figure 4.1: Best viewed on a screen, in color. Images taken in underwater environments (left column) suffer from severe loss of colors due to the combination of spectrally-selective light attenuation and backscattering due to particles in water. This effect is more pronounced as the distance between the camera and imaging target increases. This chapter presents DeepSeeColor, an efficient technique to reconstruct true-color images (right column) for use in realtime onboard decision making by an autonomous underwater vehicle.

the intensity of light within the red, green, and blue wavelength "channels"[1] at each pixel.

Regardless of wavelength, the apparent intensity of light decreases in relation to an observer's distance from its source according to the inverse-square law. However, the presence of an intervening medium further attenuates the apparent intensity of a target through *absorption* and *scattering*. Furthermore, many media are *spectrally-selective*: they preferentially attenuate some wavelengths of light over others. Water is much more spectrally-selective than air: red light is attenuated over a distance of 1m about as much as green light is over 10m, or blue light is over 100m [154].

Backscattering is the process by which particles in the water column between the target and camera reflect light from sources other than the target into the camera. This is particularly problematic at shallow depths (<100m), long target ranges (>1m), and in turbid waters.[2] At shallow depths, light from the surface (e.g., sunlight) is scattered in all directions by tiny particulate in the water; the spectrum of this scattered light consist of wavelengths that were not absorbed higher up in the water column, and thus tends to be blue or blue-green. The greater the range to the target, the more intervening particulate there is to backscatter this light into the camera; the resulting "haze" in the image makes it more difficult to see the target, especially when it saturates the camera's dynamic range.

We denote the intensity of light in channel $c \in \{\mathtt{R}, \mathtt{G}, \mathtt{B}\}$ emitted by the target along the ray to pixel (i, j) in the camera image sensor as $J_c(i, j)$, and the intensity measured by the sensor at that pixel as $I_c(i, j)$. Spectrally-selective light attenuation and backscattering can be modelled together using the underwater image formation model,

$$I_c(i, j) = J_c(i, j)A_c(i, j) + B_c(i, j), \tag{4.1.1}$$

where $A_c(i, j)$ and $B_c(i, j)$ represent wavelength-dependent light attenuation and backscatter, respectively. This reflects the form of models used by many works in underwater color reconstruction [148]–[152], which take A and B to be

$$A_c(i, j) = \exp(-a_c \cdot z_{i,j}), \tag{4.1.2}$$

$$B_c(i, j) = \gamma_c^\infty(1 - A_c(i, j)), \tag{4.1.3}$$

where $z_{i,j}$ is the range of the target. The values of $a_c, \gamma_c^\infty \in \mathbb{R}_{\geq 0}$ are determined by the camera system and environmental parameters, including as the water type, target reflectance,

[1] These channels are roughly defined as the wavelength ranges 575-725nm (red), 475-625nm (green), and 400-550nm (blue), respectively.

[2] Note "depth" refers to the vertical distance of the target from the water surface, while "range" refers to the distance from the camera to the target.

illumination sources, image sensor characteristics, and camera depth from the water surface, which are, for now, all assumed to be fixed.

More recently, [155] found that Eq. (4.1.2), derived from an atmospheric dehazing model, neglects the range-dependence of underwater light attenuation coefficient a_c, and incorrectly assumes that the coefficients governing the range-dependence of attenuation and backscattering are the same. [155] presented a new model capable of capturing these complexities,

$$A_c(i,j) = \exp(-a_c(z_{i,j}) \cdot z_{i,j}), \tag{4.1.4}$$

$$B_c(i,j) = \gamma_c^{\infty}(1 - \exp(-\beta_c z_{i,j})), \tag{4.1.5}$$

with scalars $\gamma_c^{\infty}, \beta_c \in \mathbb{R}_{\geq 0}$ and a parametric function $a_c(z)$.[3,4] This is the model used by Sea-Thru [153] and DeepSeeColor, and will be explored further in Section 4.2.

4.1.2 Methods for Color Reconstruction

Spectrally-selective light attenuation and backscattering have been considered by underwater color reconstruction algorithms since the seminal works of [148]–[150]. These works presented each presented a method to estimate their respective image formation model parameters from a single image. Given these parameter estimates, the true color of the target at pixel location (i,j) can be reconstructed as

$$J_c(i,j) = D_c(i,j)A_c(i,j)^{-1}, \tag{4.1.6}$$

where $D_c(i,j) \coloneqq I_c(i,j) - B_c(i,j)$ denotes the "direct" signal. The components of this approach to color reconstruction are depicted in Fig. 4.2. More recent works have explored techniques for better estimating the parameters a_c, γ_c^{∞} through leveraging additional information, such as the optical properties of known water profiles [156].

Among the most relevant of prior works is [152], which recognized that leveraging an accurate range map, which describes the distance of each pixel in an image, could better constrain estimates of the image formation model parameters. The method was designed for use on AUV collected data, and applied structure-from-motion estimation to generate range maps based on the overlapping field-of-view and real-world camera displacement between pairs of captured images [152]. This displacement was estimated using navigational information collected by other sensors on the AUV, but estimation could be avoided by using a

[3] β_c is most accurately modelled as a function of the range z, like $a_c(z)$, but this effect is negligible within a fixed water type [155].

[4] Details on the interpretation of each parameter are given in [153].

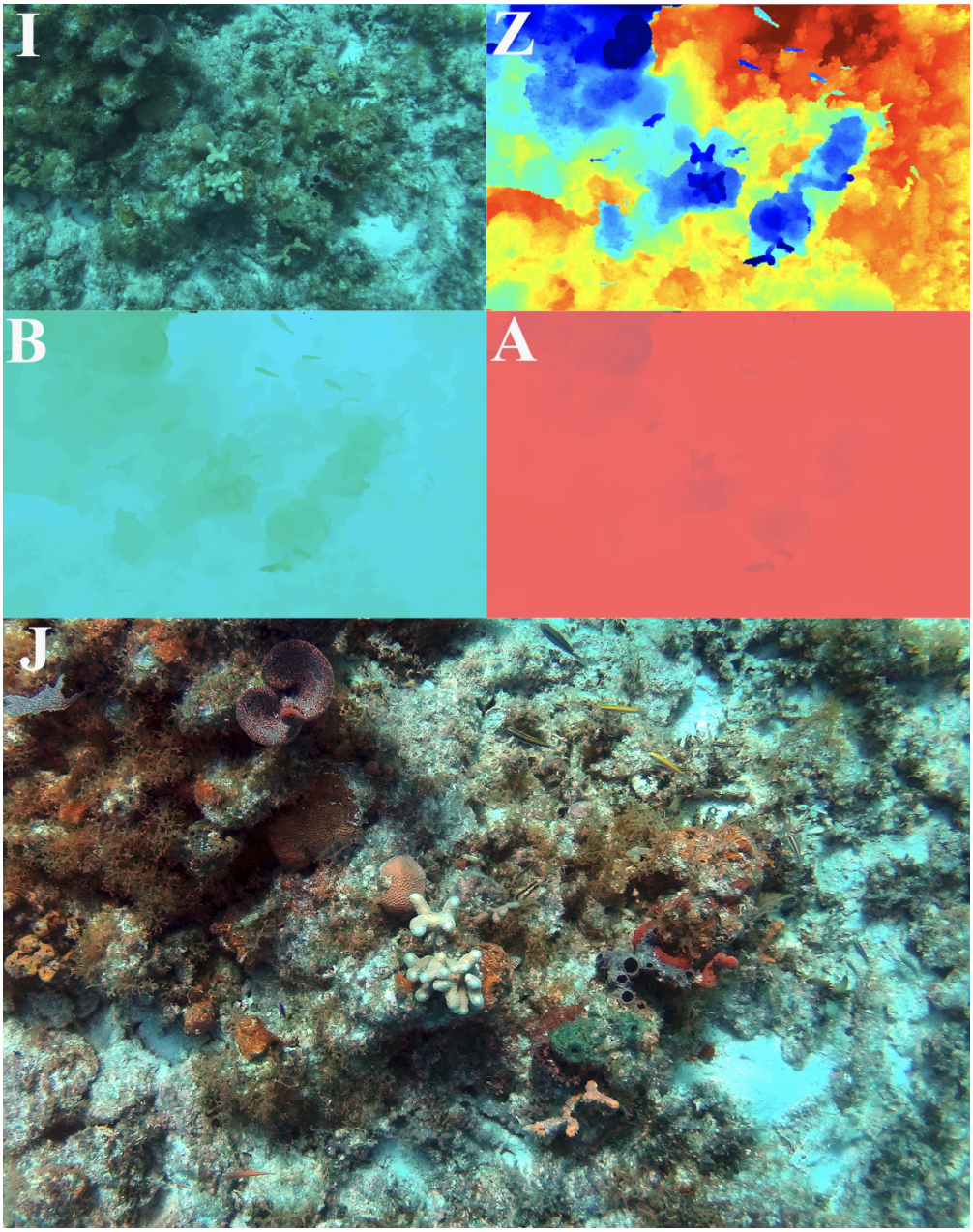


Figure 4.2: Examples of an input image I and corresponding range map Z (color-coded such that blue is "close" and red is "far"), best viewed on a screen in color. The DeepSeeColor method efficiently learns to approximate the backscatter image B and attenuation factor image A (both shown with contrast enhanced), enabling recovery of the true-color image J.

synchronized stereo-camera system to simultaneously capture pairs of images with a known, fixed camera displacement.

The previously discussed methods were developed for estimating the parameters of the traditional underwater image formation model described by Eqs. (4.1.2) and (4.1.3). Estimating the parameters of the improved underwater image formation model presented in Eqs. (4.1.4) and (4.1.5) originally required the use of multiple color chart calibration targets *in situ* [155]. The idea of using range maps, instead of color charts, to guide parameter estimation was adapted by the *Sea-Thru* algorithm [153], which enabled learning the improved model parameters using only the captured image and range map.

There has also been progress in learning-based methods for color correction [157], including for underwater imagery [158], [159]. However, these methods do not guarantee consistency or accuracy in the color reconstruction of an image stream, or images captured in different environmental conditions. This makes them less suitable than physics-based color correction methods for use in supporting realtime, possibly safety-critical, autonomous underwater behaviours.

4.1.3 AUV Vision System Considerations

Though [152] specifically explored color reconstruction for images *collected* by AUVs, to our knowledge no prior works have developed robust, physics-based underwater color reconstruction methods intended to run *onboard* an AUV's highly constrained computational resources in realtime. In fact, each of the aforementioned color reconstruction methods relies on solving optimization problems with computational complexity that is at least linear in the number of pixels in the input image [148]–[153]. The significant computational complexity of Sea-Thru [153], the only prior work to solve the parameters of the improved image formation model in Eqs. (4.1.5) and (4.2.6), will be explored further in Section 4.2.

Computationally expensive methods could be acceptable if the image formation parameters only needed to be computed once. However, prior works have found that these image formation parameters can change significantly in time and space due to factors including variation in depth, lighting conditions, exposure time, turbidity, and imaging angle [153], [155]. This demands the usage of an *adaptive* color correction method that is robust to changes in lighting and other environmental parameters, and can be run in realtime on each image as it is collected.

4.2 The DeepSeeColor Method

DeepSeeColor estimates the backscatter and attenuation parameters of the underwater image formation model from [155] by training two convolutional neural networks [160], depicted in Figures 4.3 and 4.4, respectively, with self-supervised loss functions based on the captured image I and range map Z. This training process can take advantage of deep learning hardware accelerators increasingly found onboard autonomous platforms [161].

4.2.1 Backscatter Estimation

DeepSeeColor makes use of the same backscatter model as Sea-Thru,

$$\hat{B}_c(i,j) = \gamma_c^{\infty}(1 - \exp(-\beta_c \cdot z_{i,j})) + \eta_c \exp(-\alpha_c \cdot z_{i,j}), \tag{4.2.1}$$

which corresponds to the backscatter model presented in Eq. (4.1.5) augmented with a residual term, described in [153], characterized by two new scalar parameters $\eta_c, \alpha_c \in \mathbb{R}_{\geq 0}$.

Inference

DeepSeeColor performs backscatter estimation using the neural network presented in Fig. 4.3. We define a novel nonlinear activation function, Gated Exponential Decay (GED), as

$$\mathrm{GED}(x) \coloneqq \begin{cases} 1 & x \leq 0, \\ \exp(-x) & x > 0. \end{cases} \tag{4.2.2}$$

We also introduce its complement, $\mathrm{CGED}(x) \coloneqq 1 - \mathrm{GED}(x)$. Observe that Eq. 4.2.1 can now be rewritten as

$$\hat{B}_c(Z) = \gamma_c^{\infty} \cdot \mathrm{CGED}(\beta_c Z) + \eta_c \cdot \mathrm{GED}(\alpha_c Z). \tag{4.2.3}$$

With its final sigmoid activation layer removed, the network depicted in Fig. 4.3 computes $\hat{B}_c$ for the backscatter parameters encoded in its convolutional layers' kernels. The operations $\beta_c Z$ and $\alpha_c Z$ for $c \in \{\mathtt{r}, \mathtt{g}, \mathtt{b}\}$ are modelled as convolutions between the tensor $Z = [z_{i,j}]$ with dimensions $(W, H, 1)$ and a kernel of shape $(1, 1, 1, 3)$ to produce an output tensor with dimensions $(W, H, 3)$, where the values of the kernel elements correspond to $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$, respectively.[5] Similarly, multiplication by the coefficients in $\boldsymbol{\gamma}^{\infty}$ and $\boldsymbol{\eta}$ can be modelled as convolutions with kernels of dimension $(1, 1, 3, 3)$, where each kernel is constrained to be

[5] Kernel shapes are given as (width, height, input channels, output channels).

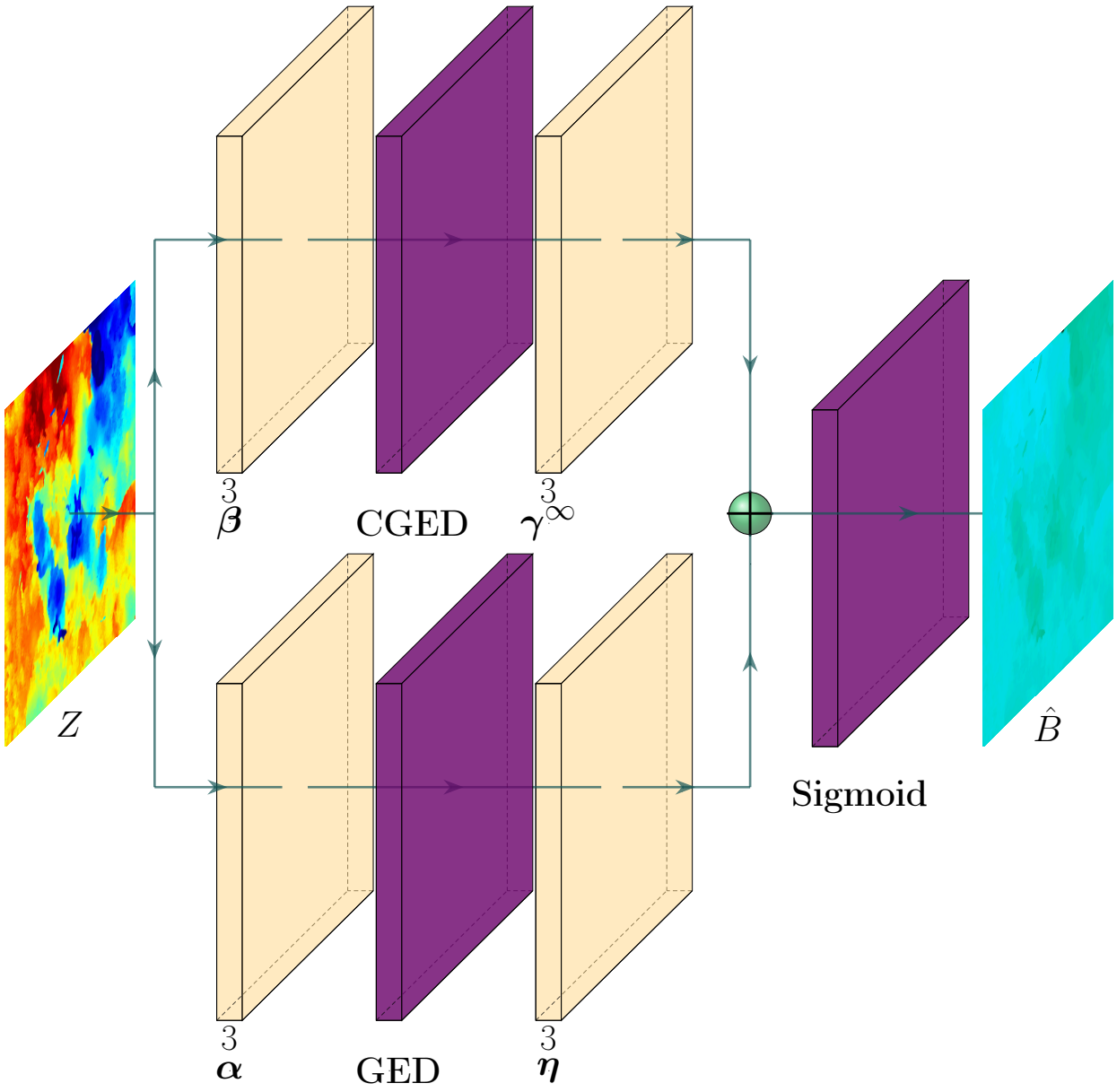


Figure 4.3: The backscatter net inputs a range map Z (visualized in color) and estimates the corresponding backscatter image $\hat{B}$. The kernel parameters in each convolutional layer map to the parameters of the backscatter estimation model in Eq. (4.2.1).

diagonal and thus has only 3 free parameters, resulting in the scaling of each channel by the corresponding γ_c^∞ or η_c. These parallel sequences of operations correspond to the top and bottom branches of the network in Fig. 4.3, respectively. The final sigmoid layer applies the activation function $\sigma(x) \coloneqq 1/(1+e^{-x})$, ensuring that the backscatter estimates remain bounded in $[0,1]$ regardless of any outliers in the range map.

Training

DeepSeeColor leverages the assumption from [153], [162] that in any given range interval there are some pixels in the image which should have zero intensity. It trains the backscatter network using a novel loss function,[6]

$$\mathcal{L}_{\text{bs}}(\hat{D}) = \sum_{(i,j)} \sum_{c} \big(\max\{\hat{D}_c(i,j), 0\} - k \min\{\hat{D}_c(i,j), 0\} \big), \tag{4.2.4}$$

with hyperparameter $k > 1$ and $\hat{D}_c \coloneqq I_c - \sigma(\hat{B}_c)$.

As $k \to \infty$, optimizing Eq. (4.2.4) becomes equivalent to finding backscatter parameters that minimize the minimum intensity of pixels at every range, while ensuring the network *never* predicts that a pixel in the direct signal would have negative intensity. As $k \to 1$, the loss function becomes tolerant of some pixels in the direct signal being predicted to have negative intensity, if that enables bringing more pixels closer to zero intensity. Intuitively, larger values of k would produce more accurate backscatter estimates if the range map was noiseless and the assumption of zero-intensity pixels at every range was satisfied, but are also less robust to noise and outliers in the image and range map. We find empirically that a value of $k = 1000$ produces accurate backscatter estimates.

For comparison, the backscatter parameters are estimated in Sea-Thru by binning the image pixels into 10 evenly spaced range windows, and assigning the darkest 1% of pixels in each bin to the set Ω [153]. Assuming these pixels should have zero-intensity, the backscatter parameters are then found by solving the nonlinear optimization problem,

$$\min_{\boldsymbol{\gamma}^\infty, \boldsymbol{\beta}, \boldsymbol{\eta}, \boldsymbol{\alpha}} \quad \sum_{(i,j)\in\Omega} \sum_{c} \left| I_c(i,j) - \hat{B}_c(i,j) \right|^2. \tag{4.2.5}$$

Constructing Ω using a bitonic merge sort has time complexity $O(NP^{-1}\log^2 N)$ for N pixels in the input image, when run in parallel on a processor with $P \leq N$ cores. In contrast, computing the loss function in Eq. (4.2.4) and backpropagating its gradients has $O(NP^{-1} + \log P)$ time complexity.

[6] Thanks to Dan Yang for finding an error in how this equation appeared in [25] (plus vs minus sign).

4.2.2 Attenuation Coefficient Estimation

The attenuation coefficient function $a_c(z)$ is well approximated as a double exponential function [153], characterized using the parameters $v_c, w_c, x_c, y_c \in \mathbb{R}_{\geq 0}$ as

$$a_c(z) = w_c \exp(-v_c \cdot z) + y_c \exp(-x_c \cdot z). \tag{4.2.6}$$

Inference

From Eqs. (4.1.4) and (4.2.6), it follows that

$$A_c(Z) = \mathrm{GED}\Big(Z \cdot \big(w_c \cdot \mathrm{GED}(v_c Z) + y_c \cdot \mathrm{GED}(x_c Z)\big)\Big). \tag{4.2.7}$$

Accordingly, the attenuation coefficient network depicted in Fig. 4.4 takes the same structure as the backscatter net, except the CGED and sigmoid activation functions are replaced with GEDs. The true-color image J is then estimated as,

$$\hat{J}_c(i,j) = \hat{D}_c(i,j) \cdot \exp(a_c(z_{i,j}) \cdot z_{i,j}). \tag{4.2.8}$$

Training

DeepSeeColor trains the attentuation network using a novel composite loss function designed to learn attenuation coefficient parameters similar to those of the Sea-Thru method,

$$\mathcal{L}_{ac}(\hat{J}; \hat{D}) = \mathcal{L}_{\text{saturation}}(\hat{J}) + \mathcal{L}_{\text{intensity}}(\hat{J}) + \mathcal{L}_{\text{var}}(\hat{J}; \hat{D}). \tag{4.2.9}$$

The saturation loss penalizes over-saturating pixels in the output image, and is defined as

$$\mathcal{L}_{\text{saturation}}(\hat{J}) = \frac{1}{3N} \sum_c \| \max\{\hat{J}_c - 1, 0\} \|^2,$$

where N is the number of pixels in $\hat{J}$. The intensity loss penalizes image channels which have a very low or very high average intensity, and is defined as

$$\mathcal{L}_{\text{intensity}}(\hat{J}) = \frac{1}{3} \sum_c \left(\frac{1}{N} \sum_{i,j} \Big(\hat{J}_c(i,j) - 0.5 \Big) \right)^2.$$

Finally, the variation loss penalizes changing the variation in each color channel across the

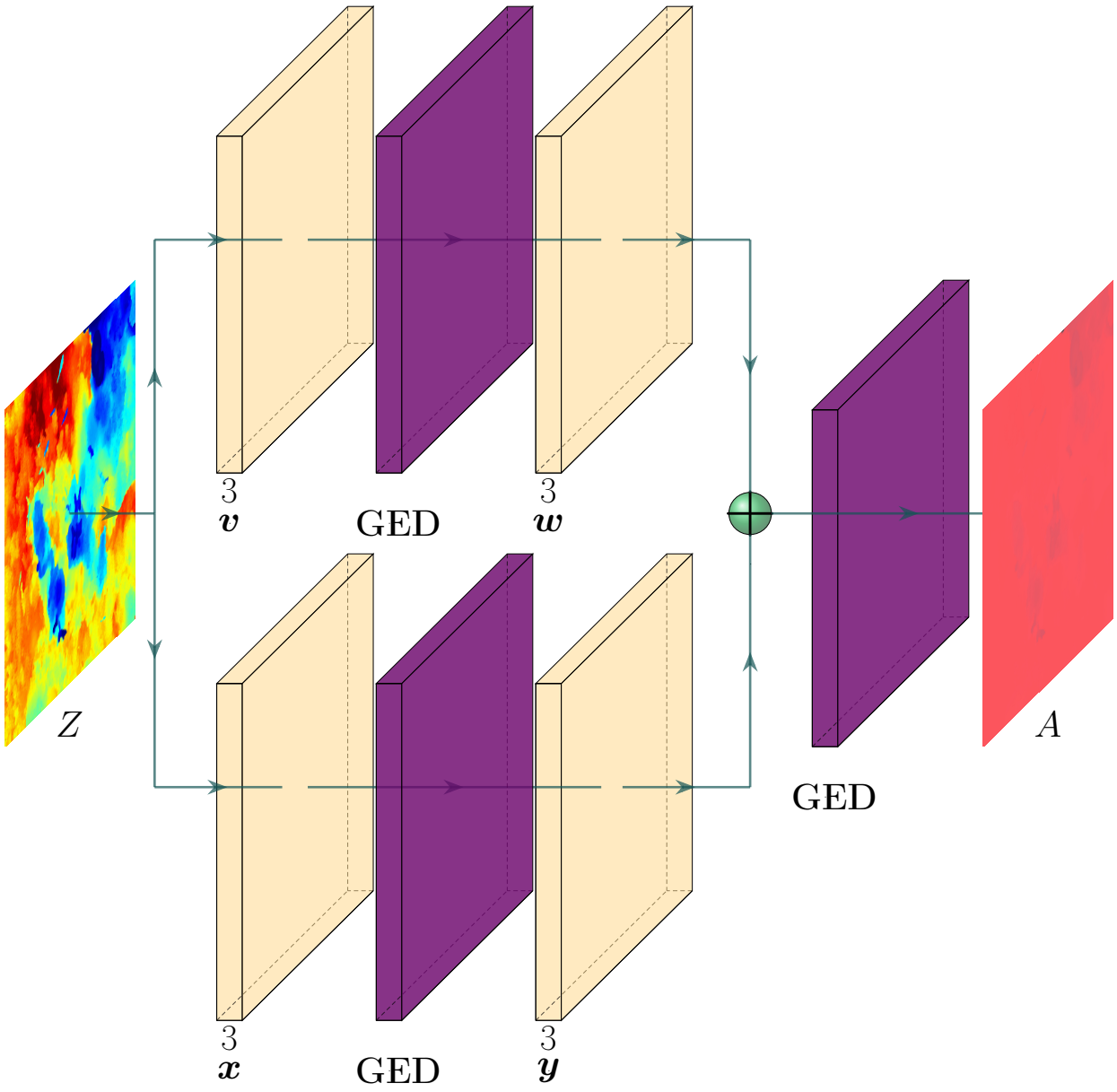


Figure 4.4: The attenuation net produces the attenuation map A from a range map Z. The kernel parameters in each convolutional layer map to the parameters of the attenuation coefficient function $a(z)$ in Eq. (4.2.6).

image, compared to the variation measured in the direct signal,

$$\mathcal{L}_{\text{var}}(\hat{J};\hat{D}) = \frac{1}{3}\sum_{c}\Big(s(\hat{J}_c) - s(\hat{D}_c)\Big)^2,$$

where $s(M)$ is the standard deviation of the values in M.

Minimizing the intensity loss function is similar to following the popular "gray-world" approach to white balancing [163], except the network is constrained to modifying the attenuation coefficients. This drives it to modify each channel's coefficients in a way that stretches contrast but biases the change towards targets far from the camera, and were thus most attenuated. The saturation and variation losses help to avoid very large attenuation coefficients; this is most relevant for images where attenuation has nearly eliminated a color channel such that very large coefficients would be required to recover it. Large coefficients cause over-saturation, and amplify noise in the input image; including the saturation and variation losses mitigates this issue.

This training process has $O(NP^{-1} + \log P)$ time complexity per training iteration when parallelized on a processor with $P \leq N$ cores. Furthermore, each iteration produces a color-corrected output image with increased quality, so the method is anytime optimal. For comparison, the Sea-Thru algorithm finds the attenuation model parameters using the LSAC algorithm [164] followed sequentially by a nonlinear optimization algorithm, which are each iterative methods with $O(N)$ time complexity per training iteration.

4.3 Experimental Results

The DeepSeeColor method has been implemented in PyTorch [165] and demonstrates strong performance on the Sea-Thru dataset [153] and on stereo-camera imagery collected with AUV deployments in the US Virgin Islands.

4.3.1 Sea-Thru Dataset

The Sea-Thru dataset consists of 1157 raw images, each with a corresponding range map generated using structure-from-motion techniques [153]. DeepSeeColor was run on all of the images available on the hosting site,[7] and a sampling of the outputs are presented in Fig. 4.5.

As in [153], the accuracy of the color reconstruction is estimated using the angular error between true gray and the grayscale patches on the color charts present in many of the

[7] http://csms.haifa.ac.il/profiles/tTreibitz/datasets/sea_thru/index.html

Table 4.1: Grayscale Patch Mean Angular Error, in degrees.

Image	Raw	Sea-Thru	DeepSeeColor (ours)
D1_3272	26	8	14
D2_3647	26	8	10
D3_4910	22	8	5
D4_0209	23	4	4
D5_3374	17/16/15/17	4/3/5/3	9/10/10/11
Average	20.25	5.375	9.125

dataset images. This angular error is computed at each grayscale patch in the color chart as

$$\psi(i,j) = \cos^{-1}\left(\frac{\sum_c J_c(i,j)}{\left(3 \cdot \sum_c [J_c(i,j)]^2\right)^{\frac{1}{2}}}\right). \tag{4.3.1}$$

The average angular error averaged over the six grayscale patches on each color chart for each of Sea-Thru and DeepSeeColor is presented in Table 4.1. As there is no public release of Sea-Thru, its errors are taken directly from [153]. The table further supports that that DeepSeeColor produces outputs with comparable quality to Sea-Thru.

The main advantage of DeepSeeColor over Sea-Thru is its fast runtime. Table 4.2 shows the time required to update the model parameters given a new input image, for each of Sea-Thru[8] and DeepSeeColor. The test image was `LFT_3374.NEF`, taken from the Sea-Thru dataset, scaled to either 1280x855 or 1920x1282 (1.1M and 2.5M, respectively), and the results were collected on a machine equipped with an Intel(R) i9-13900K processor and NVIDIA RTX A6000 GPU. As seen in the table, DeepSeeColor can perform updates at over 200x the rate of Sea-Thru on the CPU, and is nearly 4000x faster with GPU acceleration. Interestingly, the update rate of GPU-accelerated DeepSeeColor is less sensitive to input image resolution than the CPU-based methods; this is a result of the GPU processing capabilities of modern deep learning frameworks like PyTorch [165], which can process many sections of an image in parallel.

[8] As there is no public release of Sea-Thru, these results were produced with the unofficial implementation found at https://github.com/hainh/sea-thru.

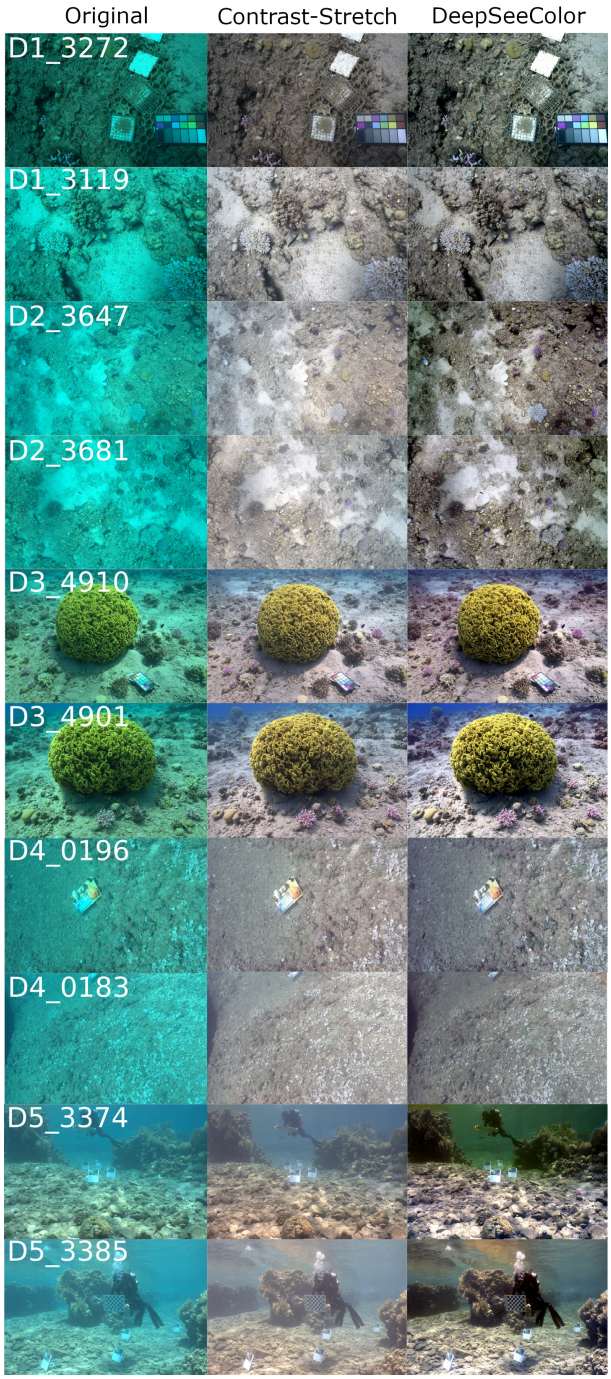


Figure 4.5: Best viewed on a screen, in color. The left column shows 10 raw images from the Sea-Thru dataset [153]. The center column shows the corresponding images after a contrast stretch, while the right column shows the outputs of DeepSeeColor. DeepSeeColor noticeably outperforms contrast stretching, especially in scenes with distant targets (e.g. 3647, 3681, and 3385), and achieves comparable performance to Sea-Thru (refer to Fig. 6 in [153] for comparison).

Table 4.2: Parameter Update Runtime Comparison.

# Pixels	Algorithm	Compute Device	Parameter Update Time (s)
1.1M	Sea-Thru	CPU	35.044
	DeepSeeColor	CPU	0.169
	DeepSeeColor	GPU	0.012
2.5M	Sea-Thru	CPU	86.310
	DeepSeeColor	CPU	0.519
	DeepSeeColor	GPU	0.022

Table 4.3: DeepSeeColor Runtime Breakdown.

# Pixels	Backscatter (per iter)	Attenuation (per iter)	Total (per iter)	Max Frequency
0.7M	6.2 ms	9.9 ms	16.1 ms	62.1 Hz
2.4M	6.4 ms	10.5 ms	16.9 ms	59.2 Hz

4.3.2 US Virgin Islands Dataset

The DeepSeeColor method was also evaluated to process imagery collected in the US Virgin Islands by the CUREE AUV [166]. The AUV was equipped with a downwards facing color stereo-camera, which enabled the collection of color imagery and range maps like those seen in Figs. 4.1 and 4.2.[9] The post-processed imagery has comparable quality to that generated by Sea-Thru, but is generated at a very high rate.

As seen in Table 4.3, DeepSeeColor can perform 60 training iterations of each network per second for this dataset, on a machine equipped with an Intel(R) Xeon E5-2630v4 and NVIDIA RTX A6000. This update rate corresponds to performing up to 60 incremental parameter updates per second based on a single image, or correcting up to 60 images per second with one incremental parameter update per image. This offers flexibility for an AUV to dedicate more processing power to training the DeepSeeColor model when imaging conditions change and the image formation parameters need updating, and less when imaging conditions are stable.

[9]Note that the range maps in this dataset used 16-bit integer ranges (in mm), while the Sea-Thru dataset uses 32-bit floating point values (in m).

4.4 Summary

In this chapter we presented DeepSeeColor, a novel color correction method that uses convolutional neural network training operations to learn the parameters of a physics-based underwater image formation model [155]. This approach to color correction is more robust than heuristic methods, and thus more appropriate for usage on AUVs performing safety-critical tasks. We demonstrated DeepSeeColor on the Sea-Thru dataset [153], as well as on images collected during field experiments in the US Virgin Islands. DeepSeeColor provides a large improvement in terms of runtime and computational complexity when compared to the Sea-Thru algorithm [153], while achieving comparable performance in color reconstruction. The results support that DeepSeeColor is well-suited for use in realtime preprocessing of imagery collected by an AUV to enable more robust execution of more sophisticated autonomous tasks.

Chapter 5

Endogenous Bayesian Risk Minimization

As discussed in Chapter 1, we use risk-based online learning to choose image queries that provide a robotic explorer with the most *useful* information for finding targets of interest. These methods have been highly effective at various online learning tasks [13], [15]–[18], but tend to be overly *myopic* by considering only the value of a single query action [19]. When the utility of any single query is close to 0, it makes it impossible to accurately determine which one would have significant longer-term utility (i.e., in combination with later queries). Most previous approaches limit themselves to considering only a single query because the complexity of computing the utility of information gained from *multiple* queries grows exponentially. Some prior works have attempted to mitigate this effect by considering a limited class of extended query sequences (e.g., [18]), but remain overly myopic [19], while new approaches abandon the key advantage of these risk-based systems to precisely quantify the utility of a query (e.g., [19], [167]–[169]).

In this chapter I present "Endogenous Bayesian Risk Minimization" (EBRM) as an approach to developing flexible, application-specific online learning algorithms.[1] In particular, I present an efficient non-myopic risk-based online learning algorithm called *AsympGreedy-EBRM*, and demonstrate that it outperforms the previous state-of-the-art across a wide range of applications beyond autonomous exploration, from assigning candidate treatments to patients in clinical trials (in order to maximize the number of successful treatments) to determining the optimal price at which a vendor should sell a product.

In Chapter 6, we will apply the AsympGreedy-EBRM algorithm to query selection in autonomous exploration. This will enable robots to learn reward models with the fewest possible number of queries/labels, and without the myopia of previous risk-based online learning algorithms. It will also enable directly quantifying the value of a query, such that

[1] The myopic algorithm I previously developed for risk-based query selection in autonomous exploration (see [13]) is equivalent to the Greedy-EBRM algorithm which will be presented in Section 5.4.1.

multiple robots can coordinate to determine which queries would be most useful to the fleet as a collective, or to determine if a query is worth asking at all. This makes AsympGreedy-EBRM a key part of our vision for human-multi-robot collaboration, in which each agent in the system is working towards the benefit of the entire team: by learning the reward model faster and with fewer queries, there is less input required from the human supervisor (thus freeing them to work on other tasks) and more communication bandwidth available for other purposes (such as sharing semantic maps).

5.1 Broader Context

There is a general trend towards autonomous decision-making in increasingly unstructured and complex tasks and environments, as autonomous decision-making agents become increasingly pervasive in many societies. Fully self-driving vehicles move passengers throughout cities, algorithms help diagnose and prescribe treatments to ill patients, and autonomous robots operate in environments ranging from homes and assisted-living facilities to Mars and the deep sea. While these agents attempt to make the best decisions possible despite limited information, decision-making under uncertainty always carries *risk*, and taking risks results in an accumulation of *regret* over past decisions. Maximizing aggregated long-term performance in a task is equivalent to minimizing the accumulation of this regret, and to do so autonomous agents employ algorithms for *online learning*, which describe techniques for "learning while doing".

Practitioners often use heuristics designed to optimally solve simple, archetypal online learning problems (OLPs) to instead solve all kinds of complex, real-world OLPs. These heuristics generally work well when compared to naïve strategies that do not leverage insights from online learning research. However, the complexities of real-world online learning problems cannot be accounted for by simple heuristics, despite the often important role of such complexities in determining the performance of a particular strategy. Due to their ease of use, practitioners faced with complex and novel online learning problems often choose to apply popular heuristics regardless, or to develop a new heuristic. This has led to a multitude of heuristics, many of which must be tuned for each new problem before they can achieve good performance. Furthermore, thorough analysis is required to determine whether the design of a candidate heuristic conflicts with desirable behaviour for the learning agent.

The most challenging part of designing an online learning algorithm is deciding how it will navigate the *exploration-exploitation trade-off*. Exploitation refers to an agent taking an action consistent with a plan that maximizes the agent's expected long-term task per-

formance, where that plan is based on a model of the likelihood of possible outcomes for each action. Finding such a plan, or a close approximation, can be achieved by a variety of planning algorithms, however such a plan is often poorly suited to discover inaccuracies in the model it is optimized for. Exploration is, conversely, the act of taking one or more actions expected to reveal missing information in the model, which may thereby enable a better (exploitative) plan to be developed for later use.

Finding and applying the correct balance between exploration and exploitation in a given task is a key online learning challenge for both researchers and practitioners; the optimal balance can shift for even slight changes in task, or in how task performance is defined. For example, Figure 5.1 describes a clinical trial online learning problem based on the work of [71]. This problem was complicated by dual objectives; the main goal of a clinical trial is to identify the best treatment among a set of candidates, but the well-being of the participants is also valued and thus it is preferable to assign as many of them to the best candidate treatments as possible. Some exploration is required to identify the leading treatments, while exploration beyond this point (i.e. continuing to assign participants to less-tested, but seemingly inferior, treatments) is beneficial for confirming the identity of the best treatment but is harmful to the study participants. Each online learning heuristic balances the trade-off differently. For this particular challenge, the researchers needed to develop a new heuristic to appropriately balance between these objectives [71].

As a result of such complexities, the field has seen the development of an overwhelming number of online learning heuristics [17]–[19], [72], [81], [83], [85], [87], [89], [167], [168], [170]–[176], many of which require extensive work to be tuned to a specific problem [177], and most offering performance guarantees in only archetypal OLPs, if any. A priority for the online learning community is to distill previous insights into an approach that is easy to use, computationally efficient, provides performance guarantees *and* can model various goals and common complexities. This work is a step in that direction, in which we present *endogenous Bayesian risk minimization* (EBRM) as an approach to solve a wide range of stochastic online learning problems efficiently, with risk (performance) bounds, and taking common complexities into account. In experiments, the EBRM-based algorithms demonstrate leading performance even in the well-studied archetypal OLPs that previous state-of-the-art algorithms compared against were designed for, while surpassing them to a greater degree in more complex and realistic problems.

An agent using an EBRM algorithm begins with a base "open-loop policy", which is the agent's best guess of which fixed sequence of future actions would maximize its overall task performance as measured by some reward function. The proposed Greedy-EBRM approach reasons about the quantity of immediately useful information each available action may pro-

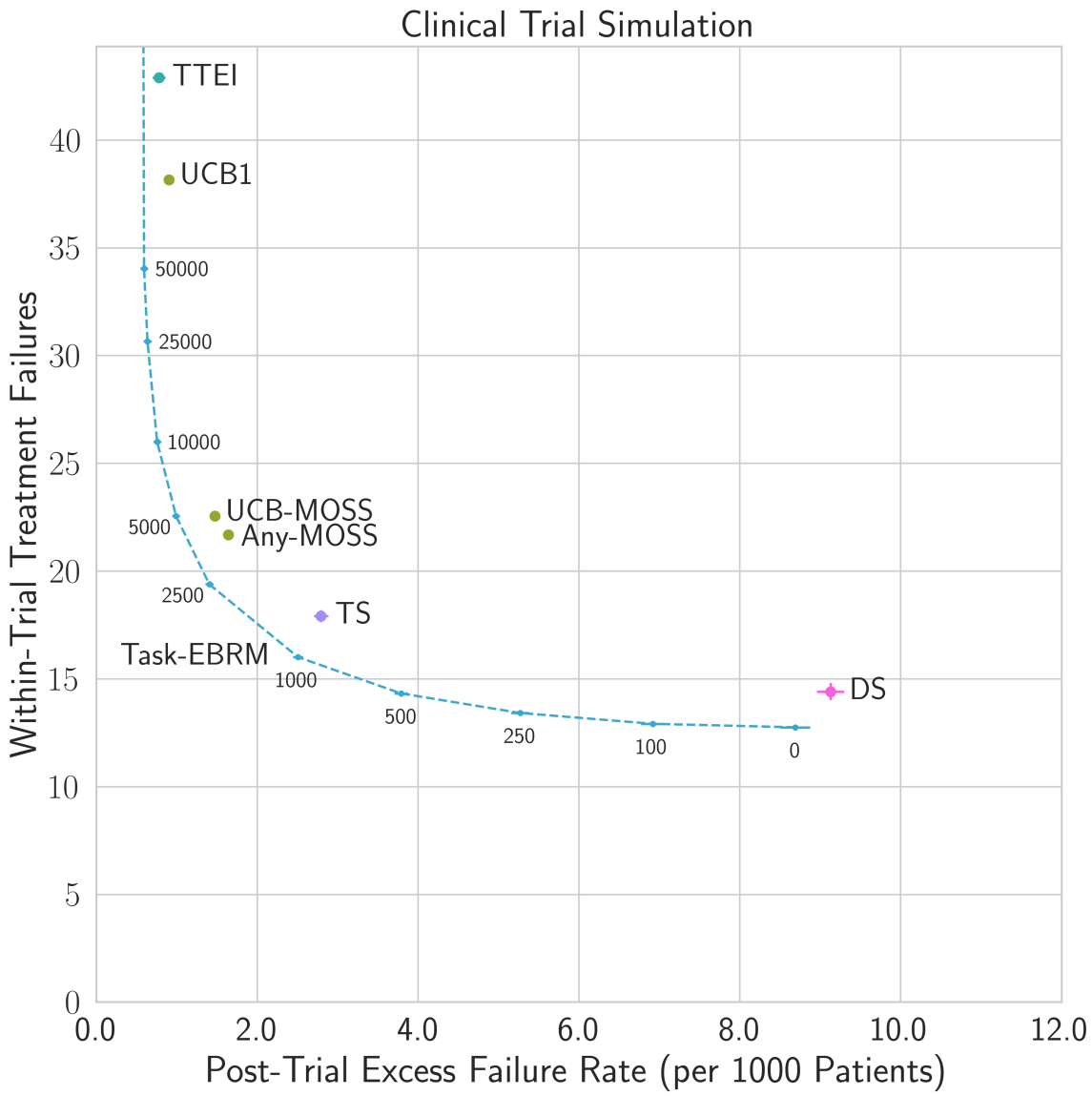


Figure 5.1: Consider a simulation where algorithms sequentially choose which of 4 candidate treatments is administered to each of 423 patients in a clinical trial (see "Multi-Arm Trial Setting" in [71]). The treatment with the highest empirical success rate in the trial is administered to a global patient population. The y-axis indicates the number of failed treatments among the patients in the trial, while the x-axis shows the expected excess failure rate of the treatment administered to the global population compared to if the *true* best treatment were identified. Algorithms further left *explore* more, ensuring that the best performing treatment is confidently identified, while algorithms further down *exploit* more, finding greater success among the 423 trial patients. The EBRM algorithm (blue curve) optimizes this trade-off by taking the global patient population size (annotations) into account in its parametric reward objective. Abbreviations are defined in Subsection 5.5.1.

vide about the hidden parameters of the online learning problem. If the total risk of taking some action and then following an improved (posterior) policy is expected to be less than the risk of the base (prior) policy, a Greedy-EBRM algorithm takes that action. AsympGreedy-EBRM, the main focus of this work, is an improvement on Greedy-EBRM which ensures asymptotic convergence to an optimal policy by taking into account the long-term value of the information provided by each action. EBRM approaches produce agent policies tailored to particular OLP specifications; thus, unlike when using heuristic approaches, the behaviour of agent is always aligned with the OLP goals (i.e., the reward function) without any hyperparameter tuning. We describe the EBRM approach in more detail in Section 5.4.

5.1.1 Summary of Contributions

In the remainder of this chapter, we present a mix of theoretical analysis and practical algorithms that serve as the foundation for further work in applying online learning principles and insights towards being able to efficiently solve complex real-world online learning problems. In particular, our contributions are applicable to a wide range of stochastic online learning problems, including problems augmented with belief-dependent rewards, time-discounted rewards, and action feasibility criteria. Specifically, we contribute:

- The decomposition of Bayes' risk into aleatoric, epistemic and process risks, which bound expected regret and provide insights into the optimal exploration-exploitation trade-off.
- The AsympGreedy-EBRM approach to online learning, which enables risk-bounded, high performance online learning in complex OLPs while remaining computationally tractable, and with guaranteed asymptotic convergence in problems with identifiable hidden parameters.
- Empirical results demonstrating the superior performance of EBRM algorithms against state-of-the-art baselines in online learning problems representative of real-world problems, including:
  - multi-armed bandit optimization,
  - best-arm identification,
  - dynamic pricing, and
  - a problem with mixed objectives, specifically rewards derived from a combination of the agent's actions *and* its posterior beliefs of the unknown problem parameters.

- Methods to efficiently compute, across a wide range of stochastic online learning problems, online regret bounds and the expected risk and value-of-information for various actions.

5.2 Online Learning as a Belief-Space Markov Decision Process

We begin by formulating stochastic online learning problems (OLPs), depicted in Figure 5.2, as belief-space Markov decision processes (BMDPs) with particular structure. We will then discuss how the performance of a *policy* is measured in an OLP, and how this relates to defining the *risk* of that policy.

Notation We index variables related to sequential decisions (BMDP states) using the time $t \in \mathbb{N}_{>0}$. We define $[K] \coloneqq \{1, \dots, K\}$ for $K \in \mathbb{N}_{>0}$. We denote the space of probability measures over a Borel measurable set $\mathcal{X}$ as $\mathcal{P}(\mathcal{X})$, and the power set of a set $\mathcal{X}$ as $\mathfrak{P}(\mathcal{X})$. The indicator function $\mathbb{1}$ is defined as $\mathbb{1}(\text{True}) = 1$ and $\mathbb{1}(\text{False}) = 0$.

5.2.1 Bayesian BMDP Formulation of Stochastic OLPs

Let $X = \{x_t \in \mathcal{X}\}_{t \in \mathbb{N}_{>0}}$ denote the hidden *outcomes* of the OLP. When an agent performs action $a_t \in \mathcal{A}$ at time t, it produces an observation $y_t = \Phi(x_t, a_t)$ and a reward $R(x_t, a_t)$, according to known functions Φ and R. A *stochastic* OLP is one in which each hidden outcome is assumed to be independently drawn from $g_\theta \in G_\Theta$, where G_Θ is a family of probability distributions over $\mathcal{X}$ parameterized by $\theta \in \Theta$; thus, the hidden outcomes are i.i.d. given θ.

The agent's actions are chosen by a policy π that takes into account the agent's current *belief* of the hidden parameters $b_t \in \mathcal{P}(\Theta)$ and an observable *auxiliary state* $\xi_t \in \Xi$. The belief describes the likelihood of hidden parameter values, and is used to infer the expected reward of each action. The auxiliary state indicates the time and resources available to the agent and determines which actions are *feasible*. Together, these compose the agent's *belief state*, $S_t = \{b_t, \xi_t\}$. The policy generates actions according to $a_{t+1} \sim \pi(S_t)$.

The auxiliary state ξ_t contains all information relevant to the agent's decision making, other than the hidden parameters. For example, if actions have time or resource costs, it indicates the agent's remaining time or spending budget. The auxiliary state changes according to $\xi_t = \delta(\xi_{t-1}, a_t)$; importantly, it changes deterministically given sequence of actions. The feasibility criterion $\Omega(\xi_t)$ indicates which actions are available in each state, so

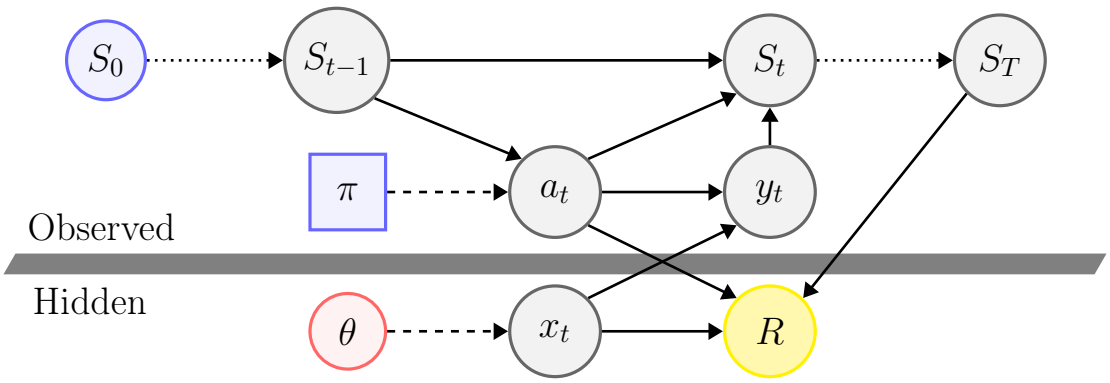


Figure 5.2: The Environment (red) picks a set of hidden parameters θ. The User (blue) specifies the learning agent's initial state $S_0 = \{b_0, \xi_0\}$ and a policy function π that defines the distribution from which to draw each action, such that $a_t \sim \pi(S_{t-1})$. The goal is to find a policy that maximizes the amount of reward (yellow) collected, R, which depends on the action-outcome pairs (x_t, a_t) for $t = 1, \ldots, T$, and on the agent's final state S_T. Dashed arrows represent random sampling from a probability distribution, and dotted arrows represent omitted parts of the graph. All variables and their relationships are defined in Table 5.1.

Table 5.1: Components of a Stochastic OLP. Vec: Vector. Dist: Distribution. Fn: Function. Comp: Compound.

OLP Parameter	Observability	Type	Symbol	Determined by	Space
Hidden Parameters	Inferred	Vec.	θ	Environment	Θ
Hidden Outcomes	Inferred	Vec.	x	$x_t \sim g_\theta$	$\mathcal{X}$
Actions	Observed	Vec.	a	$a_t \sim \pi(S_{t-1})$	$\mathcal{A}$
Observations	Observed	Vec.	y	$y_t = \Phi(x_t, a_t)$	$\mathcal{Y}$
Auxiliary State	Observed	Vec.	ξ	$\xi_t = \delta(\xi_{t-1}, a_t)$	Ξ
Belief Distribution	Observed	Dist.	b	Eq. (5.2.1)	$\mathcal{P}(\Theta)$
Belief State	Observed	Comp.	S	$S_t = \{b_t, \xi_t\}$	$\mathcal{S} = \mathcal{P}(\Theta) \times \Xi$
Policy	Known	Fn.	π	User	$\mathcal{S} \to \mathcal{P}(\mathcal{A})$
State Update Fn.	Known	Fn.	δ	OLP	$\Xi \times \mathcal{A} \to \Xi$
Action Reward Fn.	Known	Fn.	$R(x, a)$	OLP	$\mathcal{X} \times \mathcal{A} \to \mathbb{R}$
Belief Reward Fn.	Known	Fn.	$R(b)$	OLP	$\mathcal{P}(\Theta) \to \mathbb{R}$
Observation Fn.	Known	Fn.	Φ	OLP	$\mathcal{X} \times \mathcal{A} \to \mathcal{Y}$
Process Model	Known	Fn.	g_θ	OLP	$\Theta \to \mathcal{P}(\mathcal{X})$
Feasibility Criterion	Known	Fn.	Ω	OLP	$\Xi \to \mathfrak{P}(\mathcal{A})$
Discount Factor	Known	Scalar	γ	OLP	$\gamma \in (0, 1]$

a "feasible" policy π must satisfy $\Pr(a_{t+1} \in \Omega(\xi_t)) = 1$ for $a_{t+1} \sim \pi(S_t)$. The agent ceases from taking further actions upon reaching a *terminal state*, indicated by $\Omega(\xi_t) = \varnothing$.

The belief distribution b_t characterizes the posterior likelihood of the hidden parameters θ based on past action-observation pairs $\{(a_1, y_1), \ldots, (a_t, y_t)\}$. For convenience, we denote arbitrary belief distributions as b or b'. The initial belief distribution $b_0(\theta)$ is specified by the user as a prior over the hidden parameters. The observation function Φ implicitly defines the likelihood $\Pr(y_t \mid \theta, a_t)$, and so b_t is given by Bayes' law,

$$b_t(\theta) \coloneqq \Pr(\theta \mid a_{1:t}, y_{1:t}) = \frac{\Pr(y_t \mid \theta, a_t) b_{t-1}(\theta)}{\Pr(y_t \mid a_t)}. \tag{5.2.1}$$

We are often interested in *counterfactual* belief states, which may arise when considering a possible next outcome-action pair (x, a). Such a belief state is denoted $S_{t+1}^{x,a} = \{b_{t+1}^{x,a}, \xi_{t+1}^{a}\}$, with

$$b_{t+1}^{x,a}(\theta) = \frac{\Pr(\Phi(x, a) \mid \theta, a) b_t(\theta)}{\Pr(\Phi(x, a) \mid a)}, \quad \xi_{t+1}^{a} = \delta(\xi_t, a). \tag{5.2.2}$$

The components discussed thus far specify the BMDP context; the remaining part of the specification is the BMDP goal. A variety of useful goals can be expressed through action-based rewards accumulated as actions are taken and an information-based reward based on the terminal belief state. Accordingly, we define the *action reward* $R(x, a) \in \mathbb{R}$ and *belief reward* $R(b) \in \mathbb{R}$, such that the goal is to maximize the sum,

$$\gamma^T R(b_T) + \sum_{\tau=1}^{T} \gamma^\tau R(x_\tau, a_\tau), \tag{5.2.3}$$

evaluated upon reaching a terminal state $S_T = \{b_T, \xi_T\}$ where $\Omega(\xi_T) = \varnothing$. The discount factor $\gamma \in (0, 1]$ specifies the degree to which earlier rewards are preferred to later rewards. The BMDP specification requires a *Markovian* reward model ρ, which depends only on the current state and action; it suffices to define,

$$\rho(S_t, a) \coloneqq \mathbb{E}_{x|b_t}\big[R(x, a) + R(b_{t+1}^{x,a}) \mathbb{1}\big(\Omega(\xi_{t+1}^{a}) = \varnothing\big)\big], \tag{5.2.4}$$

where $b_{t+1}^{x,a}$ and ξ_{t+1}^{a} are defined as in Eq. (5.2.2). Together, the reward model ρ and discount factor γ formally specify the BMDP goal.

5.2.2 Optimal Online Learning

Every policy π has an associated *value function*, $V^\pi(S)$, defined recursively by the Bellman equation [90],

$$V^\pi(S_t) \coloneqq \mathbb{E}_{a\sim\pi(S_t)}\big[\rho(S_t,a) + \mathbb{E}_{x|b_t}\big[\gamma V^\pi(S_{t+1}^{x,a})\big]\big], \tag{5.2.5}$$

$$\Pr(x \mid b_t) = \int_\Theta g_\theta(x) b_t(\theta) d\theta. \tag{5.2.6}$$

By construction, $V^\pi(S)$ is equal to the terminal reward expected to be received by following the policy π until reaching a terminal state, as computed in Eq. (5.2.3).

The calculation in Eq. (5.2.5) assumes the distribution of each observation follows from the conditional density $\Pr(x \mid b)$ generating the hidden outcome x based on the previous belief state. We also consider *conditional* value functions $V^\pi(S_t;\theta)$ and $V^\pi(S_t;X)$, which represent the reward achieved by the policy π for a specific θ or sequence of hidden outcomes X respectively,

$$V^\pi(S_t;\theta) \coloneqq \mathbb{E}_{a\sim\pi(S_t)}\big[\mathbb{E}_{x|\theta}\big[R(x,a) + R(b_{t+1}^{x,a})\mathbb{1}\left(\Omega(\xi_{t+1}^a) = \varnothing\right) + \gamma V^\pi(S_{t+1}^{x,a};\theta)\big]\big], \tag{5.2.7}$$

$$V^\pi(S_t;X) \coloneqq \mathbb{E}_{a\sim\pi(S_t)}\big[R(x_{t+1},a) + R(b_{t+1}^{x_{t+1},a})\mathbb{1}\left(\Omega(\xi_{t+1}^a) = \varnothing\right) + \gamma V^\pi(S_{t+1}^{x_{t+1},a};X)\big]. \tag{5.2.8}$$

From these definitions, it follows that $V^\pi(S_t) \equiv \mathbb{E}_{\theta|b_t}[V^\pi(S_t;\theta)] \equiv \mathbb{E}_{X|b_t}[V^\pi(S_t;X)]$.

Types of Policies

A *deterministic* policy π is one that always picks the same action for a given belief state; that is, if $a \sim \pi(S)$ then $\exists a' : \Pr(a = a') = 1$. Any policy which is not deterministic is called a *stochastic* policy. An *open-loop* policy is one for which the action distribution is a function of only the auxiliary state ξ_t; that is, one for which $\pi(S) = \pi(S')$ holds $\forall S, S'$ such that $\xi = \xi'$. Otherwise, the policy is a *closed-loop* policy. A policy which is both deterministic *and* open-loop can be completely characterized, for a given initial belief state, by the fixed sequence of actions that it would take from that state. We denote a deterministic open-loop policy and its corresponding action sequence as $\hat{\pi}$ and $A^{\hat{\pi}}$, respectively.

Policy Optimality

A b-optimal policy $\pi_b^\star$ is a solution to,

$$\pi_b^\star \in \arg\max_{\pi\in\Pi} V^\pi(S_0) \tag{5.2.9}$$

where Π is the set of all policies. A b-optimal policy achieves the most terminal reward possible, in expectation, from any initial state S_0, assuming that $\theta \sim b_0$. Similarly, a θ-optimal policy $\pi^\star_\theta$ and X-optimal policy $\pi^\star_X$ satisfy, from the initial state S_0,

$$\pi^\star_\theta \in \arg\max_{\pi \in \Pi} V^\pi(S_0; \theta), \tag{5.2.10}$$

$$\pi^\star_X \in \arg\max_{\pi \in \Pi} V^\pi(S_0; X). \tag{5.2.11}$$

As we assume the hidden outcomes are independently and identically distributed, it suffices to search the set of *deterministic* policies in order to find a b-, θ-, or X-optimal policy for an OLP.[2]

Most optimal policies are closed-loop, even if they are deterministic. However, deterministic open-loop policies are useful to consider as they are straightforward to describe and analyze. In particular, later sections will often refer to "optimal deterministic open-loop policies", defined below.

Definition 1. An **optimal deterministic open-loop (ODOL) policy** $\hat{\pi}^\star_{S_t}$ is characterized by, where $\hat{\Pi} \subset \Pi$ denotes the set of deterministic open-loop policies,

$$\hat{\pi}^\star_{S_t} \in \arg\max_{\hat{\pi} \in \hat{\Pi}} V^{\hat{\pi}}(S_t). \tag{5.2.12}$$

5.2.3 Measuring Policy Risk

A *risk function* measures how much less reward is expected to be attained by one policy π_1 compared to another policy π_2. We define three fundamental risk functions in Table 5.2.

Risk is often defined relative to an optimal policy so that it represents a shortfall relative to the maximum achievable reward.[3] We define some Bayesian risks relative to the X- and θ-optimal policy "families",

$$\overline{\overline{r}}(\pi \| \pi^\star_X; S_t) \coloneqq \mathbb{E}_{X|b_t}[r(\pi \| \pi^\star_X; S_t, X)], \tag{5.2.13}$$

$$\overline{\overline{r}}(\pi \| \pi^\star_\theta; S_t) \coloneqq \mathbb{E}_{\theta|b_t}[\overline{r}(\pi \| \pi^\star_\theta; S_t, \theta)]. \tag{5.2.14}$$

It is important to note the abuse of notation here, where the terms $\pi^\star_\theta$ and $\pi^\star_X$ on the left side of each equation are not specific policies, but rather represent conceptual policies optimal

[2] Prior works have explored optimal online learning policies when the hidden outcomes are generated by an *adversarial* process, for which stochastic policies can greatly outperform deterministic ones.

[3] In this case it also represents the (expected) future *regret* of a policy; regret is the evaluation metric for most online learning problems. A.1 presents further discussion on the relationship between risk and regret.

Table 5.2: Risk functions; the risk of policy π_1 is defined with respect to policy π_2.

Name	Definition
Instance Risk	$r(\pi_1 \| \pi_2; S, X) \coloneqq V^{\pi_2}(S; X) - V^{\pi_1}(S; X)$
Expected Risk	$\overline{r}(\pi_1 \| \pi_2; S, \theta) \coloneqq V^{\pi_2}(S; \theta) - V^{\pi_1}(S; \theta)$
Bayesian Risk	$\overline{\overline{r}}(\pi_1 \| \pi_2; S) \coloneqq V^{\pi_2}(S) - V^{\pi_1}(S)$

for *any* particular realization of θ and X, respectively. We can think of $\pi^\star_\theta$ and $\pi^\star_X$ in such contexts as "cheating" policies, which suggest actions based on information unavailable in the belief state.

We demonstrate in Section 5.3 that insights can be made by considering the risk of a policy with respect to the various optimal policies. Lemma 1 shows how a "hierarchy" of risk arises from comparing a policy π against these different optimal policies.

Lemma 1. *For any policy $\pi \in \Pi$ and any belief state $S_t \in \mathcal{S}$, we have that*

$$\overline{\overline{r}}(\pi \| \pi^\star_X; S_t) \geq \overline{\overline{r}}(\pi \| \pi^\star_\theta; S_t) \geq \overline{\overline{r}}(\pi \| \pi^\star_b; S_t) \geq 0.$$

Proof. Expand the risk functions in terms of the (conditional) value functions, express the values of optimal policies as maximization problems based on Eqs. (5.2.9)–(5.2.11), and apply Jensen's inequality. □

5.2.4 Exogenous and Endogenous Risk Metrics

We distinguish between *exogenous* and *endogenous* measures of a learning agent's performance, which characterize whether a metric is computed from only the belief state S_t, or from external information.

An *exogenous* metric is a function that requires inputs which are not part of S_t. This makes it a means of external evaluation of the performance of a policy, as it requires knowing values of variables not provided to the learning agent. This may include the true values of X and θ, which could be known under experimental conditions by the experimentalist or in other settings by an "oracle". Such a metric can be an effective tool for evaluating different policies, but a real learning agent is unable to compute it online.

Conversely, an *endogenous* metric depends only on the agent's own state estimate; that is, it is a function of only S_t. As such, when an agent estimates the risk of its own policy we refer to it as an endogenous risk estimate. These are endogenous in that they can be computed using only S_t, the information available to the agent at time t. The algorithms

contributed by this work are based on minimizing endogenous Bayesian risk *online*.

5.3 Online Computation of Endogenous Risks

Quantifying the risk of policies is useful for many tasks, such as identifying the best policy from some set or providing safety and performance guarantees. This section will explain how endogenous risk measures provide insights into the nature of an OLP and into the behaviour of a specific policy, and will discuss the feasibility of computing the risk and value of various classes of policies. Note that all measures of risk in this section will be endogenous, in that X and θ are taken to be random variables distributed according to $\Pr(X, \theta \mid b_t)$.

5.3.1 Endogenous Bayesian Risk Functions

The risk of a policy, as defined in Subsection 5.2.3, is always given relative to some "reference" policy; the choice of this reference policy offsets the risk estimate, as $\forall \pi_1, \pi_2, \pi_3$,

$$\overline{\overline{r}}(\pi_1 \| \pi_3; S) = \overline{\overline{r}}(\pi_1 \| \pi_2; S) + \overline{\overline{r}}(\pi_2 \| \pi_3; S). \tag{5.3.1}$$

By choosing reference policies that are "optimal", the risk computed can give insight into the OLP and into policies of interest. The first, and perhaps most important, risk function is the *total Bayesian risk*.

Definition 2. The **total Bayes' risk** of a policy π from some belief state S is

$$\text{TotalRisk}(\pi; S) \coloneqq \overline{\overline{r}}(\pi \| \pi^\star_X; S).$$

We refer to this quantity as the "total risk" because $\pi^\star_X$ is the "policy" with the highest possible value, so the total Bayes' risk is the largest possible risk of π. The total risk has an intuitive interpretation: it indicates how much better an agent following some policy π could perform if it instead made perfect use of complete information regarding all of the hidden outcomes.

Total risk is a well-studied metric in online learning but is often challenging to compute, and considering it in isolation conceals insights into the OLP structure, the belief state, and the policy. The remainder of this subsection will explore a useful decomposition of the total risk into three distinct and edifying components: the *aleatoric* and *epistemic* risks of a belief state and the *process* risk of the policy.

Aleatoric Risk

Due to the stochastic nature of the hidden outcomes X, even complete information about the parameters θ is generally not enough for the agent to be able to achieve the maximum possible reward. This limitation is inherent to the OLP itself, and is captured by the *aleatoric* risk. The aleatoric risk thus provides insight into the difficulty of an OLP; if it is by far the largest component of the Bayes' risk of a policy π, then most of the risk of the policy is due to random chance and cannot be eliminated. If this risk is unacceptably high, it may indicate that the OLP cannot be satisfactorily solved.

Definition 3. The **aleatoric risk** of an OLP with parameters θ is

$$\text{AleatoricRisk}(\theta) \coloneqq \overline{r}(\pi^\star_\theta \| \pi^\star_X; \theta).$$

The **aleatoric Bayes' risk** of a belief state S is accordingly defined as

$$\text{AleatoricBayesRisk}(S) \coloneqq \overline{\overline{r}}(\pi^\star_\theta \| \pi^\star_X; S) = \mathbb{E}_{\theta|b}[\text{AleatoricRisk}(\theta)].$$

For any parameters θ, the aleatoric risk is non-negative (see Lemma 1), showing that even with perfect knowledge of the parameters θ, the optimal policy $\pi^\star_\theta$ may not achieve the maximum possible reward. The aleatoric *Bayes'* risk of S_t measures how much aleatoric risk is expected to remain even after being given complete information on θ, assuming the parameters θ were sampled from b_t. Below, we provide an example of computing aleatoric risk and aleatoric Bayes' risk in a toy problem.

Problem 1. *Suppose a coin flips heads with probability $\theta \in [0, 1]$ and produces hidden outcomes in $\mathcal{X} = \{\text{Heads}, \text{Tails}\}$. An agent has two actions $\mathcal{A} = \{1, 2\}$, and the action reward function is:*

$$R(x, 1) = \begin{cases} 1, & x = \text{Heads}, \\ 0, & x = \text{Tails}. \end{cases} \qquad R(x, 2) = \begin{cases} 0, & x = \text{Heads}, \\ 1, & x = \text{Tails}. \end{cases}$$

There is no belief reward or discounting, so $R(b) = 0$ and $\gamma = 1$. The agent plays for n rounds, so $\Omega(\xi_t) = \mathcal{A}$ for $t < n$ and $\Omega(\xi_n) = \varnothing$. The initial belief distribution is $b_0 = \text{Beta}(1, 1)$.

Example 1. In Problem 1, the unique X-optimal policy is to take action 1 when $x_t = \text{Heads}$ and action 2 when $x_t = \text{Tails}$. An θ-optimal policy is to always take action 1 if $\theta > 0.5$, and to

take action 2 otherwise. It is straightforward to compute that, over n rounds, $V^{\pi^\star_X}(S_0;\theta) = n$ and $V^{\pi^\star_\theta}(S_0;\theta) = n\max\{(1-\theta),\theta\}$. Thus $\text{AleatoricRisk}(\theta) = n - \max\{n(1-\theta), n\theta\}$ and

$$\begin{aligned}\text{AleatoricBayesRisk}(b_0) &= \mathbb{E}_{\theta|b_0}[n - n\max\{(1-\theta),\theta\}],\\ &= \int_0^{0.5}(n - n(1-\theta))d\theta + \int_{0.5}^{1}(n-n\theta)d\theta = \frac{n}{4}.\end{aligned}$$

The aleatoric risk and aleatoric Bayes' risk for various θ and b, respectively, are shown in Figure 5.3.

Epistemic Risk

As a learning agent gradually learns more about the hidden parameters θ, it is typically able to improve its performance. While it can never eliminate the aleatoric risk inherent to the OLP, this learning process reduces the *epistemic* (knowledge-based) risk of the belief state, defined as follows.[4]

Definition 4. The **epistemic risk** of a belief state S_t is

$$\text{EpistemicRisk}(S_t) \coloneqq \overline{\overline{r}}(\hat{\pi}^\star_{S_t} \| \pi^\star_\theta; S_t).$$

By Lemma 1, the epistemic risk is always non-negative. Importantly, however, the epistemic risk of an agent *can be reduced* through performing actions that lead to belief states with better estimates of the hidden parameters. The reduction of epistemic risk to 0 depends on the hidden parameters being *identifiable*.

Definition 5. Let $\xrightarrow[t\to\infty]{P}$ denote convergence in probability. The hidden parameters θ are **identifiable** from b_0 if and only if there exists an infinite sequence of actions $A \in \mathcal{A}^\infty$ such that $\forall \eta \in \Theta : |\eta - \theta| > 0$,

$$\mathbb{E}_{X|\theta}\left[b_t^{X,A}(\eta)\right] \xrightarrow[t\to\infty]{P} 0, \tag{5.3.2}$$

where $b_t^{X,A}(\eta)$ is defined recursively by Bayes's law as, with $y_t = \Phi(X(t), A(t))$ and $b_0^{X,A} = b_0$,

$$b_t^{X,A}(\eta) = \frac{\Pr(y_t \mid \theta, A(t)) b_{t-1}^{X,A}(\eta)}{\Pr(y_t \mid A(t))}. \tag{5.3.3}$$

[4] In reinforcement learning, some works (e.g. [106], [178], [179]) refer to the epistemic risk of a learned *policy*, which results from a lack of training data in the vicinity of the S_t. In our notation, this risk would be denoted as $\overline{\overline{r}}(\pi \| \pi^\star_{S_t}; S_t)$.

Theorem 1. *Suppose the hidden parameters θ are identifiable over some infinite sequence of actions. Then, over this sequence of actions,*

$$\text{EpistemicRisk}(S_t) \xrightarrow[t\to\infty]{P} 0. \tag{5.3.4}$$

Proof. See A.4. □

Theorem 1 implies that, if the belief distribution converges upon the true parameters θ, then the difference in value between an ODOL policy $\hat{\pi}^\star_{S_t}$ and any θ-optimal policy converges to 0. This reflects that $\hat{\pi}^\star_{S_t}$ generally "improves" as actions are performed and the belief state converges. Estimates of the changes in epistemic risk that would result from various actions are useful as they can guide the learning agent towards states where the ODOL policy $\hat{\pi}^\star_{S_t}$ is near-optimal; this strategy will be discussed further in Section 5.4. Example 2 demonstrates how to compute epistemic risk in a simple OLP, while Figure 5.4 depicts how the epistemic risk in Problem 1 varies with the belief distribution.

Example 2. In Problem 1, the deterministic open-loop-optimal policy is to choose action 1 if $\mathbb{E}_{\theta|b}[\theta] \geq 0.5$, and action 2 otherwise. For a beta belief distribution $b_0 = \text{Beta}(\alpha, \beta)$ with $\alpha, \beta \in \mathbb{R}_{>0}$, then

$$\mathbb{E}_{x|b_0}\left[R(x, \hat{\pi}^\star_{S_0}(S_0)\right] = \frac{\max\{\alpha,\beta\}}{\alpha+\beta} \implies V^{\hat{\pi}^\star_{S_0}}(S_0) = \frac{n\max\{\alpha,\beta\}}{\alpha+\beta}$$

Thus,

$$\text{EpistemicRisk}(\text{Beta}(\alpha,\beta)) = \mathbb{E}_{\theta\sim\text{Beta}(\alpha,\beta)}[n\max\{1-\theta,\theta\}] - \frac{n\max\{\alpha,\beta\}}{\alpha+\beta}$$

The epistemic risk for various belief distribution parameters α and β is presented in Figure 5.4.

Process Risk

Lastly, the *process* risk of any policy π measures the excess risk of π over an ODOL policy $\hat{\pi}^\star_{S_t}$. Accordingly, the process risk of any open-loop policy is non-negative, but it can be negative with respect to a closed-loop policy. Essentially, process risk measures how much better or worse π performs relative to the best policy that ignores any and all future observations. This means it plays a key role in evaluating the *value-of-information*; if the agent can't manage to find any (closed-loop) policy with process risk $\ll 0$, it means that there is little value in future observations. For example, in human-in-the-loop autonomous exploration,

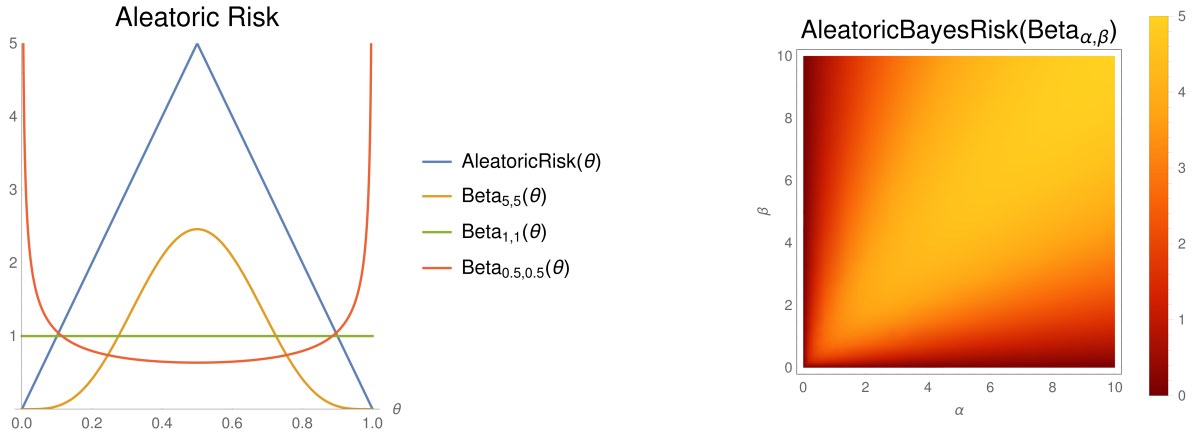


Figure 5.3: Refer to Problem 1 and Example 1. Left: The aleatoric risk over $\theta \in [0, 1]$ assuming $n = 10$ rounds, shown with three belief distribution densities. The aleatoric risk is highest for fair coins, for which each flip result is unpredictable, and lowest for a trick coin that always lands on one side. Right: The aleatoric Bayes' risk of belief distributions $b_t = \text{Beta}(\alpha, \beta)$ for various $\alpha, \beta \in \mathbb{R}_{>0}$. Observe that aleatoric Bayes' risk increases along the line $\alpha = \beta$ as $\alpha, \beta \to \infty$.

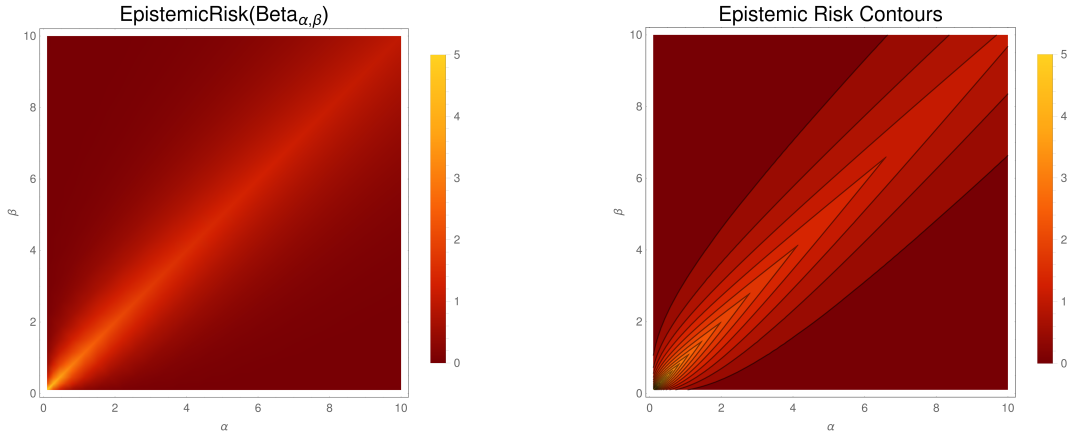


Figure 5.4: Refer to Problem 1 and Example 2. Left: The epistemic risk for various initial belief distributions, assuming $n = 10$ rounds. It is largest near the origin, for which the belief distribution is highly uncertain about which action is optimal. Right: The contours highlight that identifying the best action is most difficult when $\alpha \approx \beta$; however, in contrast to the aleatoric Bayes' risk (see Figure 5.3), the epistemic risk is asymptotically decreasing as $\alpha, \beta \to \infty$.

the process risk of a policy that sends some queries to a human supervisor measures the (negative) utility of those queries; if the process risk of that policy is near-zero, the robot could save time, communications bandwidth, and the human's effort by simply skipping those queries.

Definition 6. The **process risk** of a policy π in some belief state S_t is

$$\text{ProcessRisk}(\pi; S_t) \coloneqq \overline{\overline{r}}(\pi \| \hat{\pi}^\star_{S_t}; S_t).$$

Unlike aleatoric and epistemic risk, the process risk can be negative; specifically, the process risk is negative for any closed-loop policy that is able to, in expectation, leverage new observations to make better decisions than $\hat{\pi}^\star_{S_t}$. Suppose that the agent is following some policy which cycles through various actions, gradually identifying the hidden parameters. Proposition 1 shows that if the parameters are identifiable, the process risk of any other policy the agent might consider using becomes non-negative. This represents improvement in the ODOL policies; perhaps unsurprisingly, it indicates that once the agent has complete information about the hidden parameters θ, the posterior ODOL policy would perform at least as well as *any* closed-loop policy, including the optimal closed-loop policy.

Proposition 1. *Suppose θ are identifiable through some infinite sequence of actions. Over this sequence of actions, then $\forall \pi \in \Pi$, $\exists c_\pi \geq 0$ such that*

$$\text{ProcessRisk}(\pi; S_t) \xrightarrow[t\to\infty]{P} c_\pi. \tag{5.3.5}$$

Furthermore, the process risk of any policy is bounded below by the process risk of $\pi^\star_b$,

$$\text{ProcessRisk}(\pi; S_t) \geq \text{ProcessRisk}(\pi^\star_b; S_t)\ \forall \pi \in \Pi, \tag{5.3.6}$$

and so, over this same sequence of actions,

$$V^{\hat{\pi}^\star_{S_t}}(S_t) \xrightarrow[t\to\infty]{P} V^{\pi^\star_b}(S_t) \tag{5.3.7}$$

Proof. Eq. (5.3.5) follows from expanding $\overline{\overline{r}}(\pi \| \pi^\star_\theta; S) = \text{EpistemicRisk}(S) + \text{ProcessRisk}(\pi; S)$ and applying Lemma 1, followed by Theorem 1, and finally the continuous mapping theorem. Eq. (5.3.6) follows from expanding the definition of process risk and applying Lemma 1, and then Eq. (5.3.7) follows from Eq. (5.3.5) while noting that

$$\text{ProcessRisk}(\pi^\star_b; S_t) \leq \text{ProcessRisk}(\hat{\pi}^\star_{S_t}; S_t) = 0$$

Table 5.3: Definitions and interpretations of various endogenous Bayesian risk functions.

Risk Function	Notation	Key Determinant	Description
Aleatoric Bayes Risk	$\bar{\bar{r}}(\pi_\theta^\star \| \pi_X^\star; B_t)$	$P(x \mid \theta)$	Inherent OLP randomness
Epistemic Risk	$\bar{\bar{r}}\left(\hat{\pi}_{S_t}^\star \| \pi_\theta^\star; B_t\right)$	b_t	Uncertainty of b_t about θ
Process Risk	$\bar{\bar{r}}\left(\pi \| \hat{\pi}_{S_t}^\star; B_t\right)$	π	Relative risk of the policy π
Total Bayes' Risk	$\bar{\bar{r}}(\pi \| \pi_X^\star; B_t)$		Cumulative risk of the above

. □

Note that in Proposition 1, the sequence of actions taken is independent of the policy π for which the risk is computed. This proposition reinforces that ODOL policies improve over time as long as the belief distribution converges on θ; under such conditions, they asymptotically reach the performance of even the best closed-loop policy $\pi_b^\star$.

The aleatoric, epistemic, and process risks decompose the total risk according to

$$\text{TotalRisk}(\pi; S) \equiv \text{AleatoricBayesRisk}(S) + \text{EpistemicRisk}(S) + \text{ProcessRisk}(\pi; S). \quad (5.3.8)$$

A summary of the definition and interpretation of each risk is presented in Table 5.3. The process risk of a policy is the only component of the total risk that depends on the policy under consideration, π. Thus, a policy with less process risk in some state than another policy also has less total risk than that policy. While computing the process risk of a general policy is just as hard as computing that policy's value, computing the process risk of policies "similar" to the ODOL policy, such as lookahead policies, can be easier. This provides an efficient way to compare such policies without needing to explicitly calculate each policy's value or total risk. This observation will be leveraged in Section 5.4 to enable computationally efficient estimates of Bayes' risk, and to efficiently identify the best policy to follow from any particular belief state.

5.3.2 Computing Policy Value

Computing the risk of one policy relative to another is equivalent to comparing their respective values, and the main obstacle to computing the value of a policy is the size of its *reachable belief set*.

Definition 7. The **reachable belief set** (RBS) from an initial state S_0 is the set of all belief states that could result from some sequence of actions $A \in \bigcup_{t=1}^{\infty} \mathcal{A}^t$ and observations $Y \in \bigcup_{t=1}^{\infty} \mathcal{Y}^t$ (where $|A| = |Y|$).

Definition 8. The **π-reachable belief set** (π-RBS) is the smallest subset of the RBS that contains all belief states that a learning agent following the policy π, initialized at S_0, could eventually transition into.

Definition 9. We call the reachable belief set **unbounded** if $\forall h \in \mathbb{N}_{>0}$, there exist a sequence of actions $a_{1:h}$ that do *not* lead to a terminal state. Otherwise, the RBS is **bounded with decision horizon** h, where h is the length of the longest sequence of actions that results in a terminal state.

Whether the RBS is bounded or unbounded depends on the specification of ξ_0, δ, and Ω. Assuming discrete action and observation sets, the size of a bounded RBS with decision horizon h is $\mathcal{O}(|\mathcal{A} \times \mathcal{Y}|^h)$.[5] This rapid growth means it is generally impractical to compute the value of an arbitrary policy (where the π-RBS may match the full RBS) using MDP algorithms like value iteration; even describing an arbitrary policy requires $\mathcal{O}(|\mathcal{A} \times \mathcal{Y}|^h)$ values to encode the action distribution of the policy at every reachable state. Accordingly, we focus our analysis on policies which have a small π-RBS, or otherwise have some structure making it tractable to compute their value; in particular, this is the case for deterministic open-loop, m-lookahead, X-optimal, and θ-optimal policies, which will be explored in the following subsections.

Deterministic Open-Loop Policies

We begin by defining the set of all terminating feasible action sequences from $S_t = \{b_t, \xi_t\}$,

$$A^{\infty}_{\Omega}(\xi_t) = \left\{ A \in \bigcup_{n=1}^{\infty} \mathcal{A}^n \;\middle|\; \begin{array}{l} \xi_{t+\tau} = \delta(\xi_{t+\tau-1}, A(\tau)) \;\forall \tau \in \{1, \dots, |A|\}, \\ A(\tau) \in \Omega(\xi_{t+\tau-1}) \;\forall \tau \in \{1, \dots, |A|\}, \\ \Omega(\xi_{t+|A|}) = \varnothing. \end{array} \right\} \tag{5.3.9}$$

A deterministic open-loop policy $\hat{\pi} \in \hat{\Pi}$ takes a fixed sequence of actions $A^{\hat{\pi}}$ from state S_t until reaching a terminal state, so its value is

$$V^{\hat{\pi}}(S_t) \coloneqq \mathbb{E}_{X|b_t}\left[\gamma^{|A^{\hat{\pi}}|} R\Big(b^{X,A^{\hat{\pi}}}_{t+|A^{\hat{\pi}}|}\Big) + \sum_{\tau=1}^{|A^{\hat{\pi}}|} \gamma^{\tau} R\big(X(t+\tau), A^{\hat{\pi}}(\tau)\big) \right], \tag{5.3.10}$$

where $b^{X,A}_{t+|A|}$ denotes the posterior belief distribution of the terminal state reached after following the action sequence A with corresponding hidden outcomes X. Now, since the set

[5] For continuous action/observation sets, there is exponential growth in the *dimensionality* of the RBS.

of all feasible deterministic open-loop policies from the state S_t is isomorphic with $\mathcal{A}_\Omega^\infty(\xi_t)$,[6] the value of the ODOL policy $\hat{\pi}^\star_{S_t}$ is the solution to a maximization problem over $\mathcal{A}_\Omega^\infty(\xi_t)$,

$$V^{\hat{\pi}^\star_{S_t}}(S_t) = \max_{A \in \mathcal{A}_\Omega^\infty(\xi_t)} \mathbb{E}_{X|b_t}\left[\gamma^{|A|} R\left(b^{X,A}_{t+|A|}\right) + \sum_{\tau=1}^{|A|} \gamma^\tau R(X(t+\tau), A(\tau))\right]. \tag{5.3.11}$$

Eq. (5.3.11) is a non-linear discrete optimization problem which may be, in general, very challenging to solve. However, there are many conditions under which it is more tractable. For example, most classical bandit and partial monitoring problems are formulated with some given $T \in \mathbb{N}_{>0}$ such that $\Omega(\xi_t) = \mathcal{A},\ \forall t < T$ (and $\Omega(\xi_t) = \varnothing,\ \forall t \geq T$) and $R(b_t) = 0\ \forall b_t$. In such problems, Eq. (5.3.11) simplifies to

$$V^{\hat{\pi}^\star_{S_t}}(S_t) = \sum_{\tau=t+1}^{T} \max_{a_\tau \in \mathcal{A}} \gamma^\tau \mathbb{E}_{x|b_t}[R(x, a_\tau)], \tag{5.3.12}$$

where the $(T-t)$ unconstrained maximization problems can be solved independently. Subsection 5.3.3 will explore broader conditions under which Eq. (5.3.11) can be simplified and solved with suitable algorithms.

Lookahead Policies

An m-step lookahead policy π is one which operates in closed-loop for $m \in \mathbb{N}_{>0}$ actions before transitioning to the ODOL policy for the posterior belief state. This results in a π-RBS of size $O(|\mathcal{A} \times \mathcal{Y}|^m + |\mathcal{Y}|^{h-m})$. The value of an m-lookahead policy is equal to the sum of the expected action rewards over the closed loop phase and the total reward expected to be collected by the posterior ODOL policy from the posterior belief state. This fact is leveraged by the proposed EBRM algorithms, and will be discussed further in Section 5.4.

X-Optimal Policies

An X-optimal policy is deterministic and open-loop because the belief and action rewards are deterministic with respect to the sequence of actions taken by the agent. The conditional value of $\pi^\star_X$ in state S_t given X is thus the solution of the maximization problem, with $b^{X,A}_{t+|A|}$

[6] Consider that any $A \in \mathcal{A}_\Omega^\infty(\xi_t)$ uniquely characterizes a feasible deterministic open-loop policy, while any $\hat{\pi} \in \hat{\Pi}$ can be represented as a fixed sequence of actions $A^{\hat{\pi}}$, and is feasible if and only if $A^{\hat{\pi}} \in \mathcal{A}_\Omega^\infty(\xi_t)$.

defined as previously,

$$V^{\pi^\star_X}(S_t; X) \coloneqq \max_{A \in \mathcal{A}^\infty_\Omega(\xi_t)} \left[\gamma^{|A|} R\Big(b^{X,A}_{t+|A|}\Big) + \sum_{\tau=1}^{|A|} \gamma^\tau R(X(t+\tau), A(\tau)) \right]. \tag{5.3.13}$$

Subsection 5.3.3 will discuss conditions under which this problem can be further simplified.

Given a means to compute the conditionl value of the X-optimal policy for a particular realization of the hidden outcomes, its unconditional value is an expectation over the belief distribution b_t,

$$V^{\pi^\star_X}(S_t) = \mathbb{E}_{X|b_t}\big[V^{\pi^\star_X}(S_t; X)\big], \tag{5.3.14}$$

which can be computed with sampling-based methods, even if it cannot be solved analytically.

θ-Optimal Policies

A θ-optimal policy is generally closed-loop, as despite knowing the exact distribution from which the hidden outcomes are generated, maximizing the belief reward requires choosing actions based on how prior observations have shaped the belief distribution. However, under certain conditions on the belief reward function $R(b_t)$, there exists a deterministic open-loop θ-optimal policy. These conditions will be discussed in Subsection 5.3.3, and enable efficient computation of the conditional value function $V^{\pi^\star_\theta}(S_t; \theta)$. Given a means to compute this conditional value, the unconditional value of $\pi^\star_\theta$ is given by the expectation,

$$V^{\pi^\star_\theta}(S_t) = \mathbb{E}_{\theta|b_t}\big[V^{\pi^\star_\theta}(S_t; \theta)\big]. \tag{5.3.15}$$

5.3.3 Efficient Construction of Optimal Deterministic Open-Loop Policies

This subsection will explore approaches to efficiently solve Eq. (5.3.11) and construct an ODOL policy $\hat{\pi}^\star_{S_t}$ for the belief state S_t, under the following simplifying assumptions.

Assumption 1 (Bounded RBS)**.** We assume that the RBS is bounded with some decision horizon $h \in \mathbb{N}_{>0}$.

Assumption 2 (Discrete Action Set)**.** We assume that $\mathcal{A}$ is discrete and finite with cardinality $K \in \mathbb{N}_{>0}$. Without further loss of generality, we can assume that $\mathcal{A} = [K]$.

Assumption 3 (Order-Independent Feasibility)**.** We assume that, if $A \in \mathcal{A}^\infty_\Omega(\xi_t)$ and A' is a permutation of A, then $A' \in \mathcal{A}^\infty_\Omega(\xi_t)$. This is equivalent to the following conditions on Ω and δ:

1. $\Omega(\delta(\xi, a)) \subseteq \Omega(\xi) \quad \forall \xi, a : a \in \Omega(\xi)$

2. $a_2 \in \Omega(\delta(\xi, a_1)) \iff a_1 \in \Omega(\delta(\xi, a_2)) \quad \forall \xi, a_1, a_2 : \{a_1, a_2\} \subseteq \Omega(\xi)$

The key result of these assumptions is that the feasibility of a deterministic open-loop policy is determined by its *action counts* $N \in \mathbb{N}^K$, where an action sequence A is "consistent with N" if and only if

$$N(k) = \sum_{\tau=1}^{|A|} \mathbb{1}(A(\tau) = k) \quad \forall k \in [K]. \tag{5.3.16}$$

For any $N \in \mathbb{N}^K$, we can easily construct an action sequence A consistent with N by taking $N(1)$ copies of action 1, followed by $N(2)$ copies of action 2, and so on. If we let A' denote any permutation of A then, by Assumption 3, $A \in \mathcal{A}_\Omega^\infty(\xi_t) \iff A' \in \mathcal{A}_\Omega^\infty(\xi_t)$. We thus define $\mathbb{N}_\Omega^K(\xi_t) \subseteq \mathbb{N}^K$ such that

$$N \in \mathbb{N}_\Omega^K(\xi_t) \iff A \in \mathcal{A}_\Omega^\infty(\xi_t). \tag{5.3.17}$$

So, we can determine whether $N \in \mathbb{N}_\Omega^K(\xi_t)$ by constructing A and testing if $A \in \mathcal{A}_\Omega^\infty(\xi_t)$. Proposition 2 and its corollary will show that we can just as easily construct the *optimal* action sequence consistent with N. First, however, we show that the expected posterior belief reward following any action sequence N is a function of only the action counts, as seen in Lemma 2.

Lemma 2. *Let A be any action sequence consistent with $N \in \mathbb{N}_\Omega^K(\xi_t)$ from some initial state $S_t = \{b_t, \xi_t\}$, and let A' be any permutation of A. Then,*

$$\mathbb{E}_{X|b_t}\Big[R\Big(b_{t+|A|}^{X,A}\Big)\Big] = \mathbb{E}_{X|b_t}\Big[R\Big(b_{t+|A'|}^{X,A'}\Big)\Big], \tag{5.3.18}$$

where $b_{t+|A|}^{X,A}$ is the posterior belief distribution from the prior b_t following action-observation pairs $\{A(\tau), Y(\tau)\}_{\tau=1}^{|A|}$ where $Y(\tau) = \Phi(X(\tau), A(\tau))$ for each hidden outcome $X(\tau)$. For convenience, we thus define,

$$\bar{R}(N; b_t) \coloneqq \mathbb{E}_{X|b_t}\Big[R\Big(b_{t+|A|}^{X,A}\Big)\Big]. \tag{5.3.19}$$

Proof. Refer to A.4. □

Proposition 2. *Let $k_1, \ldots, k_K$ be an ordering of the elements of $\mathcal{A}$ such that, given S_t,*

$$\mathbb{E}_{x|b_t}\big[R(x, k_1)\big] \geq \mathbb{E}_{x|b_t}\big[R(x, k_2)\big] \geq \cdots \geq \mathbb{E}_{x|b_t}\big[R(x, k_K)\big]. \tag{5.3.20}$$

For any $N \in \mathbb{N}_\Omega^K(\xi_t)$, an optimal action sequence $A_N^\star \in \mathcal{A}_\Omega^\infty(\xi_t)$ consistent with N can be constructed by taking $N(k_1)$ copies of action k_1, followed by $N(k_2)$ copies of action k_2, and so on. The policy $\hat{\pi}_N^\star$ described by $A_N^\star$ satisfies,

$$V^{\hat{\pi}_N^\star}(S_t) \geq V^{\hat{\pi}}(S_t) \quad \forall \hat{\pi} : A^{\hat{\pi}} \text{ is consistent with } N. \tag{5.3.21}$$

Accordingly, we define the action count optimal value function,

$$\hat{V}(N; S_t) \coloneqq V^{\hat{\pi}_N^\star}(S_t) \tag{5.3.22}$$

$$= \gamma^{n_K} \bar{R}(N; b_t) + \sum_{i=1}^{K} \frac{\gamma^{n_{i-1}}\left(1 - \gamma^{N(k_i)}\right)}{1-\gamma} \mathbb{E}_{x|b_t}[R(x, k_i)], \tag{5.3.23}$$

where $n_0 \coloneqq 0$ and $n_i \coloneqq \sum_{j=1}^{i} N(k_j),\ \forall i \in [K]$.[7]

Proof. Follows from Lemma 2 and the monotonically non-increasing weight of later action rewards. □

Corollary 1. *The value of the ODOL policy $\hat{\pi}_{S_t}^\star$ in state S_t is*

$$V^{\hat{\pi}_{S_t}^\star}(S_t) = \max_{N \in \mathbb{N}_\Omega^K(\xi_t)} \hat{V}(N; S_t). \tag{5.3.24}$$

Corollary 1 means an exhaustive search for the ODOL policy can be done by evaluating the $O(h^K)$ elements of $\mathbb{N}_\Omega^K(\xi_t)$, rather than all $O(K^h)$ sequences in $\mathcal{A}_\Omega^\infty(\xi_t)$. This is often an improvement as, typically, $h \gg K$; however, it is still intractable for moderate to large values of h or K. The following subsection will explore conditions in which we can mor efficiently find the optimal action counts, through formulating it as an integer program. In any case, once $N^\star$ is found, an ODOL policy can be constructed by the approach used in Proposition 2.

Common Conditions for Integer Programming Solutions

The maximization problem in Eq. (5.3.24) is a non-linear integer program without a general polynomial-time solution. However, for many common feasibility criteria, reward functions, and discount factors, it can be further simplified to be efficiently solved or approximated by existing algorithms.

The first element to consider is the feasibility criterion; most integer programming algorithms permit only *linear* constraints on the optimization variable. This is equivalent to

[7] Note that $\lim_{\gamma \to 1}\left[\gamma^{n_{i-1}}\left(1 - \gamma^{N(k_i)}\right)(1-\gamma)^{-1}\right] = N(k_i)$.

requiring that there exist a weight matrix $W \in \mathbb{R}^{M\times K}$ and budget vector $c \in \mathbb{R}^M$ such that $N \in \mathbb{N}_\Omega^K(\xi_t) \iff WN \preceq c$. This requirement is satisfied for *temporal* and *knapsack* feasibility criteria, defined below.

Definition 10. A **temporal feasibility criterion** is one defined as, given some horizon $T \in \mathbb{N}_{>0}$,

$$\xi_0 = 0, \qquad \delta(\xi_{t-1}, a_t) = \xi_{t-1} + 1, \qquad \Omega(\xi_t) = \begin{cases} \mathcal{A}, & \xi_t < T \\ \varnothing, & \text{otherwise.} \end{cases}$$

Definition 11. A **knapsack feasibility criterion** is defined as, given a budget $c \in \mathbb{R}_{>0}$, a weight vector $w \in \mathbb{R}_{>0}^K$, and bounds $u_1, \ldots, u_K \in \mathbb{N} \cup \{+\infty\}$,

$$\xi_0 = \mathbf{0}^K, \qquad \delta(\xi_{t-1}, a) = \xi_{t-1} + \chi_a, \qquad k \in \Omega(\xi_t) \iff \begin{cases} \langle w, \xi_t \rangle + w(k) \leq c, \\ \quad \text{and } \xi_t(k) < u_k, \end{cases}$$

where the characteristic vector $\chi_k \in \{0,1\}^K$ satisfies $\chi_k(i) = 1 \iff i = k$.

Next, we consider how the action reward interacts with the feasibility criterion and discount factor. If action rewards are negative, the agent may be driven towards a terminal state as quickly as possible if the feasibility criterion permits it. As this may not be desirable behaviour in all cases, care must be taken when defining action rewards or shifting their values by a constant. Furthermore, negative action rewards can lead to the agent instead seeking *delays*, particularly if $\gamma < 1$; if taking an action with negative expected reward is required to satisfy the feasibility criterion, the optimal solution may precede that action with an arbitrarily long sequence of low or zero reward actions, in order to discount it.

Lastly, we consider how the belief reward interacts with the feasibility criterion and discount factor. As with action reward, if the belief reward can be negative, the agent may seek out or delay termination; as belief reward is earned only at termination, care must also be taken if actions can reduce the expected posterior belief reward. The problem is thus simpler if the belief reward function is non-negative and *adaptive monotone* [180]: $\forall b_t, \forall a,\ \mathbb{E}_{x|b_t}\left[R(b_{t+1}^{x,a})\right] \geq R(b_t) \geq 0$. Such belief reward functions are common, as belief rewards generally measure how much "information" has been learned about one or more of the hidden parameters; as the hidden parameters θ are fixed, taking an action never causes a learning agent to lose information. By Jensen's inequality, adaptive monotonicity is satisfied if $R(b)$ is convex in b; such belief rewards include the negative entropy of b, and the log generalized-precision $\log\det\Sigma^{-1}$, where Σ is the covariance matrix of b. Even with a non-negative, adaptive monotone belief reward function, termination-seeking behaviour may arise for a discount factor $\gamma < 1$; if the agent cannot increase its terminal belief reward by a factor of at least γ^{-1} with each action, it will seek out a terminal state as quickly as possible.

Based on the issues discussed above, we assume a non-negative adaptive monotone belief reward and either non-negative action rewards or a temporal feasibility criterion.[8] Given these assumptions, the choice of integer programming algorithm depends primarily on the discount factor and form of the expected posterior belief reward as a function of N. For example, efficient algorithms are known for $\bar{R}(N; b_t)$ that are concave and modular or submodular in N.[9] The contribution of action reward as a function of N is linear and modular when $\gamma = 1$, and generally $M^\natural$-*concave* [181] and submodular otherwise. A non-exhaustive list of algorithms for efficiently solving Eq. (5.3.24) given various combinations of feasibility criterion, belief reward, and discount factor is presented in Table 5.4. The approximation factor measures the ratio between the value of the policy found and the value of the true ODOL policy.

Applications to Solving θ- and X-Optimal Policies

The condition which most simplifies finding and solving the value of θ-optimal policies is an *observation-independent* belief reward, such that the reward assigned to a posterior belief distribution is a function of only the initial belief distribution and the actions taken. As a common example, if the posterior belief distribution b is normal with covariance matrix Σ, then a belief reward function satisfying $\exists f : R(b) = f(\Sigma)$ is observation-independent. Under this condition, finding the θ-optimal policy reduces to solving for action counts, so one of the solvers from Table 5.4 may apply.

While there is always a deterministic open-loop X-optimal policy for a given X, the main source of complexity in finding it and solving its value is, likewise, the belief reward function. A strong condition which simplifies finding and solving the value of X-optimal policies is a *separable* belief reward and a temporal feasibility criterion, such that $\exists f : R(b_{t+1}^{x,a}) = R(b_t) + f(x, a)$. The problem still cannot be reduced to solving for action counts, however it can be simplified into $O(h)$ univariate optimization problems.

5.4 Endogenous Bayesian Risk Minimization

We propose Endogenous Bayesian Risk Minimization (EBRM) as a general-purpose approach to optimal online learning. The goal of an EBRM policy is to, at each belief state S_t, find and imitate a policy that is optimal from that state. To make this approach computationally

[8] These assumptions are also driven by practical considerations as solvers for many classes of integer programs, such as knapsack problems, typically expect non-negative objectives.

[9] Submodularity is equivalent to only non-positive off-diagonal elements in the Hessian of $\bar{R}(N; b_t)$, while supermodularity is equivalent to only non-negative off-diagonal elements. A modular function is both submodular and supermodular.

Table 5.4: Solvers for Eq. (5.3.24) under common feasibility criteria and expected belief reward properties.

Feasibility Criterion	γ	$\bar{R}(N;b)$	Problem Class	Solver	Approx. Factor	Runtime Complexity
Temporal	$(0,1]$	$0, \forall N$	Concave Maximization	Greedy	1	$O(K)$
Temporal	1	Concave & Modular	Separable Concave Maximization	Iterative Greedy	1	$O(Kh)$
Temporal	$(0,1]$	M-Concave [182]	Integer M-Concave Maximization [182]	Steepest Ascent [183]	1	$O(K^2h)$
Temporal	$(0,1]$	Concave & Submodular	Monotone Submodular Maximization [184]	Iterative Greedy	$1-e^{-1}$	$O(Kh)$
Knapsack	1	Linear & Modular	(Un-)Bounded Knapsack [185]	Dynamic Program	1	$O(Kh)$
Knapsack	$(0,1]$	$0, \forall N$	Submodular Cost, Submodular Knapsack [186]	Iterative Greedy	$1-e^{-1}$	$O(Kh)$
Knapsack	1	Concave & Submodular				

tractable, we constrain our search to a small set of simple candidate policies $\Pi_t \subset \Pi$. In each state, the EBRM policy imitates the policy in Π_t with the maximum value,

$$\pi_{\text{EBRM}}(S_t) \coloneqq \arg\max_{\pi \in \Pi_t} V^{\pi}(S_t) \tag{5.4.1}$$

The set of candidate policies chosen characterizes the EBRM algorithm.

By imitating the candidate policy with the highest value in each decision step, the value of the best candidate policy is a lower bound on the value of the EBRM policy, and the Bayes' risk of the best candidate policy similarly upper bounds the Bayes' risk of EBRM. Thus *an EBRM policy accumulates, in expectation, more reward than any individual candidate policy would.*

5.4.1 Greedy-EBRM: EBRM using 1-Step Lookahead Policies

Greedy-EBRM is the simple yet highly effective online learning algorithm which results from choosing the candidate policy set composed of all deterministic 1-step lookahead policies,

$$\Pi_t = \{\pi_{t,a}\}_{a \in \mathcal{A}}, \tag{5.4.2}$$

$$\pi_{t,a}(S_t) = \text{Take action } a\text{, observe } y\text{, proceed according to the posterior ODOL policy.} \tag{5.4.3}$$

The value of a deterministic 1-step lookahead policy is therefore

$$V^{\pi_{t,a}}(S_t) = \mathbb{E}_{x|b_t}\left[R(x,a) + \max_{N \in \mathbb{N}_{\Omega}^{K}\left(\xi_{t+1}^{a}\right)} \gamma \hat{V}\left(N; S_{t+1}^{x,a}\right)\right], \tag{5.4.4}$$

where $\hat{V}(N; S_{t+1}^{x,a})$, defined in Eq. (5.3.24), is the value of the best open-loop policy for $S_{t+1}^{x,a} = \{b_{t+1}^{x,a}, \xi_{t+1}^{a}\}$ consistent with action counts N. The maximization problem is equivalent to the one in Eq. (5.3.24), but with a shorter decision horizon, so it can be calculated using the same algorithm that found the ODOL policy. However, there is generally no closed-form solution to the *expectation* of the maximum, and so the expectation often requires sampling-based methods to compute.

Fortunately, we can use the epistemic and process risk functions to help eliminate sub-optimal policies and obtain probabilistic bounds on the expectation in Eq. (5.4.4). First, we observe that

$$\pi_t^{\star} \in \arg\max_{\pi \in \Pi_t} V^{\pi}(S_t) \iff \pi_t^{\star} \in \arg\min_{\pi \in \Pi_t} \text{ProcessRisk}(\pi; S_t). \tag{5.4.5}$$

We then decompose the process risk of each $\pi_{t,a}$ into two terms: the *immediate risk* of a, which represents the "opportunity cost" of choosing action a instead of following the ODOL policy, and the *expected value of information* gained from the observation y which would be generated by the action a. The relationship between these components and the process risk is presented in 1, along with the complete specification of the Greedy-EBRM algorithm.

Algorithm 1: Greedy-EBRM

Given: OLP Specification
Input: Belief State S_t
$\texttt{InitialRisk}(S_t) \leftarrow \texttt{AleatoricBayesRisk}(S_t) + \texttt{EpistemicRisk}(S_t)$
foreach $a \in \mathcal{A}$:
  $\texttt{ProcessRisk}(\pi_{t,a}; S_t) \leftarrow \texttt{ImmediateRisk}(S_t, a) - \gamma \cdot \texttt{ExpectedVoI}(S_t, a)$
end foreach
$a^\star_{t+1} \leftarrow \arg\min_{a \in \mathcal{A}}[\texttt{ProcessRisk}(\pi_{t,a}; S_t)]$ `// Optimal action`
$U_t \leftarrow \texttt{InitialRisk}(S_t) + \texttt{ProcessRisk}(\pi_{t,a^\star_{t+1}}; S_t)$ `// Upper bound on Greedy-EBRM Bayes' risk`
return $a^\star_t, U_t$

We now define the immediate risk of an action a, which quantifies how much better the current ODOL policy is expected to perform than *the best deterministic open-loop policy that begins by picking* a. Equivalently, this measures how much additional regret is expected to be incurred over the ODOL policy by being forced to take action a as the next action. Importantly, both policies are constructed according only to the current belief state, without considering possible observations.

Definition 12. The **immediate risk** of performing an action $a \in \mathcal{A}$ in a belief state $S_t \in \mathcal{S}$ is defined as

$$\text{ImmediateRisk}(S_t, a) \coloneqq V^{\hat{\pi}^\star_{S_t}}(S_t) - \left(\mathbb{E}_{x|b_t}[R(x,a)] + \max_{N \in \mathbb{N}_\Omega^K(\xi^a_{t+1})} \mathbb{E}_{x|b_t}\left[\gamma \hat{V}\left(N; S^{x,a}_{t+1}\right)\right] \right). \tag{5.4.6}$$

The term on the right side of Eq. (5.4.6) represents the value of the best deterministic open-loop policy which begins by taking action a.[10] At least one of these policies is an ODOL policy for the state S_t; therefore, computing all K immediate risks requires computing the value of exactly K deterministic open-loop policies, and it holds $\forall S_t, \forall a$ that $\text{ImmediateRisk}(S_t, a) \geq 0$.

[10] Recall that, even if the ODOL policy $\hat{\pi}^\star_{S_t}$ would take action a at least once, it only begins with a if $a \in \arg\max_a \mathbb{E}_{x|b}[R(x,a)]$.

Next, the expected value of information of an action a, defined below, measures how much the value of an ODOL policy for the posterior state $S_{t+1}^{x,a}$ constructed with the additional observation $\Phi(x,a)$ surpasses, in expectation, one that is constructed without it.

Definition 13. The **expected value of information** gained by performing action a in belief state S_t is

$$\text{ExpectedVoI}(S_t,a) \coloneqq \mathbb{E}_{x|b_t}\left[\max_{N\in\mathbb{N}_\Omega^K\left(\xi_{t+1}^a\right)} \hat{V}\left(N;S_{t+1}^{x,a}\right)\right] - \max_{N\in\mathbb{N}_\Omega^K\left(\xi_{t+1}^a\right)} \mathbb{E}_{x|b_t}\left[\hat{V}\left(N;S_{t+1}^{x,a}\right)\right]. \tag{5.4.7}$$

While the expected value of information is driven by *uncertainty* in the belief distribution and the amount of *information* gained from an observation, it is measured in units of real reward (value). It is also closely related to epistemic risk, as shown in Lemma 3.

Lemma 3. *The expected value of information is bounded according to,*

$$0 \le \text{ExpectedVoI}(S_t,a) \le \text{EpistemicRisk}\left(\mathbb{E}_{x|b_t}\left[S_{t+1}^{x,a}\right]\right) - \mathbb{E}_{x|b_t}\left[\text{EpistemicRisk}\left(S_{t+1}^{x,a}\right)\right]. \tag{5.4.8}$$

Proof. Refer to A.5. □

The expected value of information represents the rate of convergence of the value of the ODOL policy to the value of the θ-optimal policy, as described in Theorem 1. Together, the expected value of information and immediate risk measure the process risk of any 1-step lookahead policy:

$$\text{ProcessRisk}(\pi_{t,a};S_t) = \text{ImmediateRisk}(S_t,a) - \gamma\cdot\text{ExpectedVoI}(S_t,a). \tag{5.4.9}$$

This relationship follows algebraically from Eqs. (5.4.4), (5.4.6), and (5.4.7).

The first useful result of this analysis, presented in Proposition 3, can enable Greedy-EBRM to disregard some candidate policies based on only their immediate risk.

Proposition 3. *If the immediate risk of an action $a\in\mathcal{A}$ exceeds the epistemic risk of the state $\mathbb{E}_{x|b_t}\left[S_{t+1}^{x,a}\right] = \{b_t,\xi_{t+1}^a\}$, then $\pi_{t,a}$ is not a minimizer of the process risk. That is,*

$$\text{ImmediateRisk}(S_t,a) > \gamma\text{EpistemicRisk}\left(\mathbb{E}_{x|b_t}\left[S_{t+1}^{x,a}\right]\right) \implies \pi_{t,a} \notin \arg\min_{\pi\in\Pi_t} \text{ProcessRisk}(\pi;S_t). \tag{5.4.10}$$

Proof. Refer to A.5. □

Applying the bound in Proposition 3 requires, for each action $a \in \mathcal{A}$, both the immediate risk of the action and the values of the ODOL and θ-optimal policies for the state $\mathbb{E}_{x|b_t}\left[S_{t+1}^{x,a}\right] = \{b_t, \xi_{t+1}^a\}$, in order to compute the epistemic risk bound. Then, estimating the process risk of each 1-step lookahead policy $\pi_{t,a}$ (excluding those eliminated from consideration under Proposition 3) can be done using sampling-based methods to estimate the expected value of information gained by action a. Using each sample to estimate the value of information requires finding and computing the value of an ODOL policy, as shown in Lemma 4.

Lemma 4. *Let $x_i \overset{\text{iid}}{\sim} \Pr(x \mid b_t)$ be independently and identically distributed hidden outcomes conditioned on the belief distribution b_t of state $S_t = \{b_t, \xi_t\}$, and define*

$$N_{t+1}^a \in \underset{N \in \mathbb{N}_\Omega^K(\xi_{t+1}^a)}{\arg\max} \mathbb{E}_{x|b_t}\left[\hat{V}\left(N; S_{t+1}^{x,a}\right)\right]. \tag{5.4.11}$$

The estimated value of information gained by action $a \in \mathcal{A}$ in this state, based on $n \in \mathbb{N}_{>0}$ samples, is

$$\nu_a(n) \coloneqq \frac{1}{n}\sum_{i=1}^{n}\left[\max_{N \in \mathbb{N}_\Omega^K(\xi_{t+1}^a)} \hat{V}\left(N; S_{t+1}^{x_i,a}\right) - \hat{V}\left(N_{t+1}^a; S_{t+1}^{x_i,a}\right)\right]. \tag{5.4.12}$$

By construction it holds that $\nu_a(n) \geq 0$, and by the linearity of expectation $\mathbb{E}[\nu_a(n)] = \text{ExpectedVoI}(S_t, a)$.

Following from Lemma 4, the value of information has finite variance if and only if $\exists \sigma_a \in \mathbb{R}$ such that

$$\mathbb{E}_{x|b_t}\left[\left(\max_{N \in \mathbb{N}_\Omega^K(\xi_{t+1}^a)} \hat{V}\left(N; S_{t+1}^{x,a}\right) - \hat{V}\left(N_{t+1}^a; S_{t+1}^{x,a}\right) - \text{ExpectedVoI}(S_t, a)\right)^2\right] = \sigma_a^2. \tag{5.4.13}$$

We further call the value of information bounded by $M_a \in \mathbb{R}$ if and only if

$$\Pr\left(\max_{N \in \mathbb{N}_\Omega^K(\xi_{t+1}^a)} \hat{V}\left(N; S_{t+1}^{x,a}\right) - \hat{V}\left(N_{t+1}^a; S_{t+1}^{x,a}\right) \leq M_a\right) = 1. \tag{5.4.14}$$

Assuming the value of information has finite variance, Theorem 2 provides probabilistic bounds on whether $\pi_{t,a} \in \arg\min_\pi \text{ProcessRisk}(\pi; S_t)$. These bounds depend on the number of samples used to estimate the value of information of each action. Corollary 2 shows how the number of samples can be picked to reach any desired level of confidence that a policy $\pi_{t,a}$ is sub-optimal, before rejecting it. Under the stronger condition that the value of information is bounded, Theorem 3 provides a lower bound on the number of samples required to bound

the expected *risk* of rejecting a policy $\pi_{t,a}$ to some value $\epsilon > 0$. Proofs for these results are provided in A.5.

Theorem 2. *Suppose that for each $a \in \mathcal{A}$, $n_a > 0$ samples have been used to estimate the expected value of information gained by action a in the belief state S_t. Then, define*

$$k \in \arg\min_{a'} \left[\text{ImmediateRisk}(S_t, a') - \gamma \cdot \nu_{a'}(n_{a'})\right], \tag{5.4.15}$$

$$\mu_a := (\text{ImmediateRisk}(S_t, a) - \text{ImmediateRisk}(S_t, k)) - \gamma(\nu_a(n_a) - \nu_k(n_k)). \tag{5.4.16}$$

If the values of information gained by actions a and k have finite variances σ_a^2 and σ_k^2, respectively, then

$$\mu_a > 0 \implies \Pr\left(\pi_{t,a} \notin \arg\min_{\pi \in \Pi_t} \text{ProcessRisk}(\pi; S_t)\right) \geq \frac{\mu_a^2 n_a n_k}{\gamma^2(\sigma_a^2 n_k + \sigma_k^2 n_a) + \mu_a^2 n_a n_k}. \tag{5.4.17}$$

Corollary 2. *It follows algebraically from Theorem 2 that, $\forall \epsilon > 0$ and $\forall a : \mu_a > 0$,*

$$n_k \geq \left\lceil \frac{2\sigma_k^2 \gamma^2 (1-\epsilon)}{\mu_a^2 \epsilon} \right\rceil, \quad n_a \geq \left\lceil \frac{2\sigma_a^2 \gamma^2 (1-\epsilon)}{\mu_a^2 \epsilon} \right\rceil \implies \Pr\left(\pi_{t,a} \in \arg\min_{\pi \in \Pi_t} \text{ProcessRisk}(\pi; S_t)\right) \leq \epsilon. \tag{5.4.18}$$

Theorem 3. *Suppose that the value of information gained by each action $a \in \mathcal{A}$ is bounded by M_a, and $n_a = n_k = n$. Then, following the notation of Theorem 2 and Corollary 2, by Hoeffding's inequality [187],*

$$\Pr\left(\pi_{t,a} \notin \arg\min_{\pi \in \Pi_t} \text{ProcessRisk}(\pi; S_t)\right) \geq 1 - \exp\left(\frac{-2n\mu_a^2}{\gamma^2 (M_a + M_k)^2}\right). \tag{5.4.19}$$

Under these assumptions, $\forall \epsilon > 0$ and $\forall a : \mu_a > 0$, if

$$n \geq \left\lceil \frac{\gamma^2 (M_a + M_k)^2}{2\mu_a^2} \ln\left(\frac{\gamma(M_a + M_k)}{\epsilon}\right) \right\rceil \tag{5.4.20}$$

then the expected risk of eliminating policy $\pi_{t,a}$ is bounded above by ϵ.

In practice, tighter bounds may be possible by taking into account the problem-specific structure of the action count value function $\hat{V}(N; S_t)$. It is worth noting that while the number of samples required to confidently reject a policy $\pi_{t,a}$ grows as $\mu_a \to 0$, small values of μ_a indicate that the expected difference in process risk (value) compared to the best reference policy $\pi_{t,k}$ is small, so the expected risk of rejecting $\pi_{t,a}$ may be small even if there is a non-trivial probability that it is optimal. Furthermore, even a single sample of the

expected value of information for each action is sufficient to provide an unbiased estimate of the best policy in the policy set. Thus, these results are best applied in settings where risk quantification is critical; in most cases, the number of samples is likely chosen based on the computational resources available.

5.4.2 Overcoming the Myopia of One-Step Lookahead Policy Sets

The expected value of information for 1-step lookahead policies cannot account for the value of information from *multiple* actions. Consequentially, these policies tend to have poor performance when multiple actions must be taken in order to produce any change in the epistemic risk; an excellent discussion of this issue with respect the Knowledge Gradient algorithm [17] for MAB problems is presented in [19]. We present a simple instance of the problem here.

Problem 2 (Apple Tasting)**.** *The apple tasting problem is characterized by* $\mathcal{X} = \{1, 2\}$*,* $\mathcal{A} = \{1, 2\}$ *and*

$$W = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad O = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix},$$

such that $R(x, a) = W(x, a)$ *and* $\Phi(x, a) = O(x, a)$*. Suppose* $x_t = 1$ *with probability* $\theta \in [0, 1]$*, and* $b_t(\theta) = \text{Beta}(\theta; \alpha_t, \beta_t)$*. If the agent takes action* 1*, it receives a binary observation that it can use to update its belief distribution according to*

$$\alpha_{t+1} = \alpha_t + y_{t+1}, \quad \beta_{t+1} = \beta_t + (1 - y_{t+1}).$$

However, if the agent takes action 2 *it receives the same observation regardless of* x_t*, and so gains no information. There is a temporal feasibility criterion and no belief reward or discounting.*

Example 3 (Failure of Greedy-EBRM)**.** The ODOL policy for Problem 2 is to always take action 1 if $\mathbb{E}[\theta] \geq 0.5$, and to otherwise take action 2. Suppose that at time t, the belief state is characterized by $(\alpha_t, \beta_t) = (1, 3)$, so $\mathbb{E}[\theta] = 0.25$. If the agent takes action 1, the possible posterior belief distribution parameters are $(2, 3)$ or $(1, 4)$; in either case, the posterior ODOL policy will be unchanged as it will still hold that $\mathbb{E}[\theta] < 0.5$. Similarly, if the agent takes action 2, its belief state is not updated and so the ODOL policy is unchanged. Therefore, as the ODOL policy in every possible posterior state is unchanged, the expected value of information from either action is zero; the agent will thus continue to select action 2 forever, gaining no new information and incurring linear regret.[11]

[11] Note that, as applied to Problem 2, Greedy-EBRM reduces to the Knowledge Gradient algorithm [17].

Accordingly, we present an asymptotic value of information approximation as a simple alteration to Greedy-EBRM that guarantees, over enough decisions, convergence on the optimal policy if the parameters θ are identifiable, as discussed in Theorem 1. This technique is discussed in the following subsection.

AsympGreedy-EBRM: Leveraging the Asymptotic Value of Information

The main limitation of Greedy-EBRM is that the expected value of information does not capture the value of information gained only from multiple actions. AsympGreedy-EBRM overcomes this issue by relying on a secondary, *asymptotic* measure of the value of information provided by some action. First, we introduce the Fisher information matrix $\mathcal{I}_a(\theta_0)$ corresponding to action a given the parameter estimates θ_0, which is defined as, with $D \coloneqq \dim \theta$ and $\ell(\theta \mid a, y) \coloneqq \log p(y \mid \theta, a)$,

$$[\mathcal{I}_a(\theta_0)]_{ij} \coloneqq \mathbb{E}_y\left[\left(\frac{\partial}{\partial \theta_i}\ell(\theta \mid a, y)\right)\left(\frac{\partial}{\partial \theta_j}\ell(\theta \mid a, y)\right)\middle|\theta = \theta_0\right], \tag{5.4.21}$$

$$i \in \{1, \dots, D\}, \tag{5.4.22}$$

$$j \in \{1, \dots, D\}. \tag{5.4.23}$$

Informally, the matrix $\mathcal{I}_a$ encodes how changes in the hidden parameters change the likelihood of the random variable y conditioned on an action a. There are cases where one or more hidden parameters do not play a role in the likelihood of the observation produced by a particular action; for example, in Problem 2, the hidden parameter does not affect the likelihood of the observation y for $a = 2$. In such cases, $\mathcal{I}_a$ is singular. However the *combined* Fisher information matrix,

$$\mathcal{I}_{\mathcal{A}}(\theta) \coloneqq \sum_{a \in \mathcal{A}} \mathcal{I}_a(\theta), \tag{5.4.24}$$

is non-singular if and only if the parameters are *identifiable*, as defined in Subsection 5.3.1. In fact, the Bernstein-von Mises theorem [188] implies that, if the parameters are identifiable and every action is taken at least $n \in \mathbb{N}_{>0}$ times, the belief distribution converges to a multivariate Gaussian such that

$$\lim_{n \to \infty} b_{K \cdot n}(\theta) \to \mathcal{N}\big(\theta; \theta^\star, n^{-1}\mathcal{I}_{\mathcal{A}}(\theta^\star)^{-1}\big), \tag{5.4.25}$$

where $\theta^\star$ is the true value of the hidden parameters.

Importantly, the Fisher information matrices $\{\mathcal{I}_a\}_{a \in \mathcal{A}}$ indicate which actions provide information, and which parameters they provide information about. Thus, even if the 1-step expected value of information for an action a is zero, a non-zero matrix $\mathcal{I}_a$ (even if

singular) indicates that the action still provides information about one or more parameters. In fact, the Bernstein-von Mises theorem implies the *asymptotic* value of the information. This concept is formalized in Theorem 4.

Theorem 4. *Let Σ_t denote the covariance of the belief distribution b_t and define $\hat{\theta}_t = \arg\max_\theta b_t(\theta)$. Suppose $\mathcal{I}_\mathcal{A}(\hat{\theta})$ is non-singular, $\Theta = \mathbb{R}^D$, and the regularity conditions of the Bernstein-von Mises theorem are satisfied [188].*[12] *As $t \to \infty$, if every action is taken at least $\lfloor t|\mathcal{A}|^{-1} \rfloor$ times, then the epistemic risk of state $S_t = \{b_t, \xi_t\}$ decreases in expectation according to*

$$\begin{aligned} &\text{EpistemicRisk}(S_t) - \mathbb{E}_{x|a}\big[\text{EpistemicRisk}(S_{t+1}^{x,a})\big] \\ &\qquad \xrightarrow[t\to\infty]{P} \text{EpistemicRisk}(\tilde{S}_t) - \text{EpistemicRisk}(\tilde{S}_{t+1}^a), \end{aligned} \tag{5.4.26}$$

$$\tilde{b}_t = \mathcal{N}\Big(\hat{\theta}_t, \Sigma_t\Big), \tag{5.4.27}$$

$$\tilde{b}_{t+1}^a = \mathcal{N}\left(\hat{\theta}_t, \left(\Sigma_t^{-1} + \mathcal{I}_a(\hat{\theta}_t)\right)^{-1}\right), \tag{5.4.28}$$

where $\tilde{S}_t = \{\tilde{b}_t, \xi_t\}$ and $\tilde{S}_{t+1}^a = \{\tilde{S}_{t+1}^a, \delta(\xi_t, a)\}$.

Proof. As the number of actions tends to infinity, the relative weight of the prior goes to zero, and so $\hat{\theta} \to \theta^\star$ and $\Sigma_t \to n^{-1}\mathcal{I}_\mathcal{A}(\theta^\star)^{-1}$; then, the theorem holds by application of Bernstein-von Mises theorem. □

We define the quantity in the right side of Eq. (5.4.26) to be the **asymptotic value of information**,

$$\text{AsymptoticVoI}(S_t, a) \coloneqq \text{EpistemicRisk}(\tilde{S}_t) - \text{EpistemicRisk}(\tilde{S}_{t+1}^a). \tag{5.4.29}$$

The asymptotic value of information calculation assumes that the parameters θ have approximately converged on their true value, and thus measures how the epistemic risk will change as the *uncertainty* of the belief distribution decreases while the expectation of the parameters stays constant. As long as the parameters are identifiable and the epistemic risk is non-zero, the asymptotic value of information will be strictly positive, as seen in Corollary 3.

Corollary 3. *If the parameters θ are identifiable, then*

$$\text{EpistemicRisk}(S_t) > 0 \implies \exists a \in \mathcal{A} : \text{AsymptoticVoI}(S_t, a) > 0. \tag{5.4.30}$$

[12] Among these, the one most of note is that $\forall\theta \in \Theta$, we require the prior to satisfy $b_0(\theta) > 0$.

Proof. By contradiction; if $\forall a$, $\texttt{AsymptoticVoI}(S_t, a) = 0$, then epistemic risk has converged. By Theorem 1, the epistemic risk must then be zero, since the parameters are identifiable. □

Corollary 3 implies that, for any OLP in which the parameters are identifiable, the asymptotic value of information will guide the agent towards complete knowledge of the parameters even if no single action is sufficient to gain useful information. Furthermore, the asymptotic value of information is measured in units of reward (value), and parallels the expected value of information in modelling how the epistemic risk changes as actions are taken. We therefore introduce, in Algorithm 2, AsympGreedy-EBRM, a modification of Greedy-EBRM for problems where the expected value information from any single action may be zero, even if significantly more information could be gained from a bit more exploration.

Algorithm 2: AsympGreedy-EBRM

Given: OLP Specification
Input: Belief State S_t
$\texttt{InitialRisk}(S_t) \leftarrow \texttt{AleatoricBayesRisk}(S_t) + \texttt{EpistemicRisk}(S_t)$
foreach $a \in \mathcal{A}$:
  $\texttt{ProcessRisk}(\pi_{t,a}; S_t) \leftarrow$
   $\texttt{ImmediateRisk}(S_t, a) - \gamma \cdot \max\{\texttt{ExpectedVoI}(S_t, a), \texttt{AsymptoticVoI}(S_t, a)\}$
end foreach
$a^\star_{t+1} \leftarrow \arg\min_{a \in \mathcal{A}}[\texttt{ProcessRisk}(\pi_{t,a}; S_t)]$ `// Optimal action`
$U_{t+1} \leftarrow \texttt{InitialRisk}(S_t) + \texttt{ProcessRisk}(\pi_{t,a^\star_{t+1}}; S_t)$ `// Upper bound on AsympGreedy-EBRM Bayes' risk`
return $a^\star_{t+1}, U_{t+1}$

Remark 1. *The assumption $\Theta = \mathbb{R}^D$ in Theorem 4 does not always hold; in this case, while the belief distribution still converges locally to a multivariate Gaussian, it may not be possible to compute the epistemic risk for a Gaussian belief. In such cases, a heuristic solution is for the agent to use the original belief distribution family for the asymptotic belief distributions, with $\tilde{b}_t = b_t$ and the parameters of $\tilde{b}^a_{t+1}$ chosen to maximize its similarity to $\mathcal{N}(\hat{\theta}_t, (\Sigma_t^{-1} + \mathcal{I}_a(\hat{\theta}_t))^{-1})$, such as by moment matching.*

Other Approaches

An alternative approach to overcome the myopia of 1-step lookahead policies is to consider the larger set of m-step lookahead policies, as for a sufficiently large m the best m-step lookahead policy is equivalent to the optimal closed-loop policy $\pi^\star_b$.[13] However the computational complexity of finding the best m-step lookahead policy is exponential in m. This large set can

[13] This occurs by $m = h - 1$, where h is, as previously, the decision horizon.

be reduced to only linear lookahead policies, which consider taking the same action multiple times but not combinations of actions; this strategy is used by the KG* algorithm [18], however it has been shown to remain overly myopic [19].

Alternatively, one may include other non-myopic policies in the candidate policy set. The EBRM solution outperforms any of the individual policies in the candidate policy set, so adding more policies generally improves the EBRM solution. The challenge of this approach is finding policies for which the value can be computed efficiently. In any case, the epistemic and immediate risk functions can be used to lower bound the process risk of any policy, even a stochastic one, as seen in Proposition 4.

Proposition 4. *The process risk, in the belief state $S_t \in \mathcal{S}$, of a policy π is bounded below as*

$$\text{ProcessRisk}(\pi; S_t) \geq \mathbb{E}_{a\sim\pi(S_t)}\big[\text{ImmediateRisk}(S_t, a) - \mathbb{E}_{x|b_t}\big[\text{EpistemicRisk}(S_{t+1}^{x,a})\big]\big]. \tag{5.4.31}$$

Proof. The process risk of an action is lower bounded by the regret incurred immediately by action a and the minimum possible process risk of any policy in the posterior belief state. The process risk of the posterior belief state $S_{t+1}^{x,a}$ is bounded below by $-\text{EpistemicRisk}(S_{t+1}^{x,a})$ (refer to Lemma 5 in A.5). □

5.4.3 Anytime-EBRM: AsympGreedy-EBRM for Unknown Time Horizons

Many classical OLPs are studied in the context of an *infinite* time horizon with no feasibility constraints (i.e. $\forall\xi, \Omega(\xi) = \mathcal{A}$), to make it easier to study the asymptotic behaviour of online learning algorithms. This is closely related to the case of an OLP that stops randomly, where algorithms are expected to minimize accumulated regret over *any* time horizon.

The AsympGreedy-EBRM algorithm cannot generally be implemented for an infinite time horizon without discounting, since it leads to infinite risk values. So, we introduce a modified *Anytime*-EBRM algorithm, which assumes that the remaining time is always equal to the current time (i.e., that it is always "halfway done"). Thus, at time t, it considers an artificial temporal feasibility criterion Ω_t such that,

$$\Omega_t(\xi) = \begin{cases} \mathcal{A}, & \xi < 2t+1, \\ \varnothing, & \text{otherwise.} \end{cases} \tag{5.4.32}$$

This provides a means to study the asymptotic behaviour of the AsympGreedy-EBRM approach and how it performs without knowledge of the time horizon. We compare Anytime-EBRM to the other EBRM algorithms and baselines in Section 5.5.

5.5 Experimental Results

In this section we explore the performance of EBRM algorithms across a range of common benchmark OLPs as well as some novel experiments that reflect real-world applications. Different OLPs are characterized by different specifications of the components listed in Table 5.1.

5.5.1 Bandit Optimization and Best-Arm Identification

Stochastic multi-armed bandits are a longstanding benchmark with which to evaluate online learning algorithms. The distinguishing features of stochastic MABs are presented in Table 5.5. We will begin by evaluating AsympGreedy-EBRM algorithms alongside popular online learning heuristics across a variety of stochastic bandits, described in Table 5.6, as well as both archetypal OLP goals of *bandit optimization* and *best-arm identification* as described in Table 5.7.

The defining characteristic of a bandit problem is "bandit feedback"; this refers to the observation function $\Phi(x_t, a_t) = [x_t]_{a_t}$, which indicates that after each action the agent is provided with only the hidden outcome corresponding to that action. The bandit optimization (BDO) goal is to choose actions that maximize the sum of these observations; this is a classic exploration-exploitation problem, as the agent must explore different actions before it can begin to identify and exploit the action with the largest observation mean. The metric for performance in bandit optimization is (exogenous) average expected *regret*,

$$t^{-1} \cdot \overline{\mathfrak{R}}(A_t^\pi; S_0, \theta) = \frac{1}{t} \sum_{\tau=1}^{t} \left(\max_{k \in [K]} \mathbb{E}_{x|\theta}[R(x,k)] - \mathbb{E}_{x|\theta}[R(x,a_t)] \right).$$

Lower average expected regret values are better; for algorithms which are asymptotically optimal at BDO, the average expected regret should tend to 0. The average expected regret represents how much larger the average reward of the θ-optimal policy is than the average reward of each online learning algorithm. For experiments in which $\theta \sim b_0$, the terminal regret of each algorithm is, in expectation, equal to the Bayes' risk of that algorithm relative to the θ-optimal policy from the initial state $S_0 = \{b_0, \xi_0\}$. Another popular metric is average *instance* regret, equal to the average expected regret plus the average aleatoric regret; the

Table 5.5: Multi-armed bandit problems (MAB) characteristics.

OLP Component	MAB Definition
Hidden Outcomes	$x_t \in \mathcal{X} \subseteq \mathbb{R}^K$
Actions	$a_t \in \mathcal{A} = \{1, \ldots, K\}$
Observations	$y_t \in \mathcal{Y} \subseteq \mathbb{R}$
Observation Fn.	$\Phi(x_t, a_t) = x_t(a_t)$

Table 5.6: OLP component specifications for various stochastic bandits.

	OLP Component	Gaussian Bandit	Bernoulli Bandit	Beta Bandit
θ	Hidden Parameters	$\mu \in \mathbb{R}^K$	$\mu \in [0,1]^K$	
	- Known Parameters	$\Sigma \in \mathbb{R}^{K \times K}$ (p.s.d.)	None	$\nu \in \mathbb{R}_{>0}$
f_θ	Process Model	$\mathcal{N}(\mu, \Sigma)$	Bernoulli(μ)	Beta($\mu\nu, (1-\mu)\nu$)
b_t	Belief Distribution	$\mathcal{N}\left(\hat{\mu}_t, \hat{\Sigma}_t\right)$	Beta(α_t, β_t)	
	- Belief Parameters	$\hat{\mu}_t \in \mathbb{R}^K$, $\hat{\Sigma}_t \in \mathbb{R}^{K \times K}$	$\alpha_t \in \mathbb{R}^K_{>0}$, $\beta_t \in \mathbb{R}^K_{>0}$	

Table 5.7: OLP specification for bandit optimization and best-arm identification.

OLP Component		Bandit Optimization	Best-Arm Identification
$R(x_t, a_t)$	Action Reward	x_{t,a_t}	0
$R(b_t)$	Belief Reward	0	$\max_k \mathbb{E}_{\mu \mid b_t}[\mu(k)] = \max_k \hat{\mu}_t(k)$ $\max_k P(\mu(k) > \mu_j \ \forall j \neq k \mid b_t)$
γ	Discount Factor	$(0, 1]$	N/A
$\Omega(\xi)$	Feasibility Criterion	Temporal with horizon $T \in \mathbb{N}_{>0}$	

aleatoric regret is, in expectation, the same for all algorithms in an experiment.

The goal of best-arm identification (BAI) is to identify the action with the largest observation mean, without regard to which actions are taken to do so. As seen in Table 5.7, success in this objective can be measured in two ways. The first is by $\max_k \hat{\mu}_t(k)$, which represents the mean action reward of the best action identified by time t [170]. This encourages identifying the action with the largest mean, and the penalty of identifying a different action is proportional to the difference in their respective action reward means. We will focus on the "Epistemic Uncertainty",

$$\mathbb{E}_{\mu|b_t}\left[\max_{k\in[K]} \mu(k) - \max_{k\in[K]} \hat{\mu}_k\right],$$

which is directly proportional to $\max_k \hat{\mu}_k$, and represents how much the observation mean of the true best action is expected to exceed the largest observed observation mean. As such, this metric converges to 0 as the agent becomes increasingly certain that no action has a larger observation mean than the best one it has identified.

Another popular metric of success in best-arm identification is the maximum probability for which any particular action is best [85]; this metric is intuitive and representative of many real-world problems, however it is difficult to optimize when the best $n > 1$ actions perform very similarly. In fact, this metric fluctuates asymptotically if the top n actions have equal observation means. As such it is poorly suited as a belief reward; we will, however, use it as an evaluation metric in some experiments. Since confidence values are often very close to 1, we display this metric in figures as "Log Uncertainty", computed as,

$$\ln\left(1 - \max_{k\in[K]} P(\mu(k) > \mu_j \ \forall j \neq k \mid b_t)\right).$$

AsympGreedy-EBRM Algorithms

All AsympGreedy-EBRM algorithms operate according to Algorithm 2; however, the immediate risk and value of information of each action depend on the OLP specification. The definitions of these functions for bandit optimization and best-arm identification objectives are presented in Tables 5.8 and 5.9. We show the corresponding functions for bandit optimization with Anytime-EBRM in Table 5.10. As a result of the choice of metric used for best-arm identification, epistemic risk for both tasks differ only by a scaling that depends on the time horizon; as such, the expected value of information and asymptotic value of information are also similarly defined for the two problems. The key difference is that in bandit optimization problems there is an immediate risk to taking an action with a lower

Table 5.8: Bayes-EBRM Algorithm Components.

EBRM Function	Bayes-EBRM
ImmediateRisk	$\max_k \hat{\mu}_t(k) - \hat{\mu}_t(a)$
EpistemicRisk	$(T-t)\mathbb{E}_{\mu\|b_t}[\max_k \mu(k) - \max_k \hat{\mu}_t(k)]$
ExpectedVoI	$(T-t-1)\mathbb{E}_{x\|b_t}\left[\max_k \hat{\mu}^{x,a}_{t+1}(k) - \max_k \hat{\mu}_t(k)\right]$
AsymptoticVoI	$(T-t-1)\left(\mathbb{E}_{\mu\|\tilde{b}_t}[\max_k \mu(k)] - \mathbb{E}_{\mu\|\tilde{b}^a_{t+1}}[\max_k \mu(k)]\right)$

Table 5.9: Epi-EBRM Algorithm Components.

EBRM Function	Epi-EBRM
ImmediateRisk	0
EpistemicRisk	$\mathbb{E}_{\mu\|b_t}[\max_k \mu(k) - \max_k \hat{\mu}_t(k)]$
ExpectedVoI	$\mathbb{E}_{x\|b_t}\left[\max_k \hat{\mu}^{x,a}_{t+1}(k) - \max_k \hat{\mu}_t(k)\right]$
AsymptoticVoI	$\mathbb{E}_{\mu\|\tilde{b}_t}[\max_k \mu(k)] - \mathbb{E}_{\mu\|\tilde{b}^a_{t+1}}[\max_k \mu(k)]$

Table 5.10: Anytime-EBRM Components for Multi-Armed Bandits (MABs).

EBRM Function	Anytime-EBRM
ImmediateRisk	$\max_k \hat{\mu}_t(k) - \hat{\mu}_t(a)$
EpistemicRisk	$(t+1)\mathbb{E}_{\mu\|b_t}[\max_k \mu(k) - \max_k \hat{\mu}_t(k)]$
ExpectedVoI	$t \cdot \mathbb{E}_{x\|b_t}\left[\max_k \hat{\mu}^{x,a}_{t+1}(k) - \max_k \hat{\mu}_t(k)\right]$
AsymptoticVoI	$t \cdot \left(\mathbb{E}_{\mu\|\tilde{b}_t}[\max_k \mu(k)] - \mathbb{E}_{\mu\|\tilde{b}^a_{t+1}}[\max_k \mu(k)]\right)$

observation mean, which does not apply to best-arm identification problems.

To distinguish between AsympGreedy-EBRM algorithms for bandit optimization and best-arm identification, we label the former as *Bayes-EBRM* and the latter as *Epi-EBRM*. This naming convention is based on how AsympGreedy-EBRM for best-arm identification is equivalent to a bandit optimization algorithm that chooses actions to minimize *only* the posterior epistemic risk (i.e., maximize value of information). Conversely, Bayes-EBRM chooses actions to minimizes the Bayes' risk by identifying the 1-step lookahead candidate policy with minimal process risk.

Baseline Algorithms

In order to provide a baseline against which to compare the EBRM algorithms, we evaluated several reference online learning algorithms. These algorithms are presented below, in groups distinguished by their design goals and whether they take the time horizon (for temporal feasibility criteria) into account.

General Online Learning Algorithms These algorithms, like the EBRM algorithms, can be applied to any OLPs which can described by the BMDP formulation presented in Section 5.2.

- *Thompson Sampling* (TS; also known as Posterior Sampling) [70]: Samples a set of hidden parameters θ from the belief distribution b_t, and then chooses a_{t+1} to be whichever action action is optimal (for bandit optimization) according to the sampled parameters. It can be easily extended to a wide range of OLPs by instead choosing the first action of the ODOL policy for the sampled parameters; we call this approach *Generalized TS*. Thompson sampling, first proposed in 1933, is remarkably effective given its simplicity [173], [189]. While classical Thompson sampling requires no tuning, recent works have added hyperparameters that can be tuned for better performance on specific bandit problems [177].

Best-Arm Identification Best-arm identification is one of the earliest fields of online learning, and is often called "pure exploration". Oftentimes, bandit optimization algorithms are used with their hyperparameters tuned to encourage exploration. We consider three dedicated BAI algorithms.

- *KG Explore* [89], [190]: Chooses the arm with the highest *knowledge gradient*. This "gradient" represents how much the largest posterior arm mean is expected to exceed the largest prior (current) arm mean, multiplied by the remaining time $(T - t)$. KG

Explore is the Greedy-EBRM algorithm for BAI, as this gradient is exactly equal the expected value of information in Eq. (5.4.7); unlike Epi-EBRM, KG Explore does not consider the asymptotic value of information.

- *KG* Explore* [18]: Improves upon KG Explore by computing the knowledge gradient based on repeating each action $m = \{1, \dots, T-t\}$ times, and chooses the m for which the knowledge gradient weighted by m^{-1} is largest. This is more effective heuristic, but expensive to compute for long time horizons [19].

- *Top-Two Expected Improvement* (TTEI) [85]: Operates by biasing sampling towards the two actions with the highest mean observations; the probability that the heuristic chooses the second-best action is controlled by a hyperparameter, but TTEI generally works well even when this hyperparameter is set to its default value. It has demonstrated superior performance when compared to various alternatives, including KG Explore [85].

Finite-Time Bandit Optimization These algorithms are designed to minimize regret in stochastic bandit optimization problems, and take into account the finite time horizon T.

- *UCB-MOSS* [191]: Chooses the action (arm) with the highest observation mean, with an added bias proportional to the remaining time horizon $(T-t)$ and inversely proportional to the number of times that action has been used so far. This bias is designed to be minimax optimal in stochastic bandit optimization problems with binary rewards.

- *Knowledge Gradient* (KG) [17]: The same as KG Explore, but adds the observation mean to the gradient. KG is the Greedy-EBRM algorithm for BDO and, unlike Bayes-EBRM, does not consider the asymptotic value of information.

- *KG** [18]: The same as KG* Explore, but adds the observation mean to the gradient. It still requires $O(Kh)$ computations for each decision, making it expensive to compute for long time horizons [19].

Asymptotic Bandit Optimization These algorithms are designed to incur the minimum worst-case regret in stochastic bandit optimization problems with unknown time horizons (including infinite horizons). Thus they are generally expected to underperform finite-time bandit optimization algorithms when T is given.

- *Double Sampling* (DS) [189]: A recent improvement upon Thompson sampling that makes modifications for a more efficient exploration-exploitation trade-off. It performs

similarly to TS when the probability of any particular action being optimal is low, but chooses the action with the highest observation mean when the probability of that action being optimal is high. DS uses a variable number of samples to make each decision, inversely proportional to its confidence in the best action.

- *UCB1* [81]: Perhaps the most popular heuristic for bandit optimization problems; as an upper confidence bound algorithm, it chooses the action with the highest observation mean, biased by an amount that is inversely proportional to the number of times that action has been chosen in the past. This bias is often multiplied by a scalar hyperparameter, which can be tuned for better performance on specific OLPs; a larger scalar encourages exploration over exploitation. Except where otherwise noted, we use the original UCB1 algorithm for which the value of this hyperparameter is 1.

- *Information Directed Sampling* (IDS/V-IDS) [19]: Chooses actions by minimizing the squared regret that the agent incurs per bit of information gained by the agent about the optimal action. This works well across a variety of bandit optimization problems, although it can be computationally expensive to compute the number of bits of information expected to gained by some action; variance-based IDS (V-IDS) instead uses a lower bound estimate of this quantity that is much more efficient to compute.

- Any-MOSS (MOSS-anytime) [192]: A variant of UCB-MOSS, this heuristic is designed to be minimax optimal for asymptotic bandit optimization problems.

Experimental Methodology

All trials used a temporal feasibility criterion with finite horizon $T \in \mathbb{N}_{>0}$. Each trial is characterized by a specific set of hidden parameters θ and hidden outcomes X. In some experiments, the hidden parameters are fixed, while in others they were randomly generated from the prior belief distribution b_0. Regardless, T hidden outcomes $x_1, \ldots, x_T$ were randomly sampled from the stochastic process f_θ. In each round $t = 1, \ldots, T$ of each trial, each algorithm chose one action a_t, and received the observation $\Phi(x_t, a_t)$. The same hidden outcome x_t was used to generate the observations and action rewards for all algorithms. The number of trials varied between experiments, as some experiments required more in order to achieve sufficiently low standard errors. All figures show 95% confidence intervals of the mean.

Results and Discussion

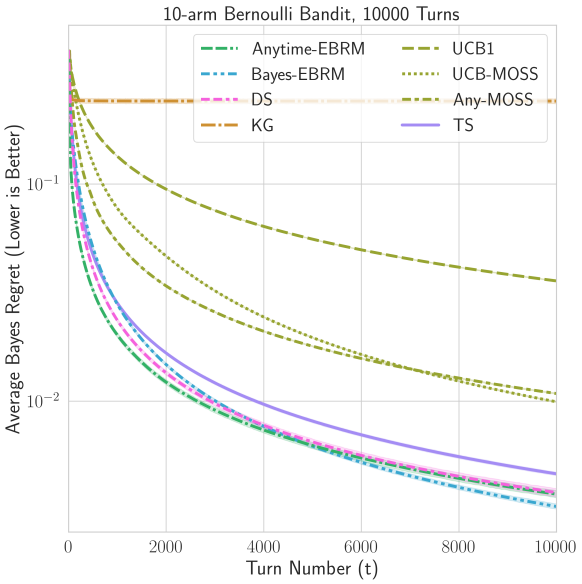
We evaluated the performance of the AsympGreedy-EBRM algorithms in a variety of numerical experiments, many of which are baselines established by or based on prior works.

One of the most comprehensive comtemporary comparisons of online learning algorithms for bandit optimization was made by Russo et al. [19]. Table 5.11 replicates their experiment consisting of a 10-arm Bernoulli bandit with hidden parameters sampled according to $\mu \sim b_0$, with $b_0 = \text{Beta}(1, 1)$, and a time horizon of $T = 1000$. The columns show the Bayes regret computed from 2000 trials, followed by various percentiles of Bayes regret. The algorithms which make use of the time horizon T are listed first, with a double line separating them from the algorithms designed for asymptotic performance.

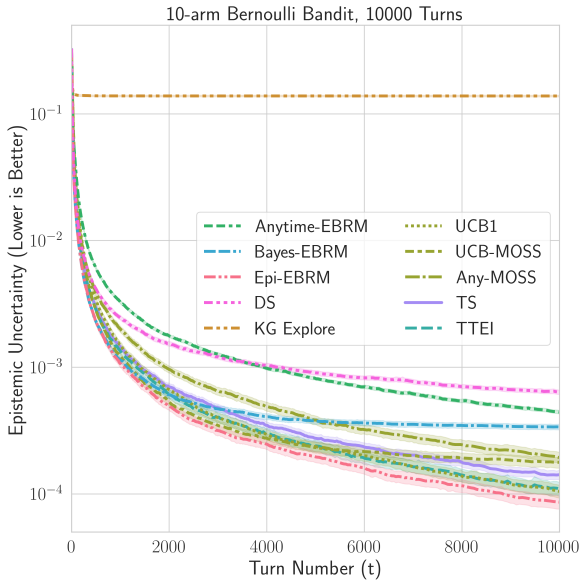
As Bernoulli bandits with beta priors are among the simplest and most well studied OLPs, it is unsurprising that Bayes-EBRM makes only a slight, although statistically significant, improvement upon the previous state-of-the-art for bandit optimization in this problem. More interestingly, it has more consistent performance than the other leading heuristics, including KG*, IDS, and V-IDS; this is particularly noticeable in the 90th and 95th percentile results. Anytime-EBRM performs comparably to IDS/V-IDS, demonstrating that the value of information computed by the AsympGreedy-EBRM approach is similarly useful to the IDS value of information function, and the performance improvement of Bayes-EBRM is mostly driven by taking the time horizon into account. The KG heuristic has been noted to perform particularly poorly for bandit problems with binary rewards [19].

In a similar Bernoulli bandit experiment with a time horizon of $T = 10^4$, shown in Figure 5.5, we find that Bayes-EBRM continues to outperform the other algorithms in bandit optimization. Similarly, for best-arm identification, Epi-EBRM achieves lower epistemic uncertainty than any baseline algorithm. This figure demonstrates aspects of the exploration-exploitation trade-off; in general, algorithms which focus on minimizing Bayes regret will explore less and have higher epistemic uncertainty [170]. However, for a finite-time horizon there is an optimal amount of exploration required to do well at bandit optimization; as seen in Figure 5.5, Bayes-EBRM spends approximately the first 4000 turns reducing its epistemic uncertainty, and performs worse on the bandit optimization objective than Anytime-EBRM or DS until this point; then, it surpasses them both by exploiting its more complete knowledge of the best action. The other algorithms all over-explore, with the UCB-based algorithms having orders of magnitude larger Bayes regret.

We next consider a 10-arm Gaussian bandit problem with time horizon $T = 1000$, and b_0 a zero-mean uncorrelated multivariate Gaussian distribution with unit variances. The results are presented in Table 5.12 and in Figure 5.6. This is another archetypal online learning



(a) Anytime-EBRM and DS over-exploit early in each trial, while Bayes-EBRM outperforms them by reaching the optimal level of epistemic uncertainty before exploiting the best arm.



(b) The stark contrast in BAI performance between Epi-EBRM and KG Explore highlights the importance of using the asymptotic value of information to escape bad local minima.

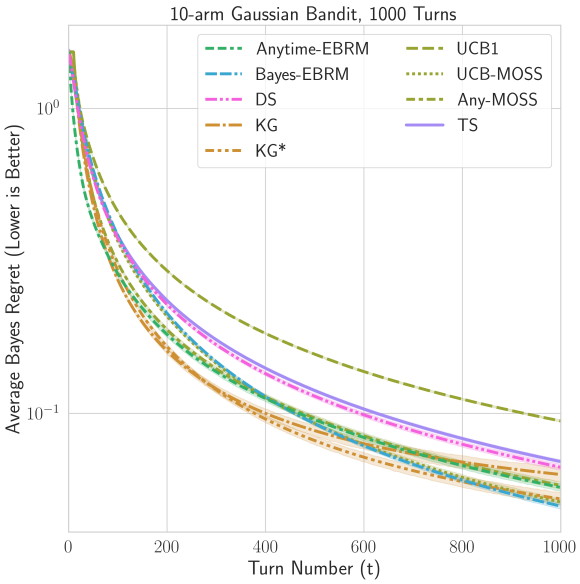
Figure 5.5: Performance of algorithms in BDO and BAI objectives while making decisions in a 10-arm Bernoulli bandit problem over 10^4 turns; lower values indicate better performance. Shaded regions indicate 95% confidence intervals over 5000 trials.

Table 5.11: Bayes Regret. 10-arm Bernoulli Bandit, 1000 Turns. Average over 2000 trials. [(1)]Results from [19].

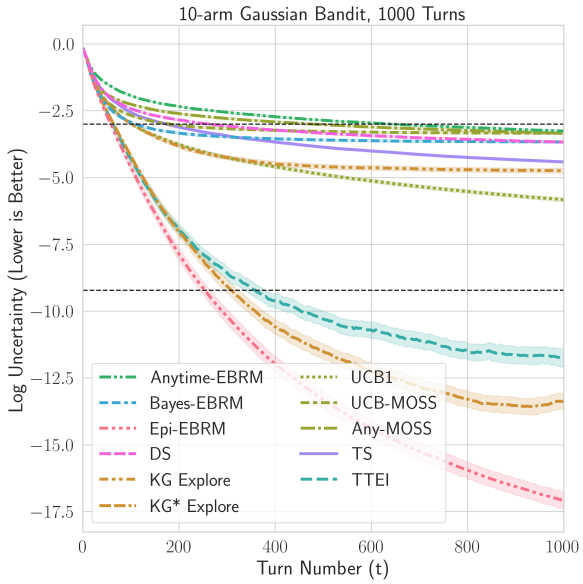
Algorithm	Mean	Percentiles					
		10%	25%	50%	75%	90%	95%
Bayes-EBRM	**17.2** ±0.3	6.9	9.0	13.2	20.1	30.7	40.6
UCB-MOSS	51.2 ±0.2	41.3	44.8	49.8	55.3	63.0	68.4
KG[(1)]	51.0 ±1.5	0.7	2.9	11.9	82.3	159.0	204.2
KG*[(1)]	18.4 ±0.6	2.9	5.4	8.7	16.3	46.9	76.6
Anytime-EBRM	19.8 ±0.4	3.6	8.6	15.9	24.7	37.9	51.4
IDS[(1)]	18.0 ±0.4	3.6	7.4	13.3	22.5	35.6	51.9
V-IDS[(1)]	18.1 ±0.4	5.2	8.1	13.5	22.3	36.5	48.8
DS	23.5 ±0.4	9.4	12.7	18.7	29.2	43.7	52.0
TS	28.4 ±0.3	13.1	17.5	25.1	35.3	47.9	56.1
Any-MOSS	53.9 ±0.2	43.7	47.7	52.1	57.9	64.5	71.6
UCB1	133.6 ±0.4	106.7	119.9	135.0	148.0	158.1	164.1

Table 5.12: Bayes Regret. 10-arm Gaussian Bandit, 1000 Turns. Average over 5000 trials. [(1)]Results from [19].

Algorithm	Mean	Percentiles					
		10%	25%	50%	75%	90%	95%
Bayes-EBRM	**49.6** ±0.6	28.4	34.0	42.1	53.8	69.1	85.5
UCB-MOSS	51.2 ±0.6	29.6	34.0	40.2	50.2	71.0	107.3
KG	63.0 ±1.7	16.3	20.3	25.8	35.7	141.9	303.5
KG*	52.4 ±1.3	18.6	23.3	29.4	39.4	79.0	196.3
Anytime-EBRM	57.2 ±0.8	24.7	30.9	41.6	60.5	98.8	145.6
V-IDS[(1)]	58.4 ±1.7	24.0	30.3	39.2	56.3	104.6	158.1
DS	66.5 ±0.8	31.2	40.6	54.8	75.5	105.9	136.2
TS	69.5 ±0.5	39.4	48.9	61.7	81.4	106.2	125.5
Any-MOSS	58.0 ±0.8	30.8	35.1	41.3	53.8	96.4	152.2
UCB1	94.4 ±0.4	64.3	74.5	90.3	109.5	129.7	143.7



(a) KG and KG* initially over-exploit the BDO objective, resulting in lower initial Bayes regret but ultimately performing worse than Bayes-EBRM and UCB-MOSS, which wait to be more confident in the best arm before exploiting it.



(b) The upper dashed line represents 95% confidence in having found the best arm, while the lower dashed line represents 99.99% confidence. Epi-EBRM achieves the highest confidence in the best arm over long horizons.

Figure 5.6: Performance of algorithms in BDO and BAI objectives while making decisions in a 10-arm Gaussian bandit problem over 1000 turns. Shaded regions indicate 95% confidence intervals over 2000 trials.

Table 5.13: Bayes Regret. 10-arm Gaussian Bandit, 10 time horizons. Average over 2000 trials. [(1)]Results from [19].

Algorithm	Time Horizon T									
	10	25	50	75	100	250	500	750	1000	2000
Bayes-EBRM	**9.1**	**14.8**	**19.7**	**23.1**	**24.9**	**33.5**	**40.8**	**45.4**	**46.6**	58.1
UCB-MOSS	15.3	21.2	26.2	29.2	31.1	38.0	43.2	46.0	49.6	**57.2**
KG	9.1	15.0	19.8	23.6	25.7	36.0	42.1	53.6	61.5	83.7
KG*	9.1	14.9	19.8	23.6	25.7	34.9	40.3	47.8	51.6	61.3
Anytime-EBRM	9.3	15.3	21.4	25.9	29.3	39.9	45.9	51.7	54.6	67.8
V-IDS[(1)]	9.8	16.1	21.1	24.5	27.3	36.7	48.2	52.8	58.3	68.4
DS	12.0	21.6	29.2	34.1	37.0	47.9	54.6	61.5	65.0	76.6
TS	12.1	21.8	29.8	34.7	37.9	49.4	58.2	64.3	67.8	80.6
Any-MOSS	15.3	20.9	25.5	28.9	30.9	40.1	47.0	51.7	56.3	67.3
UCB1	15.3	24.1	34.3	40.9	45.9	63.0	77.5	86.5	93.4	112.5

problem, and Bayes-EBRM again demonstrates leading performance by a small margin.[14] We also experiment with varying the time horizon as in [19]. Each of the values in Table 5.13 represents the mean average instance regret over 2000 trials, so the table represents 20000 trials for each of 9 algorithms. We observe that Bayes-EBRM is the only algorithm which performs consistently well across all time horizons; KG and V-IDS each perform similarly well to Bayes-EBRM for short time horizons, but begin to perform much worse from $T = 500$. Eventually, UCB-MOSS manages to outperform Bayes-EBRM, despite poor performance over short time horizons.

In best-arm identification on the 10-arm Gaussian bandit problem, we find that Epi-EBRM significantly outperforms the previous state-of-the-art beyond the first 100 turns. TTEI performs well up to this point, but its focus on the top-two arm candidates causes it to neglect the other 8 arms which may each still have a non-negligible chance of being the largest.

In A.2, we present the results of additional experiments involving a 2-arm Bernoulli bandit, a 5-arm Gaussian Bandit, and a 10-arm Beta bandit with time horizons of $T = 200$, $T = 100$, and $T = 10^4$, respectively. These results are qualitatively similar to the previous results, demonstrating that the AsympGreedy-EBRM algorithms match or surpass state-of-the-art performance in bandit optimization and best-arm identification regardless of the type of bandit, the number of arms, or the time horizon.

Hyperparameter Tuning

As noted in previous sections, some online learning heuristics have hyperparameters that can be tuned in order to achieve better performance on specific OLPs. NoTeS is a tuning algorithm designed specifically to optimize the hyperparameters of online learning algorithms in order to minimize Bayes' regret (risk) [177]. It is an iterative algorithm, which reports the best hyperparameters found by the time it reaches a user-defined tuning budget (number of iterations). It outperformed various baseline algorithms in being able to find the lowest Bayes' regret tuning in the smallest tuning budget [177].

Table 5.14 presents the average Bayes' regret of UCB1 and TS for 2- and 10-arm Bernoulli bandit problems with $T = 200$ and $T = 10^4$ respectively, alongside the results for Bayes-EBRM, Anytime-EBRM, DS, and UCB-MOSS. Bayes-EBRM and Anytime-EBRM outperform UCB1 and TS in both experiments, even when the latter are tuned over 1000 iterations. As usual, Bayes-EBRM demonstrates better performance than Anytime-EBRM by taking the time horizon into account.

[14] IDS is missing from this comparison as the authors note that it is too computationally expensive, and V-IDS achieves comparable performance [19].

Table 5.14: Bayes Regret, Bernoulli Bandits. Values averaged over 10^4 trials. *Results from [177].

Algorithm	2 Arms, 200 Turns, $\mu = [0.4 \quad 0.6]$					10 Arms, 10^4 Turns, $\mu(k) \sim \text{Beta}(1,1)$				
	Tuning Budget					Tuning Budget				
	Initial	50*	200*	1000*	Max*	Initial	50*	200*	1000*	Max*
UCB1	10.2	5.3	4.7	4.4	4.2	357.7	63.6	52.5	49.2	47.2
TS	5.5	5.1	4.8	4.6	4.3	46.3	74.9	42.9	36.4	33.8
Bayes-EBRM	**3.8**	—	—	—	—	**32.8**	—	—	—	—
Anytime-EBRM	4.2	—	—	—	—	37.2	—	—	—	—
DS	5.2	—	—	—	—	38.4	—	—	—	—
UCB-MOSS	7.4	—	—	—	—	99.4	—	—	—	—

Computational Decision Complexity Comparison

Most of the baseline algorithms are designed to be fast, and use $O(K)$ computations to compute some simple heuristic; often, these require taking a sample from b_t, or computing functions of its mean or covariance. The exceptions are KG*, which use $O(KT)$ computations, and DS, which uses a tunable number of samples from b_t for each decision. The EBRM algorithms require sampling to compute the expected value of information function for each action; in bandit problems, each action only provides information about one hidden parameter and constructing an ODOL policy from this information has cost $O(1)$, so the cost of generating a sample is $O(1)$. Thus, assuming a fixed number of samples, EBRM bandit decisions have a time complexity of $O(K)$. The actual decision time, however, depends heavily on the cost of each sample. In practice, for bandit problems we can accurately compute the expected value of information by using integration by quadrature at a relatively small, fixed number of sample locations in Θ.

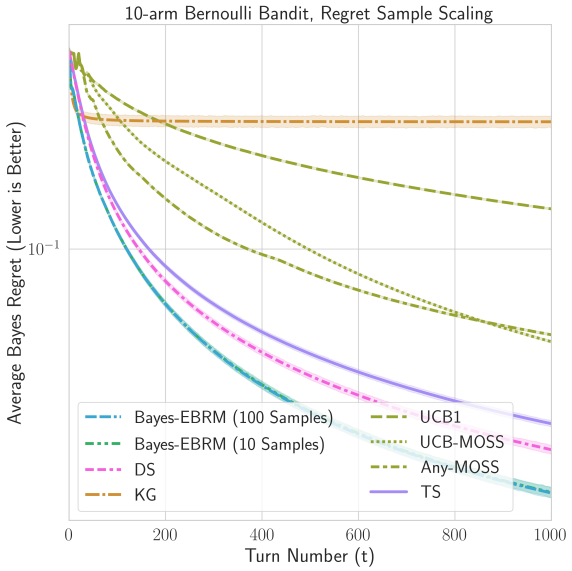
In Figure 5.7, we explore how the choice of the fixed number of samples used to estimate the expected value of information for each action affects the performance and running time of Bayes-EBRM. In general, we find that BDO performance is largely insensitive to the number of samples used. Furthermore, unlike KG*, the time for an EBRM algorithm to make a bandit decision is independent of the time horizon T.

5.5.2 Combined Belief- and Action- Rewards

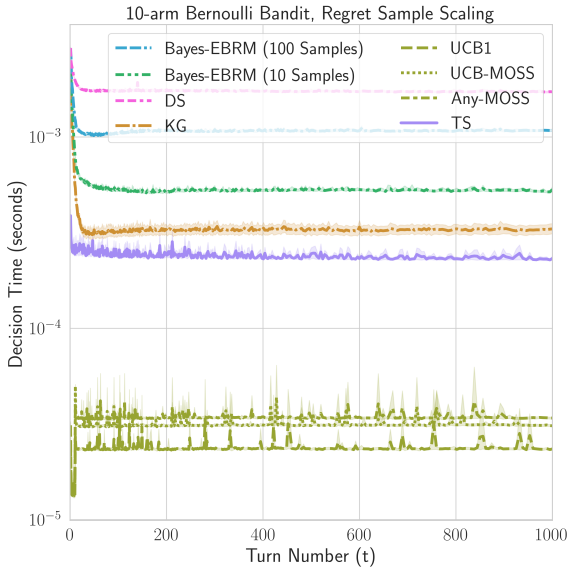
As noted in Section 5.1, and highlighted in Figure 5.1, online learning algorithms must balance between multiple competing OLP objectives. For all of the baseline algorithms discussed thus far, this balance is an implicit, fixed parameter of the algorithm. However, EBRM-based approaches are uniquely capable of achieving OLP objectives that can be expressed as the sum of action-based and belief-based rewards which meet the conditions described in Section 5.2. We demonstrate this by revisting the example in Figure 5.1.

Suppose there is a clinical trial to be conducted with a study group of 423 patients. The four treatments to be tested have *a priori* unknown success rates, $\mu_1 = \mu_2 = \mu_3 = 0.3$ and $\mu_4 = 0.5$. After the trial is completed, the best performing treatment will be administered to N patients outside of the trial who are likewise awaiting treatment. The goal is to have as many successful outcomes as possible across all $423 + N$ patients; the *Task Regret* represents the number of unsuccessful treatment outcomes. This example is adapted from the multi-arm trial setting considered in [71].

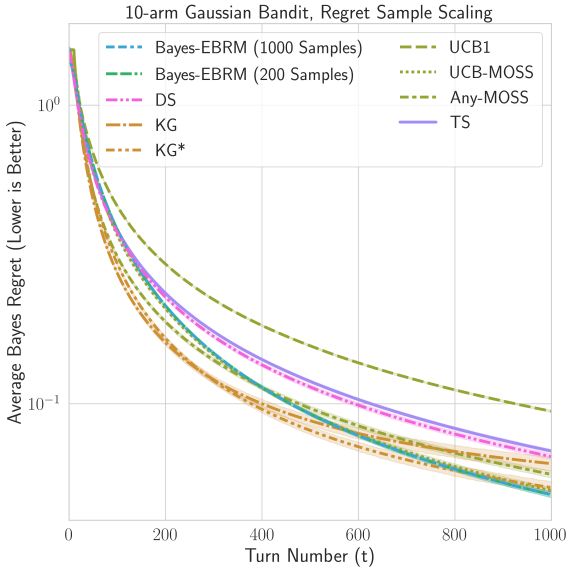
The AsympGreedy-EBRM algorithm for this problem is trivial to design, and presented in Table 5.15 as *Task-EBRM*. Note that in this example, epistemic uncertainty and instance



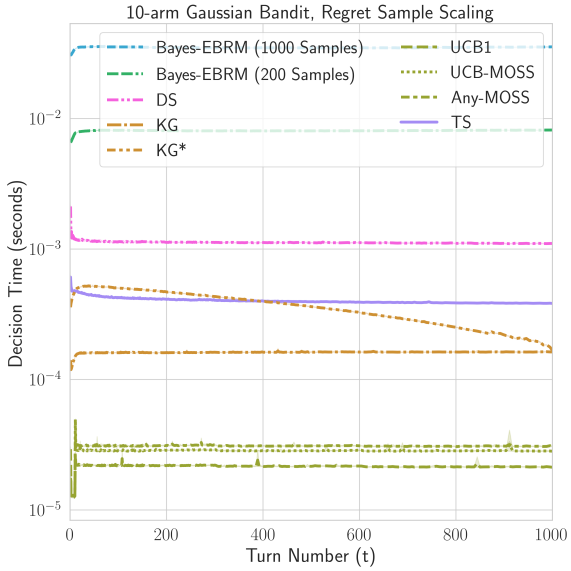
(a) The quality of Bayes-EBRM decisions with 100 samples is indistinguishable from its performance with only 10 in this Bernoulli bandit problem, resulting in a single blue-green curve.



(b) The Bayes-EBRM methods for this 10-arm Bernoulli bandit problem use comparable amounts of computation to the non-UCB baselines.



(c) The quality of Bayes-EBRM decisions with 1000 samples is indistinguishable from its performance with 200 in this Gaussian bandit problem, producing a single solid blue-green curve.



(d) Bayes-EBRM decisions with 200 samples require a fifth as much computation time as those with 1000 in this problem. We also see the proportionality of the KG* decision time to $T - t$.

Figure 5.7: We re-evaluate the performance of Bayes-EBRM for the Bernoulli and Gaussian bandit experiments (each with 10-arms and $T = 10^3$) with different numbers of samples used to estimate the expected value of information. We further report the average time for each algorithm to make a decision as a function of the turn number.

regret are equivalent to the x- and y-axis labels in Figure 5.1, respectively. This algorithm is parameterized by N, and directly attempts to maximize the total number of successful patient outcomes. For $N = 0$ Task-EBRM is equivalent to Bayes-EBRM, while as $N \to \infty$ its behaviour approaches that of Epi-EBRM. We evaluate this algorithm for various N alongside the baseline algorithms, and present the results in Figure 5.8.

As expected, the relative performance of most algorithms varies greatly for different values of N, while Task-EBRM consistently achieves the least task regret. For example, TTEI performs relatively poorly even when $N = 2500$, but is the best of the baseline algorithms as $N \to \infty$. Similarly, DS and Anytime-EBRM perform very well for $N \leq 500$, but perform poorly for $N \geq 2500$. UCB-MOSS and Any-MOSS are the most versatile of the baseline algorithms, but choosing either of these still results in up to three times as many unsuccessful patient outcomes as $N \to \infty$. In fact, by evaluating Task-EBRM at additional values of N we find that AsympGreedy-EBRM *dominates* the baseline algorithms, in that for any of the baseline algorithms, there is a value of $N \geq 0$ such that Task-EBRM simultaneously achieves *both* lower epistemic uncertainty *and* less Bayes' regret. This is presented in Figure 5.1, where the x-axis shows epistemic uncertainty scaled by 10^3. These results support that AsympGreedy-EBRM algorithms are the superior solutions for real-world OLPs.

5.5.3 Partial Monitoring and Dynamic Pricing

Partial monitoring is a more general class of online learning than bandit optimization, and enables the study of more interesting reward and observation models. Stochastic partial monitoring is a specific case of the OLP structure presented in Section 5.2, characterized by finite action and outcome sets with an "observation matrix" H and a "loss matrix" L, which define the observation model and action rewards, respectively. The full specification, less problem-specific components, is given in Table 5.16.

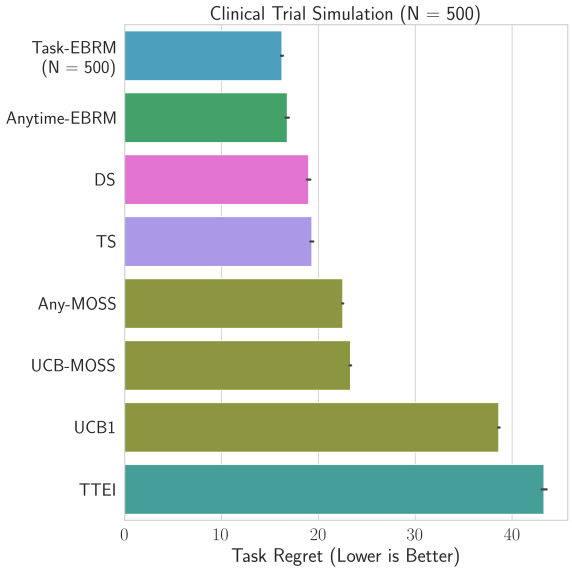
Dynamic Pricing is a prototypical partial monitoring problem which has complexities beyond that of bandit optimization. In it, each of the K actions represent a "sales price", and each of the $M = K$ hidden outcomes represents a customer's "willingness to pay". If the sales price is higher than a customer's willingness to pay, there is no sale and the agent incurs some fixed loss $c \in \mathbb{R}_{>0}$. Otherwise, a sale is made, and the agent incurs a loss based on how much lower the price was than the customer's willingness to pay. Importantly, the customer's willingness to pay is never directly revealed. The dynamic pricing specification,

Table 5.15: Task-EBRM Algorithm Components.

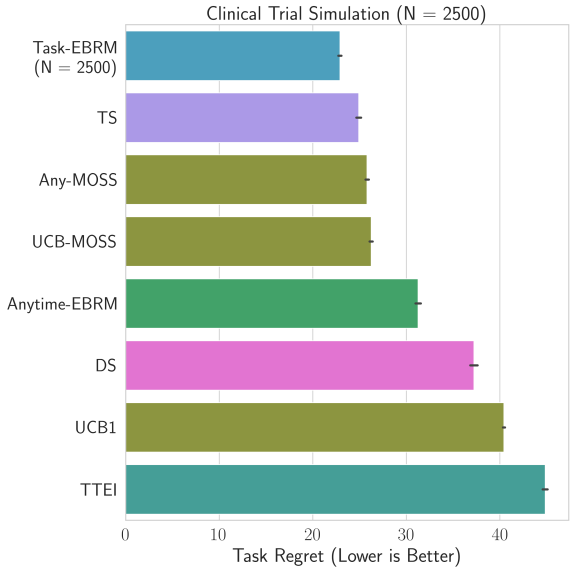
EBRM Component	Task-EBRM
ImmediateRisk(B_t, a)	$\max_k \hat{\mu}_t(k) - \hat{\mu}_t(a)$
EpistemicRisk(B_t)	$(N+T-t)\mathbb{E}_{\mu\|b_t}[\max_k \mu(k) - \max_k \hat{\mu}_t(k)]$
ExpectedVoI(B_t, a)	$(N+T-t-1)\mathbb{E}_{x\|b_t}\left[\max_k \hat{\mu}^{x,a}_{t+1}(k) - \max_k \hat{\mu}_t(k)\right]$
AsymptoticVoI(B_t, a)	$(N+T-t-1)\mathbb{E}_{\mu\|\tilde{b}^a_{t+1}}[\max_k \mu(k)] - \mathbb{E}_{\mu\|\tilde{b}_t}[\max_k \mu(k)]$

Table 5.16: Partial Monitoring Problem Characteristics.

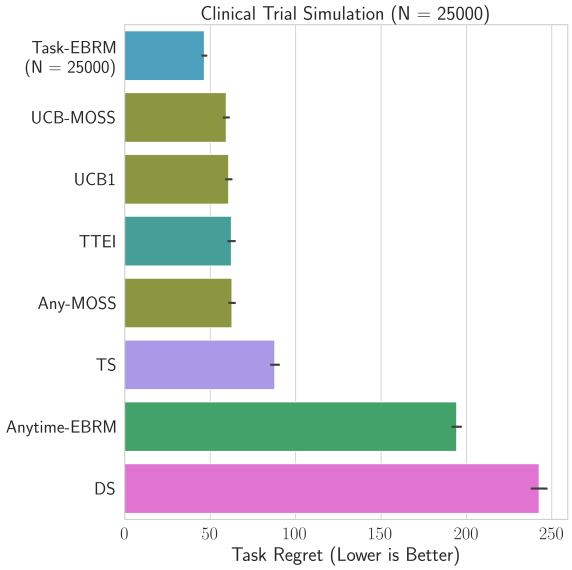
	OLP Component	Partial Monitoring Definition
$\mathcal{X}$	Hidden Outcomes	$\mathcal{X} = \{1, \ldots, M\}$
$\mathcal{A}$	Actions	$\mathcal{A} = \{1, \ldots, K\}$
Θ	Hidden Parameters	$\Theta = \Delta^M \coloneqq \{\theta \in \mathbb{R}^M_{\geq 0} : \|\theta\|_1 = 1\}$
f_θ	Process Model	$\Pr(x \mid \theta) = \theta(x)$
Φ	Observation Model	$\Phi(x, a) = H(x, a)$
	- Known Parameters	$H \in \mathcal{Y}^{M \times K}$
$R(x, a)$	Action Reward	$R(x, a) = -L(x, a)$
	- Known Parameters	$L \in \mathbb{R}^{M \times K}$
$R(b)$	Belief Reward	$R(b) = 0 \ \ \forall b$
$\Omega(\xi)$	Feasibility Criterion	Temporal with horizon $T \in \mathbb{N}_{>0}$



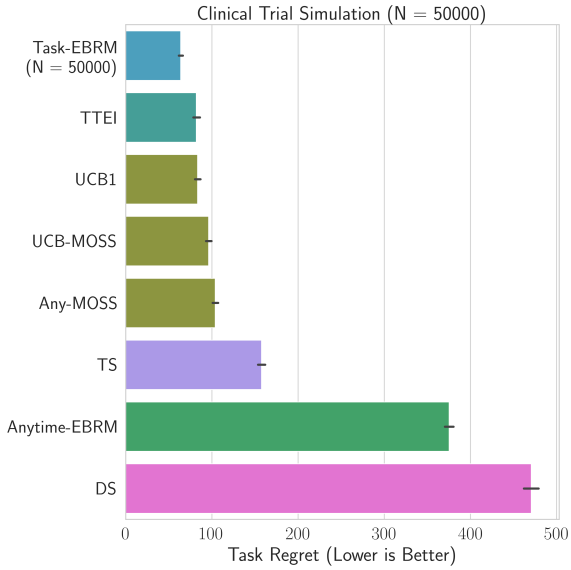
(a) At $N = 500$, we see similar relative performance as for the BDO objective in the Bernoulli bandit experiment from Figure 5.5, as Task-EBRM approximates Bayes-EBRM.



(b) At $N = 2500$, the lack of exploration performed by DS and Anytime-EBRM relative to Task-EBRM results in more unsuccessful outcomes among post-trial patients.



(c) At $N = 25000$, the large amount of exploration preferred by the UCB-based algorithms and TTEI begins to result in relatively strong performance.



(d) By $N = 50000$, we see similar relative performance as for the BAI objective in the Bernoulli bandit experiment from Figure 5.5, as Task-EBRM approximates Epi-EBRM.

Figure 5.8: Performance of the AsympGreedy-EBRM and baseline algorithms on the task described in 5.5.2, for various sizes of global patient population N. Task regret indicates the number of failed treatments among $423 + N$ patients.

of H and L is thus, as given in [193] and where "y" denotes a sale and "n" denotes no sale,

$$H = \begin{bmatrix} \text{y} & \cdots & \cdots & \text{y} \\ \text{n} & \text{y} & \cdots & \text{y} \\ \vdots & \ddots & \ddots & \vdots \\ \text{n} & \cdots & \text{n} & \text{y} \end{bmatrix}, \qquad L = \begin{bmatrix} 0 & 1 & \cdots & K-1 \\ c & 0 & \cdots & K-2 \\ \vdots & \ddots & \ddots & \vdots \\ c & \cdots & c & 0 \end{bmatrix}. \tag{5.5.1}$$

Dynamic pricing is more difficult than bandit optimization because the problem is not *locally observable*. In short, this means that the relative expected reward (equivalently, loss) of some pairs of actions a_1, a_2 cannot be determined without taking a third action a_3. This presents an issue if a_1, a_2 are candidates for the best action, while a_3 has much lower expected reward. In our formulation, this is a case where the (expected) value of information of various actions is highly correlated with their immediate risk, as the actions with low risk provide little or no information. A more thorough analysis of the issue, and its related implications to the difficulty of partial monitoring problems, is given by [75].

Letting $\mu = \langle -L, \theta \rangle$ and $\hat{\mu}_t = \mathbb{E}_{\theta|b_t}[\mu]$, the immediate risk, epistemic risk, EVoI and AVoI of the AsympGreedy algorithm for dynamic pricing match those of the Bayes-EBRM algorithm, as given in Table 5.8. The ODOL policy in any belief state is to choose the action with the highest expected reward, $\arg\max_{k \in [K]} \hat{\mu}_t(k)$.

Experimental Methodology

To generate each dynamic pricing problem instance, we generated a hidden parameter vector from a uniform distribution over the K-dimensional probability simplex Θ. The initial belief distribution, however, was taken to be a normal distribution with mean $\mathbb{E}[\theta]$ and an identity covariance matrix I_K. While this prior is unbiased, it is not the actual distribution from which hidden parameters are drawn. We generated 5000 problem instances for each experiment, and set the fixed cost for no sale to $c = 2$.

We use the BPM (Bayes-update Partial Monitoring) approach presented in [193] to generate a Gaussian posterior belief distribution following each new action-observation pair. Samples taken from these belief distributions may lie outside of the probability simplex Θ; in such cases, as in [193], we project these samples to the nearest point in Θ. We compare against the BPM-TS algorithm presented in [193], which is the generalized Thompson sampling strategy (discussed in Subsection 5.5.1) using the same BPM update rule.

Results and Discussion

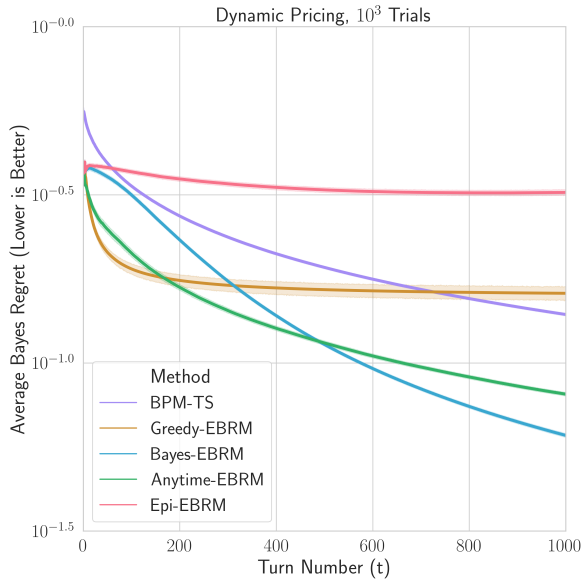
Figures 5.9 and 5.10 show average Bayes' regret and epistemic uncertainty for each of the

two experiments, with time horizons $T = 10^3$ and $T = 10^4$, respectively. The results parallel those of the bandit experiments; the Bayes- and Anytime-EBRM algorithms outperform the baseline by effectively managing the trade-off between exploration and exploitation, balancing immediate risk with the expected and asymptotic values of information. Knowledge of the time horizon enables Bayes-EBRM to outperform Anytime-EBRM, but the gap shrinks over longer horizons. The Greedy-EBRM algorithm, which ignores the asymptotic value of information, fails to sufficiently explore and incurs linear regret.

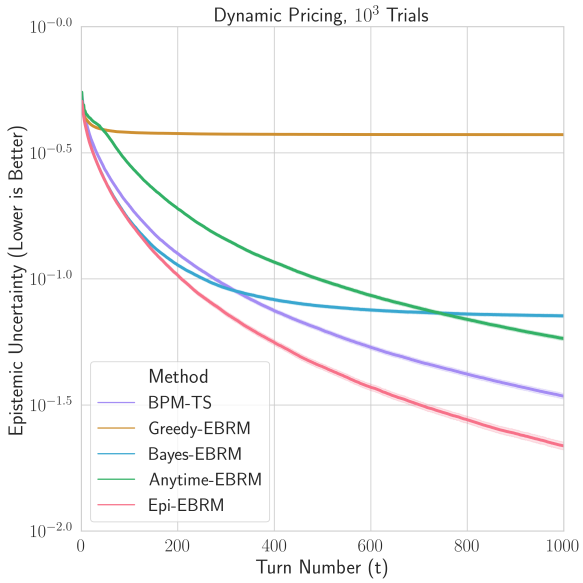
Impact of Prior Misspecification

As noted previously, the normal prior used in the dynamic pricing experiments does not match the true uniform distribution from which the hidden parameters are drawn. To determine the sensitivity of the algorithms to the choice of prior, we explored scaling the prior covariance matrix to sI_K, for some $s > 0$.

The results for $s \in \{2, 10, 0.5, 0.1\}$ are presented in Figure 5.11. In general, the most algorithms show little performance change over the wider priors $s \in \{2, 10, 0.5\}$, but performance degrades when $s = 0.1$. This is to be expected, as a wide prior can be compensated for by taking a few actions with high value of information, and any corresponding immediate risk incurred has little impact over longer time horizons. Conversely, a narrow prior can cause algorithms to underestimate the value of exploration and, in the case of EBRM, encourages following a sub-optimal ODOL policy.

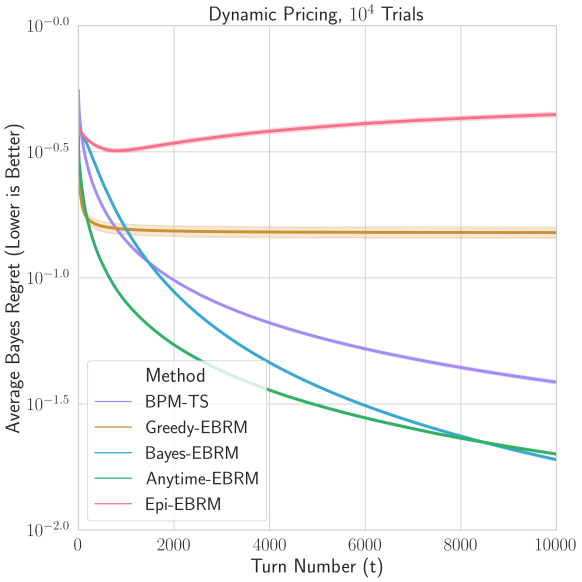


(a) The Anytime-EBRM algorithm excels at minimizing Bayes regret at the beginning of each experiment, but insufficient exploration causes it to fall behind for $t > 500$.

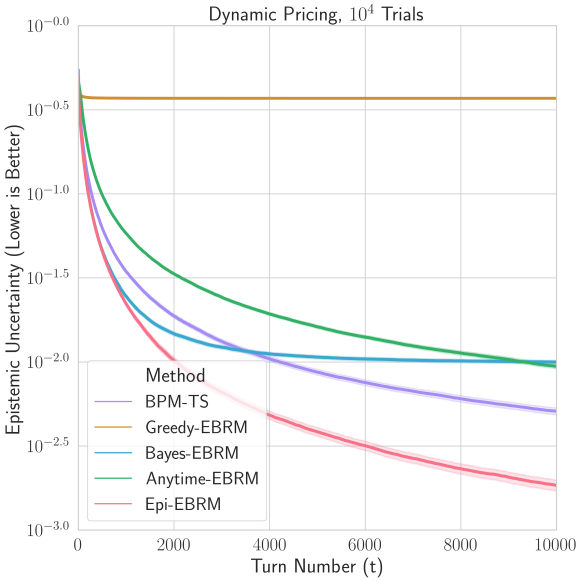


(b) By reasoning explicitly about the value of information, Epi-EBRM reduces the agent's epistemic uncertainty significantly faster than BPM-TS.

Figure 5.9: Results of the dynamic pricing experiment with $T = 10^3$. The Greedy-EBRM algorithm fails to sufficiently explore, suffering linear regret. The Anytime-EBRM algorithm, like BPM-TS, lacks knowledge of the time horizon but achieves superior performance by explicitly reasoning about immediate risks and the expected and asymptotic values of information. By further taking the time horizon into account, Bayes-EBRM significantly outperforms the other methods.

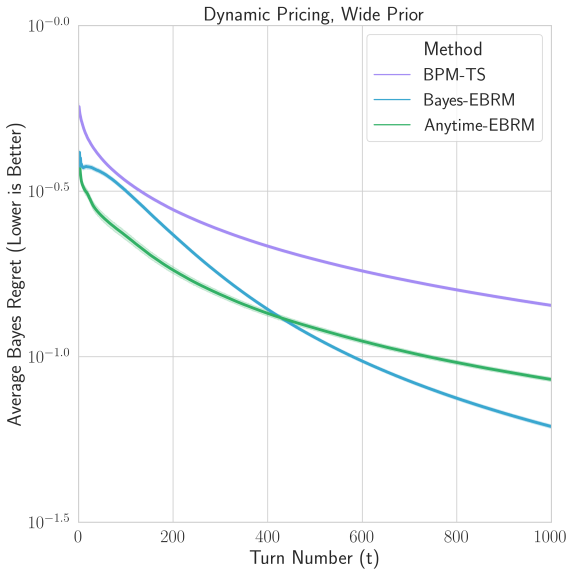


(a) While broadly similarly to the results in Figure 5.9(a), the difference in average Bayes regret between the Anytime-EBRM and Bayes-EBRM algorithms is reduced.

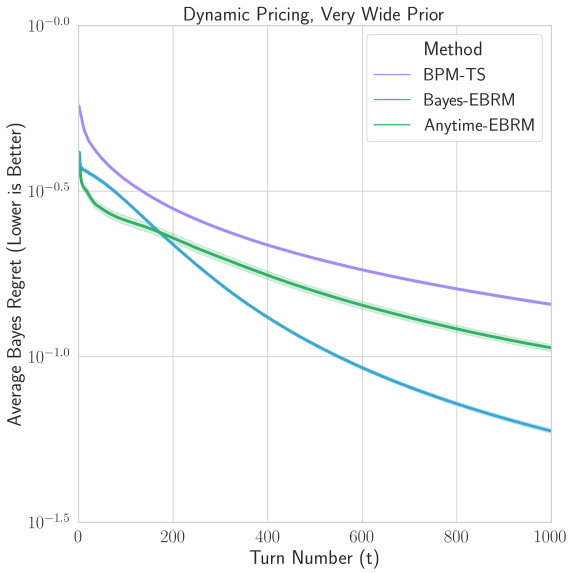


(b) While Epi-EBRM continues to most effectively reduce epistemic uncertainty, as the behaviour of BPM-TS gradually shifts away from exploration.

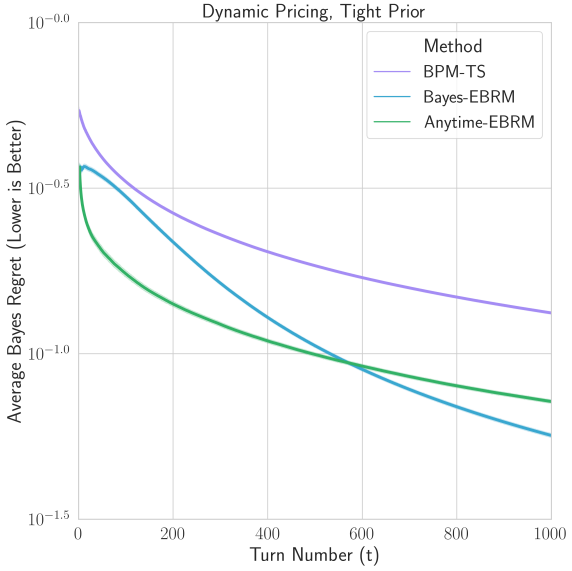
Figure 5.10: Results of the dynamic pricing experiment with $T = 10^4$.



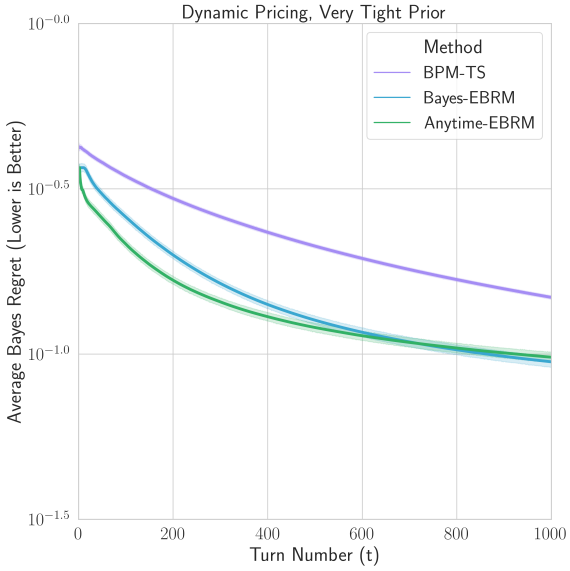
(a) At $s = 2$, we see very similar performance as for the baseline case $s = 1$ across all algorithms.



(b) At $s = 10$, Anytime-EBRM alone shows a noticeable increase in average regret.



(c) At $s = 0.5$, Anytime-EBRM and Bayes-EBRM each have a slight increase in performance.



(d) At $s = 0.1$, Anytime-EBRM and Bayes-EBRM are both negatively impacted.

Figure 5.11: Performance of the EBRM algorithms and BPM-TS on the dynamic pricing task for various priors, differing only in the scale of their respective covariance matrices, s. In general, "wider" priors result in better performance than overly "tight" or "narrow" priors, which can cause the algorithms to under-explore.

5.6 Summary

The BMDP model in presented in Section 5.2 provides a standard way to model online learning problems with combined action- and belief-based rewards, action-based costs and various feasibility criterion. The notion of measuring risk with respect to X- and θ-optimal policies as well as ODOL policies presents new ways to understand online learning problems and analyze policies through aleatoric, epistemic, and process risks. The EBRM-approach of searching for policies with minimal process risk has been shown to be feasible and highly effective at solving bandit problems, with AsympGreedy-EBRM algorithms matching or exceeding the state of the art in every experiment. The proposed approach is unique in that deriving the immediate risk and value-of-information functions for a particular online learning problem characterizes the AsympGreedy-EBRM algorithm for that problem.

The EBRM approach presented in this chapter represents a change in direction from previous online learning algorithms, which were each designed for optimal performance over some archetypal class of online learning problem and objective, to the design of an algorithm that is *parameterized by* the online learning problem specification itself. While more complicated, this ensures that the behavior of an EBRM algorithm is always aligned with the task goals, and eliminates the need for hyperparameters. As such, EBRM approaches are a highly effective compromise between POMDP solutions like reinforcement learning, which can optimally solve complex online learning problems but often have significant setup and computational costs, and online learning heuristics, which are easy to implement and compute but cannot capture the complexities of specific online learning problems.

The advantages of AsympGreedy-EBRM over previous methods are particularly valuable for use in our proposed approach to autonomous exploration from Chapter 1. Query actions (which represent sending image to a human supervisor for labeling via some underwater communications device) have energy costs and feasibility constraints, which EBRM algorithms are able to take into account. Furthermore, as observed in Section 5.5, myopic risk-based online learning algorithms are particularly prone to failure in cases of binary feedback (see, e.g., KG in Figure 5.5), which is the case when requesting human labels for phenomena of interest; by overcoming this myopia, AsympGreedy-EBRM achieves leading performance in such problems. Lastly, by explicitly estimating the expected/asymptotic value of information, a robotic explorer can weigh the utility of a query against the time and effort of the human supervisor. Specifically, by assigning a "cost" for the supervisor's time to query actions, and assigning zero cost to skipping a query, queries will only be made if they are sufficiently useful. Ultimately, this improves human-robot collaboration, as it avoids wasting human time and energy, as well as communications bandwidth, on low-value queries.

Chapter 6

Adaptive Exploration with Risk-Minimizing Communication

This chapter presents a system for vision-based robotic exploration of unfamiliar environments with severe communication bandwidth restrictions ($\lesssim 10$ kbps) and *a priori* unknown targets of interest, addressing the challenges first discussed in Chapter 1. The robot iteratively selects images to send to its human supervisor using the AsympGreedy-EBRM algorithm presented in the previous chapter; the supervisor labels those images based on whether they contain targets of interest. The robot models the co-occurrence of these targets with various semantic classes, learned online via unsupervised spatiotemporal semantic models. To do so, it uses a novel *beta-concentration* probabilistic reward model, which infers the concentration of targets of interest associated with each semantic class and can be updated efficiently with new labels. The robot simultaneously extrapolates its semantic map in order to predict where to find more targets, and uses a hotspot-based greedy planner to cover the regions where the concentration of targets of interest is expected to be the highest. This approach was evaluated in simulations using large maps of real coral reefs constructed from data collected in the U.S. Virgin Islands.

Summary of Contributions

In this chapter we demonstrate what is, to our understanding, the first complete system for human-in-the-loop adaptive visual exploration with severe communications restrictions. In constructing this system, we integrate and build upon recent developments in unsupervised semantic mapping, online learning, and topological path planning. Specifically, we present the following contributions:

1. An EBRM-based query selection algorithm which optimizes the use of a severely band-

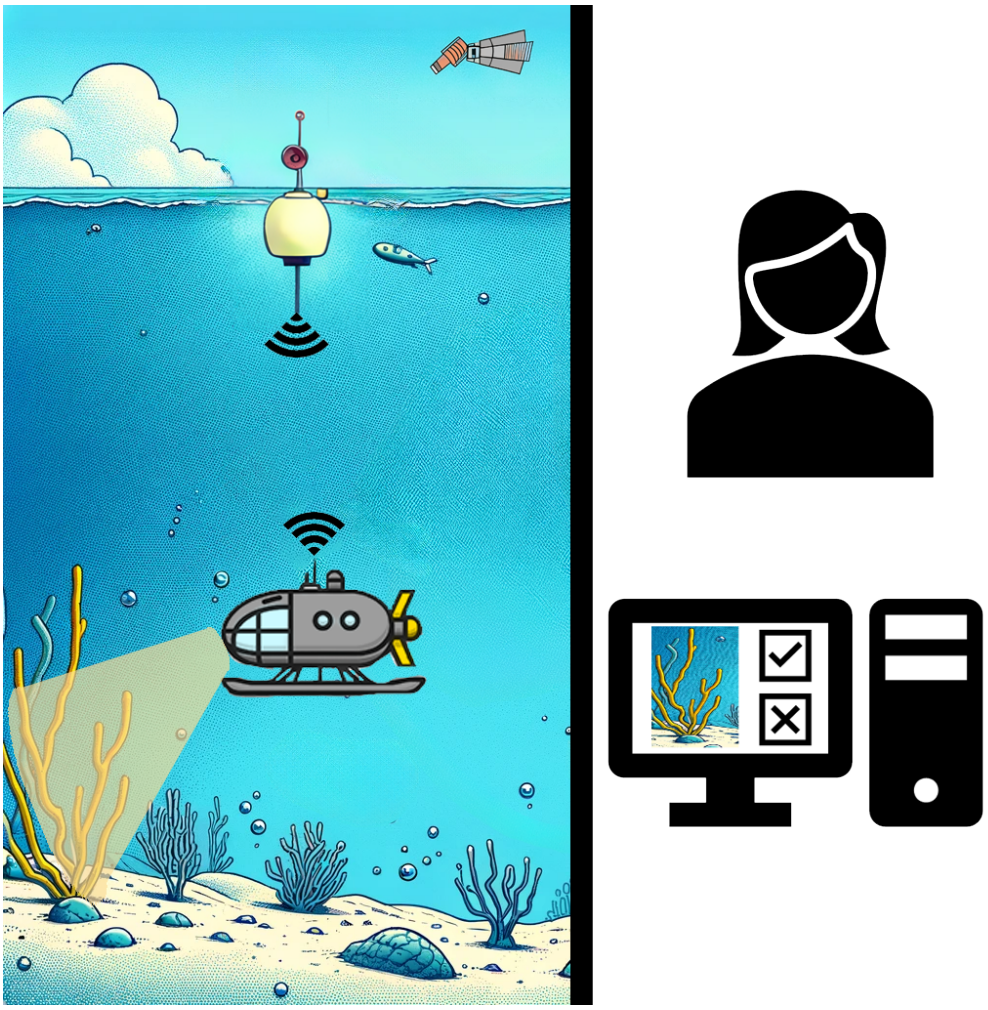


Figure 6.1: We present an approach to maximally utilize a low-bandwidth communications channel between a robotic explorer (left) and a remote human supervisor (upper right) for realtime mission adaptation. The robot uses a risk-aware online learning algorithm to pick individual images to send to the supervisor, carefully chosen such that they are representative of the various observation classes encountered and can be used to augment the mission plan in realtime, based on the supervisor's feedback (lower right), for more targeted observation. Note: Parts of this figure were generated using DALL·E 3.

width limited communications channel for *in situ* mission adaptation,

2. The novel Beta-Concentration reward model, which provide interpretable mappings from learned semantic distributions to mission objectives and can be efficiently updated with new labels,

3. A novel two-stage hotspot-based path planner which uses an initial sparse survey to enable semantic map extrapolation and collect a diverse set of candidate queries, and then effectively maximizes the collection of mission-relevant observations, and

4. A comprehensive evaluation of the entire system in simulations of a robot exploring real coral reef environments built from high-resolution maps collected in the US Virgin Islands by a real AUV, wherein the robot collects up to 75% more useful observations at the maximum long-distance underwater communications bandwidth rate (13.2 kbps) and up to 50% more at the lowest rate tested (825 bps).

6.1 Related Works

Robotic explorers already serve many purposes varying from the highly specific, such as locating search and rescue targets [194]–[196], to the highly general, such as surveying the deep ocean [197]–[199] or even other worlds [200]. At both extremes, greater autonomy has significantly increased robotic explorers' operational efficiency. In narrowly specified missions, developments in autonomous sensing, perception, and modelling have enabled robots to autonomously detect predefined targets of interest using a range of modalities including visual, thermal, and chemical signatures [201]–[203]. For general mapping, the goal of maximizing area coverage subject to obstacle avoidance and other constraints is solved by coverage-based path planners and frontier-based exploration [204]–[207], and novelty-seeking algorithms can draw a "curious" robot towards the most unusual phenomena in its environment [9]. In practice, the goals and methods of exploration tend to fall between such extremes; while a mission's objective might begin as, for example, conducting a broad survey of a region, observations made *in situ* may call for changes in the robot's behaviour.

An autonomous robot's behaviour is governed by its *reward model*. Reward models can describe a wide variety of tasks, including a variety of "information-gathering" objectives which are often pursued in autonomous exploration and informative path planning [132], [136], [208]–[214]. These objectives are usually defined with respect to some scalar field(s) that the robot is intended to explore. For example, some missions may seek to find and collect observations at the maxima of such fields (e.g., [4]). In these examples, the robot's behaviour

is driven by reducing uncertainty in its understanding of the world, while the *parameters* of its reward model are known and fixed. Conversely, in the adaptive exploration problem, the parameters that map semantic classes to reward are taken to be *a priori* unknown; learning these parameters online is how the robot adapts to new mission objectives. In order to learn a reward model online from queries, the model needs to have a very small number of parameters [13]. Furthermore, risk-based query selection methods like AsympGreedy-EBRM need to sample possible reward model parameters, thereby requiring that the reward model is probabilistic and maintains a distribution over possible parameter values. As this distribution will be updated regularly and *in situ* as the robot receives labels from the human supervisor, updating this distribution must also be computationally affordable.

The degree to which a robot can efficiently exploit its knowledge of the world to collect as much reward as possible depends on the quality of its *planner*. In the adaptive exploration problem, the planner is responsible for constructing trajectories that locate as many targets of interest as possible, based on the robot's knowledge of the semantic map and reward model parameters, which can together produce a reward map. Sampling based planners such as Monte-Carlo Tree Search (MCTS) can produce arbitrarily close-to-optimal trajectories given sufficient computation time [20], [210], [211]. However, these can be computationally intractable for planning over long horizons; this is especially problematic when the reward model changes significantly, requiring new trajectories to be planned from scratch. An alternative approach is *topological planning* [214], which creates a graph representation of the environment over which simple planners, such as greedy selection strategies, can produce high quality trajectories. These are particularly effective when planning over continuous fields which have strong topological structure [214]. Semantic maps tend to be highly spatially correlated, making them well suited for this kind of abstraction [11], [215].

6.2 The Adaptive Exploration Problem Setup

In this section, we formalize the human-in-the-loop adaptive exploration problem. We begin by introducing the notation used throughout this chapter. We use $n \in \mathbb{N}$ to index discrete timesteps and $t \in \mathbb{R}$ to denote continuous time. As a subscript, n denotes the state of a variable at time t_n. We use i and j as general-purpose indices, and use $\mathbf{e}_i \in \mathbb{R}^D$ to denote the unit basis vectors with $i \in \{1, \ldots, D\}$. The unit simplex is denoted $\Delta^K \coloneqq \{\boldsymbol{p} \in \mathbb{R}_+ : \|\boldsymbol{p}\|_1 = 1\}$ (for some $K \in \mathbb{N}_+$), and the space of probability measures over a set $\mathcal{X}$ is denoted $\mathcal{P}(\mathcal{X})$. All vectors are column vectors.

Consider a robotic explorer collecting visual images at a set of locations $\{\boldsymbol{x}_i \in \mathbb{R}^D\}$.[1] The

[1]We typically use $D = 2$ as we typically expect an AUV to collect downward facing imagery while

robot is cooperating with a remote human supervisor who, if they had complete knowledge of the environment, could label all targets of interest by defining a function $R : \mathbb{R}^D \to \{0, 1\}$. However, in practice this function can only be evaluated at a given location $\boldsymbol{x}$ if the robot sends an image of that location to the supervisor. In fact, these hidden labels are precisely equivalent to the hidden *outcomes* described in Chapter 5, where the observation function provides the true label $\boldsymbol{y}_i = R(\boldsymbol{x}_i)$ for some location only if an action is taken to make an image query from location $\boldsymbol{x}_i$. Communication bandwidth limitations limit the robot to a small number of queries over the course of a mission.

The robot's goal is to maximize its accumulated *reward* by the time the mission ends at time $T \in \mathbb{R}_+$, where the reward is defined by the type of mission. We will focus on a type of mission called *adaptive target search*, for which the robot is free to travel anywhere within some predefined area of operations while searching for targets of interest. This matches the general autonomous exploration paradigm presented in Chapter 1.

We assume that, at the end of the mission, the robot is recovered and offloads all of its observations using a high-bandwidth communications link. The robot receives one unit of reward for each distinct target of interest that it observes. That is, assuming that its sensing radius (field of view) is $\gamma \in \mathbb{R}_+$, it cannot receive multiple rewards for observations collected at locations $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ such that $\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_\infty < \gamma$. For simplicity, we will henceforth assume that $\gamma = 1$ so that high level path planning can be performed on a grid with discrete coordinates $\boldsymbol{x} \in \mathbb{Z}^D$ wherein a reward can only be collected once at each coordinate.

We denote the planned *trajectory* of the robot at timestep n as $\tau_n : \mathbb{R} \to \mathbb{Z}^D$, such that $\tau_n(t)$ represents the planned location of the robot on the grid at time t. Note that for $t \in [0, t_n]$ then $\tau_n(t)$ indicates the robot's past and present locations, while for $t > t_n$ it represents where the robot plans to be at some future time. We use $\Omega(\tau_n) \subset \mathbb{Z}^D$ to denote the set of locations visited by τ_n, where

$$\Omega(\tau_n) \coloneqq \left\{ \boldsymbol{x} \in \mathbb{Z}^D \;\middle|\; \exists t \in [0, T] : \tau_n(t) = \boldsymbol{x} \right\}, \tag{6.2.1}$$

and similarly define, for convenience, $\mathcal{X}_n$ to be the set of only *previously* visited locations,

$$\mathcal{X}_n \coloneqq \left\{ \boldsymbol{x} \in \mathbb{Z}^D \;\middle|\; \exists t \in [0, t_n] : \tau_n(t) = \boldsymbol{x} \right\} \subseteq \Omega(\tau_n). \tag{6.2.2}$$

maintaining a roughly constant altitude above the seafloor, however in some applications it may be better to model the robot's position in 3-dimensions.

The total reward collected by a trajectory τ_n is thus

$$R_{\text{ATS}}(\tau_n) \coloneqq \sum_{\boldsymbol{x} \in \Omega(\tau_n)} R(\boldsymbol{x}). \tag{6.2.3}$$

which can only be computed after the vehicle has been recovered and all observations have been offloaded.

6.3 Online Reward Learning in Adaptive Exploration

Let us assume that the robot is equipped with a predictive spatial semantic model that computes $P(\boldsymbol{z} \mid \boldsymbol{x})$, which describes the likelihood that an observation collected at $\boldsymbol{x} \in \mathbb{Z}^D$ is mapped to the semantic class distribution $\boldsymbol{z} \in \Delta^{\kappa_n}$, where κ_n denotes the number of distinct semantic classes in use at timestep n. We do not generally expect a *deterministic* relationship between the observed semantic class distributions $\boldsymbol{z}_i$ and the reward labels $R(\boldsymbol{x}_i)$ at locations $\boldsymbol{x}_i$. Instead, we consider the conditional expectation $\mathbb{E}[R(\boldsymbol{x}) \mid \boldsymbol{z}]$, which describes the expected reward of an observation with the semantic representation $\boldsymbol{z}$. Given such a model, the robot can predict the reward of an observation collected at any $\boldsymbol{x} \in \mathbb{Z}^D$ as

$$\mathbb{E}[R(\boldsymbol{x}) \mid \boldsymbol{x}, \mathcal{D}_n] = \int_{\Delta^{\kappa_n}} \mathbb{E}[R(\boldsymbol{x}) \mid \boldsymbol{z}] P(\boldsymbol{z} \mid \boldsymbol{x}) d\boldsymbol{z}. \tag{6.3.1}$$

To approximate this expectation, we must make the structural assumption that $\mathbb{E}[R(\boldsymbol{x}) \mid \boldsymbol{z}] \approx g(\boldsymbol{z}; \theta)$ for some function $g : \Delta^{\kappa_n} \to \mathbb{R}$ with hidden parameters $\theta \in \Theta$; this function is called the *reward model*, and these hidden parameters are of precisely the same kind as the ones discussed in Chapter 5. The problem can thus be framed as actively acquiring a set of representation-label pairs $\mathcal{D}_n = \{(\boldsymbol{z}_1, y_1), \dots\}$, where $y_i = R(\boldsymbol{x}_i)$, from which to learn the reward model parameters θ online. We assume that we can identify a reasonable prior for these parameters, $P(\theta)$, and compute the posterior belief distribution with Bayes' law,

$$b_n = P(\theta \mid \mathcal{D}_n) = \frac{P(\mathcal{D}_n \mid \theta) P(\theta)}{P(\mathcal{D}_n)}. \tag{6.3.2}$$

The reward model of the robot is thus defined as

$$\mathbb{E}[R(\boldsymbol{x}) \mid \boldsymbol{x}, \mathcal{D}_n] \approx \int_{\Theta} \int_{\Delta^{\kappa_n}} g(\boldsymbol{z}; \theta) P(\boldsymbol{z} \mid \boldsymbol{x}) b_n(\theta) d\boldsymbol{z} d\theta. \tag{6.3.3}$$

As in Chapter 5, we use S_n to denote the state of the robot at timestep n, which includes its belief distribution b_n as well as all other variables relevant to its planning procedure.

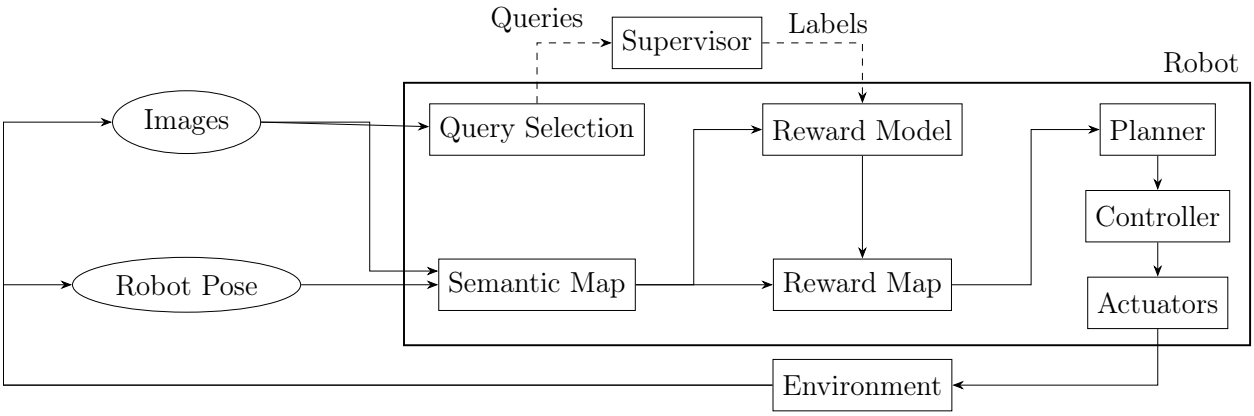


Figure 6.2: The proposed system architecture. A query selection algorithm selects which images to send over a low-bandwidth communications link (dashed arrow) to the human supervisor, who responds to each image with a label indicating whether its contents are relevant to the mission.

6.4 Proposed System Architecture

The proposed approach has five components. The first is the unsupervised vision-based semantic model, which produces a set of semantic classes that distinguish the various phenomena which the robot has encountered, without requiring any examples or pre-training. These semantic classes generally represent various habitats, terrain types, or anomalies within the environment. Next is online Bayesian interest regression, which models the co-occurrence of the learned semantic classes with the supervisor's reward labels to produce a reward model. The third component is the spatial semantic model, which produces a semantic map predicting which semantic classes the robot is most likely to find in new (unvisited) locations; this can be used in combination with the reward model to construct a scalar reward map. The fourth is hotspot-based adaptive path planning, which identifies the regions of the reward map with the highest reward concentration and plans paths which densely cover these "hotspots". Finally, the last component is an AsympGreedy-EBRM[2] based communications policy which is used to learn the mission objectives (i.e., the reward model) with the minimal number of queries. The following subsections describe each component in detail.

6.4.1 Unsupervised Vision-Based Semantic Modelling

We model the world as a semantic field $\boldsymbol{Z}_n : \mathbb{Z}^D \to \Delta^{\kappa_n}$ over the learned semantic classes, such that every location $\boldsymbol{x} \in \mathbb{Z}^D$ is associated with a semantic class distribution $\boldsymbol{Z}_n(\boldsymbol{x})$. Specifically, we make use of the spatiotemporal semantic mapping system presented in Sub-

[2]Refer to Chapter 5 for details.

section 2.2 to develop learned semantic representations online without needing any prior training or examples of the target environment.

6.4.2 Reward Model Learning

The robot routinely uses a query selection algorithm to choose a previously observed location $\boldsymbol{x} \in \mathcal{X}_n$, and sends the human supervisor the image most representative of it; the supervisor then provides a binary response $y = R(\boldsymbol{x})$ indicating the relevance of the image observation contents to the mission objectives. We define the label set $\mathcal{L}_n \subseteq \mathcal{X}_n$ as the set of locations labelled by timestep n. From the training dataset $\mathcal{D}_n$, we construct the feature matrix $\boldsymbol{\Phi}_n \in [0,1]^{|\mathcal{L}_n| \times \kappa_n}$ and label vector $\boldsymbol{y}_n \in \{0,1\}^{|\mathcal{L}_n|}$, such that

$$\boldsymbol{\Phi}_n = \Big[\boldsymbol{Z}_n(\boldsymbol{x})^\top\Big]_{\boldsymbol{x}\in\mathcal{L}_n}, \qquad \boldsymbol{y}_n = \Big[R(\boldsymbol{x})\Big]_{\boldsymbol{x}\in\mathcal{L}_n}. \tag{6.4.1}$$

As discussed in Subsection 6.3, we must choose a model $g : \Delta^{\kappa_n} \times \Theta \to \mathbb{R}$ for which we assume that $\exists\theta \in \Theta : g(\boldsymbol{Z}_n(\boldsymbol{x}); \theta) \approx \mathbb{E}[R(\boldsymbol{x})]$ holds $\forall x$. There are several considerations in choosing this model: firstly, it must be tractable to compute Eq. (6.3.3). Secondly, the model's sample complexity should be low, proportional to the available communications bandwidth; this generally means that $\dim\Theta$ must be small, such that a relatively few labels are required to obtain a good estimate of the true model parameters $\theta \in \Theta$. Relatedly, there must be an efficient inference algorithm to infer the posterior parameter distribution $P(\theta \mid \mathcal{D}_n)$ given $P(\theta \mid \mathcal{D}_{n-1})$. This is particularly important for AsympGreedy-EBRM query selection, as it must consider many counterfactual posterior belief distributions. Lastly, it is preferable to choose a model which is interpretable, such that the robot can describe its model to the human supervisor in order to explain why it is sending a specific query or choosing a specific plan; this is particularly important for building trust in the system, which is critical in high-budget exploration use cases (e.g. exploring the deep ocean).

Logistic regression, the classical approach to problems of this type, assumes there is a weight vector $\boldsymbol{w} \in \mathbb{R}^{\kappa_n}$ with prior $\boldsymbol{w}_n(k) \sim \mathcal{N}(0, \sigma_w^2)$ such that

$$g(\boldsymbol{z}; \boldsymbol{w}_n) = \sigma\big(\boldsymbol{w}_n^\top \boldsymbol{z}\big), \tag{6.4.2}$$

where $\sigma(x) = (1 - e^{-x})^{-1}$ is the logistic sigmoid function. The posterior belief distribution is usually modelled as a multivariate Gaussian distribution $b_n(\boldsymbol{w}) = P(\boldsymbol{w} \mid \boldsymbol{\Phi}_n, \boldsymbol{y}_n)$, for which the parameters can be solved for using techniques such as variational inference. Though popular, we do not find this model well suited for the adaptive exploration use case. The expectation in Eq. (6.3.3) cannot be solved analytically for this model, so computing them

requires either approximations or sampling-based methods. Furthermore, despite the model's simplicity, it can require significant computation in order to fully converge on the posterior belief distribution parameters, and they will vary depending on the weight initialization. As an alternative, we propose a novel reward models with interpretable parameters for which expectations and posteriors can be computed efficiently and precisely.

The Beta-Concentration Reward Model

This model assumes that each semantic class $k \in [\kappa_n]$ is associated with a particular *concentration* $\boldsymbol{\rho}_n(k) \in [0, 1]$ of producing valuable observations, and the expected value of an observation for some distribution of semantic classes is a linear mixture of these concentrations,

$$g(\boldsymbol{z}; \boldsymbol{\rho}_n) = \boldsymbol{\rho}_n^\top \boldsymbol{z}. \tag{6.4.3}$$

We set a Beta distribution prior on each concentration, $\boldsymbol{\rho}_n(k) \sim \text{Beta}(\alpha_0, \beta_0)$ with hyperparameters $\alpha_0 > 0$ and $\beta_0 > 0$. We approximate the posterior belief distribution with another Beta distribution using an analytical update rule,

$$\boldsymbol{\rho}_n \mid \boldsymbol{\Phi}_n, \boldsymbol{y}_n \sim \text{Beta}(\boldsymbol{\alpha}_n, \boldsymbol{\beta}_n), \tag{6.4.4}$$

$$\boldsymbol{\alpha}_n = \boldsymbol{\alpha}_0 + \boldsymbol{\Phi}_n^\top \boldsymbol{y}_n, \tag{6.4.5}$$

$$\boldsymbol{\beta}_n = \boldsymbol{\beta}_0 + \boldsymbol{\Phi}_n^\top \big(\mathbf{1}_{|\mathcal{L}_n|} - \boldsymbol{y}_n\big), \tag{6.4.6}$$

$$\mathbf{1}_{|\mathcal{L}_n|} \coloneqq \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix}^\top \in \mathbb{R}^{|\mathcal{L}_n|}. \tag{6.4.7}$$

This update is exact if $\boldsymbol{\Phi}_n \in \{0, 1\}^{|\mathcal{L}_n| \times \kappa_n}$, which corresponds to each labelled example being representative of only a single semantic class.[3] When this is not the case, the true posterior is not analytical. However, we find that this approximate update rule works very well for small numbers of labels, particularly when this condition is nearly satisfied (i.e., when the robot only requests labels for observations dominated by a particular semantic class).[4]

A key advantage of the beta-concentration model is that it is linear in both $\boldsymbol{z}$ and $\boldsymbol{\rho}_n$.

[3] Recall that each row of $\boldsymbol{\Phi}_n$ is a distribution over semantic classes.

[4] We present a comparison of the proposed beta-concentration reward model and a traditional logistic model trained on randomly generated data in Figure B.1, and find that the beta-concentration model performs similarly well despite its substantially reduced computational complexity.

This makes the spatial reward model separable such that Eq. (6.3.3) can be simplified as,

$$\int_{\Theta}\int_{\Delta^{\kappa_n}} g(\boldsymbol{z};\boldsymbol{\rho}_n)P(\boldsymbol{z}\mid\boldsymbol{x})b_n(\boldsymbol{\rho}_n)d\boldsymbol{z}d\boldsymbol{\rho}_n = \int_{\Theta}\int_{\Delta^{\kappa_n}} \boldsymbol{\rho}_n^{\top}\boldsymbol{z}\cdot P(\boldsymbol{z}\mid\boldsymbol{x})b_n(\boldsymbol{\rho}_n)d\boldsymbol{z}d\theta \tag{6.4.8}$$

$$= \mathbb{E}\big[\boldsymbol{\rho}_n^{\top}\mid b_n\big]\mathbb{E}[\boldsymbol{z}\mid\boldsymbol{x}] = \frac{\boldsymbol{\alpha}_n^{\top}\mathbb{E}[\boldsymbol{z}\mid\boldsymbol{x}]}{\boldsymbol{\alpha}_n+\boldsymbol{\beta}_n} \tag{6.4.9}$$

6.4.3 Semantic Map Extrapolation

As a result of the choice of beta-concentration reward model, the robot needs only to be able to model the *expected* semantic distribution $\mathbb{E}[\boldsymbol{z}\mid\boldsymbol{x}]$ at a each location $\boldsymbol{x}$, rather than a distribution over semantic distributions. As such, we propose that the robot predicts the values of the semantic map $\boldsymbol{Z}_n$ at unobserved locations through simple K-nearest neighbours extrapolation [216], [217] of locations that have been observed. The extrapolated semantic map $\tilde{\boldsymbol{Z}}$ is thus, for each location $\boldsymbol{x}$, a simple weighted sum of that location's nearest neighbours,

$$\tilde{\boldsymbol{Z}}_n(\boldsymbol{x}) \coloneqq \mathbb{E}[\boldsymbol{z}\mid\boldsymbol{x}] = \begin{cases} \boldsymbol{Z}_n(\boldsymbol{x}), & \boldsymbol{x}\in\mathcal{X}_n, \\ \frac{\sum_{\boldsymbol{v}\in N(\boldsymbol{x})} w(\boldsymbol{x},\boldsymbol{v})\boldsymbol{Z}_n(\boldsymbol{v})}{\sum_{\boldsymbol{v}\in N(\boldsymbol{x})} w(\boldsymbol{x},\boldsymbol{v})}, & \text{otherwise}, \end{cases} \tag{6.4.10}$$

where the neighbourhood $N(\boldsymbol{x})$ contains the $\nu\geq 1$ distinct elements of $\mathcal{X}_n$ which maximize the weighting function $w(\cdot,\cdot)$. As in [218], we set $\nu=3$ and inverse squared Euclidean distance weights $w(\boldsymbol{u},\boldsymbol{v}) = \|\boldsymbol{u}-\boldsymbol{v}\|_2^{-2}$.

We show in Section 6.5 that, in practice, this relatively simple extrapolation approach can often enable the robot to explore nearly as effectively as if it had access to the complete semantic map *a priori*. More sophisticated extrapolation strategies exist, however, and may provide superior performance in other environments or mission types. For example, Gaussian-Dirichlet Random Fields (GDRF) are able to jointly learn semantic classes as well as their spatial distribution in the world, which includes the option to learn a separate lengthscale and spatial covariance structure for each semantic class [12], [219].

Constructing the Reward Map

The robot's *reward map* $R_n : \mathbb{Z}^D \to [0,1]$ indicates the expected reward of visiting any given cell. We assume that the robot only receives a reward the first time it visits a cell, as the goal is to observe as many *distinct* phenomena of interest as possible, so we define it to be

$$R_n(\boldsymbol{x}) = \begin{cases} 0, & \boldsymbol{x}\in\mathcal{X}_n, \\ \frac{\boldsymbol{\alpha}_n^{\top}\tilde{\boldsymbol{Z}}_n(\boldsymbol{x})}{\boldsymbol{\alpha}_n+\boldsymbol{\beta}_n}, & \text{otherwise}. \end{cases} \tag{6.4.11}$$

6.4.4 Efficient Trajectory Planning for Adaptive Target Search

The reward map R_n indicates where the robot is expected to find the most interesting observations; however, it is accurate only where both the extrapolated semantic map and the reward model parameters are reliable. The extrapolated semantic map is generally only reliable within some characteristic lengthscale $r \in \mathbb{R}_+$ of previous observations.[5] Even when the predicted semantic distributions are reliable, the variance of the reward prediction generally depends on having queried observations with related semantic distributions.

In order to address both of these considerations, we have the robot initially track an exploratory trajectory through the operational area, such that as much of the map is covered within the first $t_1 \in \mathbb{R}_+$ seconds. In addition to improving semantic map extrapolation, this exploration provides a diverse and representative collection of semantic observations for the robot to use as queries. The robot can send queries to the supervisor during this time in order to develop its reward model. Then, it plans an exploitative trajectory that maximizes the number of mission-relevant observations it collects from time t_1 until the mission terminates at time T; it continues to update this trajectory as it makes new observations and queries. We discuss both stages in more detail, below.

Stage 1 (Coarse Survey)

In this stage, the robot constructs an initial "exploratory" trajectory $\tau_0 : [0, t_1] \to \mathbb{R}^D$ to maximize the total area within one lengthscale r of any point along the trajectory. This can be formulated as an optimization problem,

$$\tau_0 \in \arg\max_{\tau} \int_{\boldsymbol{x} \in (\tau([0,t_1]) \oplus \mathcal{B}_p(r))} d\boldsymbol{x}, \tag{6.4.12}$$

where $\oplus$ denotes the Minkowski sum and $\mathcal{B}_p(r)$ is the L^p-ball of radius r,

$$\mathcal{B}_p(r) \coloneqq \left\{ \boldsymbol{a} \in \mathbb{R}^D \;\middle|\; \|\boldsymbol{a}\|_p \leq r \right\}. \tag{6.4.13}$$

Finding an optimal solution to Eq. (6.4.12) in the presence of obstacles is generally NP-hard, however there are a wide range of algorithms that can efficiently produce near-optimal trajectories [220]. Given *a priori* knowledge of obstacles, an optimal coverage trajectory can generally be constructed from boustrophedonic paths [204], [205].

[5] We require that $r \gg \gamma$ in order to predict the semantic distribution of locations beyond those that have been previously visited; we typically expect $r \geq 3\gamma$.

Stage 2 (Hotspot Coverage)

In this stage, the robot uses its reward map to plan "exploitative" trajectories that cover regions expected to contain a high concentration of mission-relevant observations. We define the robot's "area of operations" at timestep n as $\mathcal{X}'_n = \mathcal{X}_n \oplus \mathcal{B}_p(r)$ (all locations within one lengthscale of any previous observation) and denote exploitative trajectories as $\tau_n : [t_n, T] \to \mathcal{X}'_n$. The goal is to find a trajectory $\tau_n^\star$ which approximately solves the optimization problem,

$$\tau_n^\star \in \arg\max_{\tau} \sum_{\boldsymbol{x} \in \Omega(\tau)} R_n(\boldsymbol{x}). \tag{6.4.14}$$

This is equivalent to the optimal deterministic open loop policy $\hat{\pi}^\star_{S_n}$

We begin by computing the mission *horizon*,

$$h_n = \frac{v(T - t_n)}{\gamma}, \tag{6.4.15}$$

where $v > 0$ is the maximum speed of the robot while collecting observations. The horizon indicates the maximum number of locations that the robot could visit before the end of the mission. We use h_n to compute a reward threshold $c_n \in [0, 1]$ as the solution to the optimization problem,

$$\begin{aligned} \max_{c \in [0,1]} \quad & c \\ \text{s.t.} \quad & h_n \leq \sum_{\boldsymbol{x} \in \mathcal{X}'_n} \mathbb{1}(R_n(\boldsymbol{x}) \geq c). \end{aligned} \tag{6.4.16}$$

We now construct a graph $(\mathcal{V}_n, \mathcal{E}_n)$ with vertices $\mathcal{V}_n = \{\boldsymbol{x} \in \mathcal{X}'_n \mid R_n(\boldsymbol{x}) \geq c_n\}$ and undirected edges between adjacent locations, $\mathcal{E}_n = \{(\boldsymbol{u}, \boldsymbol{v}) \mid \|\boldsymbol{u} - \boldsymbol{v}\|_1 = 1, (\boldsymbol{u}, \boldsymbol{v}) \in \mathcal{V}_n\}$. The connected components of the graph represent "hotspots" in the reward map; let $\mathscr{H}_n$ denote the set of these connected components. We construct a sequence of hotspots, called the meta-trajectory, $\hat{\tau}_n$, and then construct τ_n to be a feasible trajectory that densely covers these hotspots. The *score* of a hotspot $\mathcal{H} \in \mathscr{H}_n$ is the total reward covered by $\mathcal{H}$ divided by its size and distance from the robot's location. The score of a meta-trajectory $\hat{\tau} = \{\mathcal{H}_1, \dots\}$ is thus

$$s(\hat{\tau}) = \sum_{i=1}^{|\hat{\tau}|} \frac{\sum_{\boldsymbol{x} \in \mathcal{H}_i} R_n(u)}{\gamma |\mathcal{H}_i| + d(\mathcal{H}_i, \mathcal{H}_{i-1})}, \tag{6.4.17}$$

$$d(\mathcal{H}_1, \mathcal{H}_2) = \gamma \cdot \left\| \sum_{\boldsymbol{x} \in \mathcal{H}_1} \frac{\boldsymbol{x}}{|\mathcal{H}_1|} - \sum_{\boldsymbol{x} \in \mathcal{H}_2} \frac{\boldsymbol{x}}{|\mathcal{H}_2|} \right\|, \tag{6.4.18}$$

$$\mathcal{H}_0 = \{\boldsymbol{x}\}, \tag{6.4.19}$$

where $\mathcal{H}_0$ represents the robot's current cell location and $d(\cdot,\cdot)$ measures the distance between hotspot centroids. We greedily construct the meta-trajectory $\hat{\tau}_n$ by iteratively choosing the next hotspot $\mathcal{H}$ to maximize the score $s(\hat{\tau}+\{\mathcal{H}\})$ until the trajectory length exceeds the horizon,

$$\sum_{i=1}^{|\hat{\tau}_n|}(\gamma\cdot|\mathcal{H}_i|+d(\mathcal{H}_i,\mathcal{H}_{i-1}))\geq h_n. \tag{6.4.20}$$

After each greedy selection, we use the 2-opt local search heuristic to optimize the ordering of the hotspots.

Lastly, we construct the trajectory τ_n to densely cover the hotspots in $\hat{\tau}$ using any standard coverage path planner. In our experiments, we assume that the robot is operating above any obstacles, so complete coverage can be achieved by chaining a sequence of Boustrophedonic coverage paths (simple lawnmower patterns), which are trivial to construct.

6.4.5 Risk-Based Online Query Selection

The optimal deterministic open loop policy for the robot state S_n is equivalent to the optimal trajectory $\tau_n^\star$, which our planner approximates with τ_n; thus, generating the ODOL policy for a counterfactual state $S_{n+1}^{y,\boldsymbol{z}}$, is equivalent to updating the reward model based on a new query-label pair $\boldsymbol{z}, y$, and then planning a trajectory for the updated (counterfactual) reward map $R_{n+1}^{y,\boldsymbol{z}}$, which we denote $\tau_{n+1}^{y,\boldsymbol{z}}$. As the approximate posterior reward model updates in Eqs. (6.4.5) and (6.4.6) are linear, updating the reward model is extremely efficient, with

$$\boldsymbol{\alpha}_{n+1}^{y,\boldsymbol{z}}=\boldsymbol{\alpha}_n+y\boldsymbol{z}, \tag{6.4.21}$$

$$\boldsymbol{\beta}_{n+1}^{y,\boldsymbol{z}}=\boldsymbol{\beta}_n+(1-y)\boldsymbol{z}. \tag{6.4.22}$$

Thus, the vast majority of the time required to generate counterfactuals is in planning an updated trajectory.

The expected value of information, as defined in Chapter 5, for a new query $\boldsymbol{z}$ is therefore

$$\mathrm{EVoI}(\boldsymbol{z};S_n)=\mathbb{E}_{y|b_n,\boldsymbol{z}}\left[\sum_{\boldsymbol{x}\in\Omega(\tau_{n+1}^{y,\boldsymbol{z}})}R_{n+1}^{y,\boldsymbol{z}}(\boldsymbol{x})-\sum_{\boldsymbol{x}\in\Omega(\tau_n)}R_{n+1}^{y,\boldsymbol{z}}(\boldsymbol{x})\right]. \tag{6.4.23}$$

The asymptotic value of information is similarly straightforward to compute,

$$\mathrm{AVoI}(\boldsymbol{z};S_n)=\mathbb{E}_{\boldsymbol{\rho}|b_n}\left[\sum_{\boldsymbol{x}\in\Omega(\tau_{\boldsymbol{\rho}}^\star)}\boldsymbol{\rho}_n^\top\tilde{\boldsymbol{Z}}_n(\boldsymbol{x})\right]-\mathbb{E}_{\boldsymbol{\rho}|\tilde{b}_{n+1}^{\boldsymbol{z}}}\left[\sum_{\boldsymbol{x}\in\Omega(\tau_{\boldsymbol{\rho}}^\star)}\boldsymbol{\rho}_n^\top\tilde{\boldsymbol{Z}}_n(\boldsymbol{x})\right], \tag{6.4.24}$$

where $\tilde{b}^{\boldsymbol{z}}_{n+1} = b^{\mu,\boldsymbol{z}}_{n+1}$, with $\mu = \frac{\boldsymbol{\alpha}_n^\top \boldsymbol{z}}{\boldsymbol{\alpha}_n + \boldsymbol{\beta}_n}$, represents the convergence of the belief distribution such that it has the same mean but a smaller variance, scaled based on the Fisher information of the new query (refer to Subsection 5.4.2 for context). Defining the EVoI and AVoI functions fully characterizes the AsympGreedy-EBRM algorithm for this online learning problem, which will select queries $\boldsymbol{z}$ to maximize $\max\{\text{EVoI}(\boldsymbol{z}; S_n), \text{AVoI}(\boldsymbol{z}; S_n)\}$.

6.5 Experimental Results

In this section we evaluate the proposed system in simulated adaptive exploration missions of real coral reefs located around the US Virgin Islands.

6.5.1 Methodology

Our team collected two coral reef image datasets in July 2023 at the "Booby Rock"[6] and "Tektite"[7] sites located nearby the Virgin Islands Environmental Resource Station (VIERS). The images were collected with a CUREE AUV platform [166]. The datasets were post-processed for color correction and the commercial Metashape software [222] was used to perform bundle adjustment of the images to establish ground truth robot positioning and to create 3D reconstructions of the sites. Top-down orthomosaics of the sites, created with Metashape, are presented for visualization purposes in Figures 6.3 and 6.6.

We assume a robot sensing radius of 0.5 m, and discretize the sites accordingly. The corresponding unsupervised semantic maps of the sites, using the same discretization, were produced with the Sunshine unsupervised semantic mapping system from [11] and are depicted in Figures 6.4 and 6.7, respectively. These semantic maps were created using original images, as in [11], rather than from the orthomosaics in Figures 6.3 and 6.6.

In each dataset, we segmented all Caribbean "sea fans" (genus: Gorgonia) within a subset of images. We then used the manual segmentations to train a deep image segmentation network [223] to segment the rest of the images in each dataset. In each image where more than 5% of the pixels were labeled as sea fans, we marked the corresponding location as containing a target of interest. This was used to create a ground truth reward map $R(\boldsymbol{x})$ for each site, which defined the responses of a simulated human supervisor to queries; these maps are depicted in Figure B.3. An example of an image and corresponding segmentation is presented in Figure 6.5. The goal of the simulated adaptive exploration missions was to locate and collect images of as many of these sea fans as possible. In testing, we found that

[6] https://www.venturefarther.com/mapObject/MapObjectSharedInfo.action?mapObject.id=4522
[7] Named after the subsea habitat which was formerly present here [221].

Figure 6.3: Booby Rock Reef. The orthomosaic above is 32 m wide by 30 m tall, and was produced at a resolution of 1 pixel per mm^2 (960MP total). The larger dark patches show reef, while the bright regions are mostly sand. The small dark patches among the sand are mostly rocks.

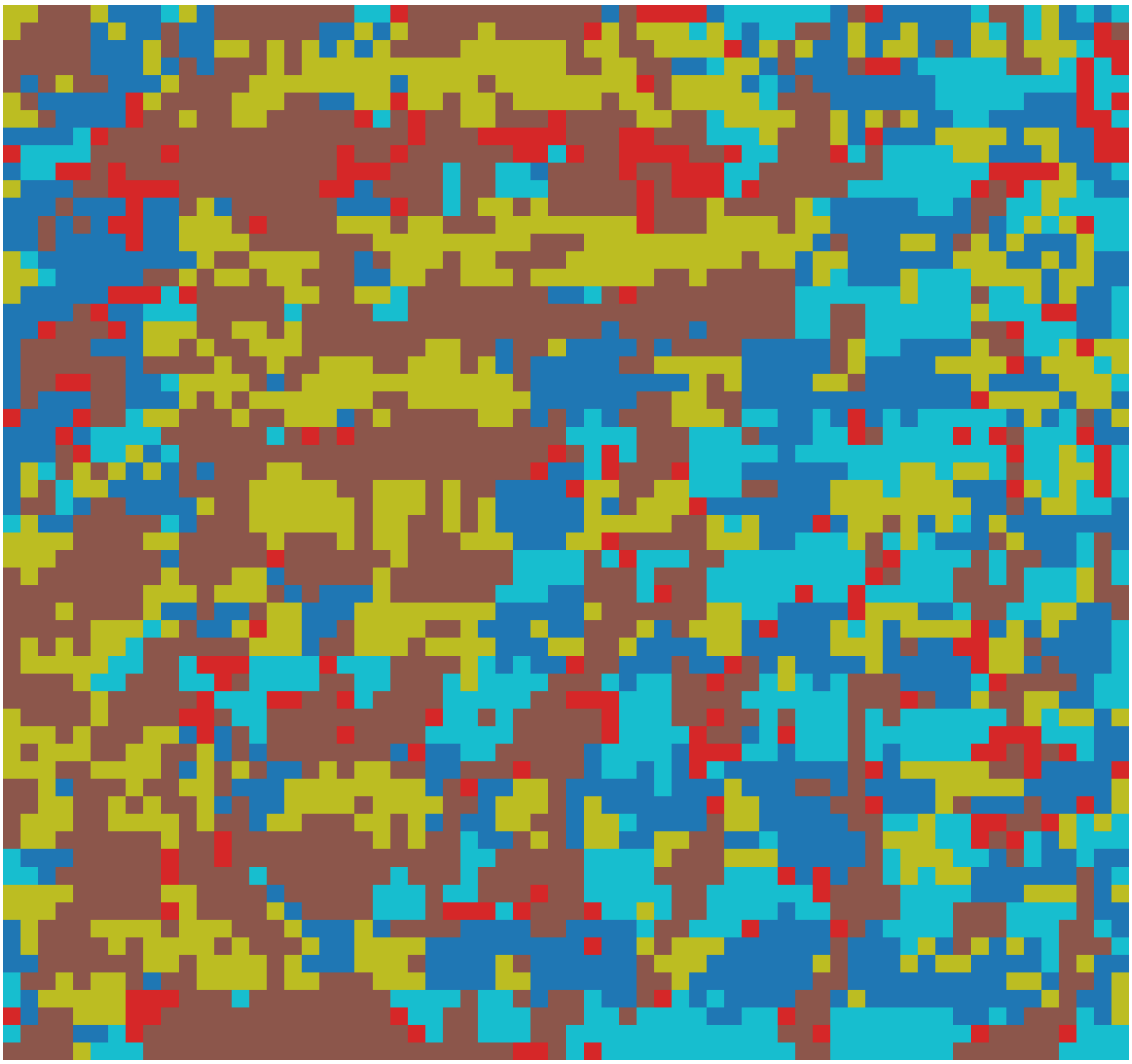


Figure 6.4: Semantic Map of Booby Rock Reef, using 0.5 by 0.5 m grid cells (64 x 60 cells). The chosen colors are arbitrary and represent only the largest component of the semantic distribution at each location. Brown and yellow patches correspond best with sandy regions, shades of blue to reef, and red to large rocks and rubble. This map was created with the Sunshine package [11] with hyperparameters $K = 8$, $\alpha = 0.01$, $\beta = 1$, and $\gamma = 0$.

most of the images in the dataset remained relatively clear when reduced in size with JPEG compression to as small as 50 kB. Example images of sizes 2 MB, 55 kB, and 30 kB are presented in Figure B.2.

Scientific Context & Motivation Caribbean sea fans have been the subject of significant study due to recent outbreaks of disease and parasites which target them [224]–[226]. They are a type of soft coral, and while they most often grow within reef habitats, they can also be found on rocks isolated in mostly sandy regions. Diseases can have a significant affect on the appearance of Gorgonia species [226], which would make it challenging to train a detector to robustly recognize specimens suffering from novel diseases, which are likely the most important to discover. In effect, we simulate a human supervisor who takes an interest in the state of the sea fans at these sites, without necessarily having planned to in advance.

We ran 1500 simulated exploration missions on each dataset, at each of 5 different bandwidth levels ranging from 13.2 kbps, near the maximum speed of mid-range acoustic communications ($\geq$ 100 m) under ideal conditions, to 825 bps, which approximates the acoustic communication bandwidth rates of deep sea vehicles. Note that these values are in *bits* per second, so at the maximum data transfer rate it required about 30 s to transmit a 50 kB (400 kbit) query and receive a response label, while at 825 bps it required 480 s (8 minutes). We tested two different mission durations, $T = 1500$ and $T = 3000$, such that the robot was able to make between 3 and 100 queries over the course of the mission.

We compared AsympGreedy-EBRM query selection to two baseline approaches:

- Most Recent Obs.: This query selection strategy always picks the most recently collected image to send to the supervisor, whenever the previous transmission has finished. This is equivalent to sending uniformly spaced query images from along the trajectory.[8]

- Semantic Diversity: This query selection strategy repeatedly cycles through the semantic classes, sending the best unlabeled "exemplar" of each for labeling. The first query will be the image observation most associated with the first semantic class, the second with the second, and so on. Once it has queried an exemplar for each of the κ_n semantic classes, it repeats from the first class, but selecting the next-best exemplars (i.e., the best *unlabeled* exemplars).

Compared to the baselines, the AsympGreedy-EBRM query selection is relatively expensive to compute; in particular, it requires generating counterfactual plans for each possible query. This is not scalable, since the number of potential queries grows linearly over the course of the mission. Accordingly, we only evaluate the AsympGreedy-EBRM algorithm

[8]Such transmissions are already commonly used in practice for monitoring the status of AUVs.

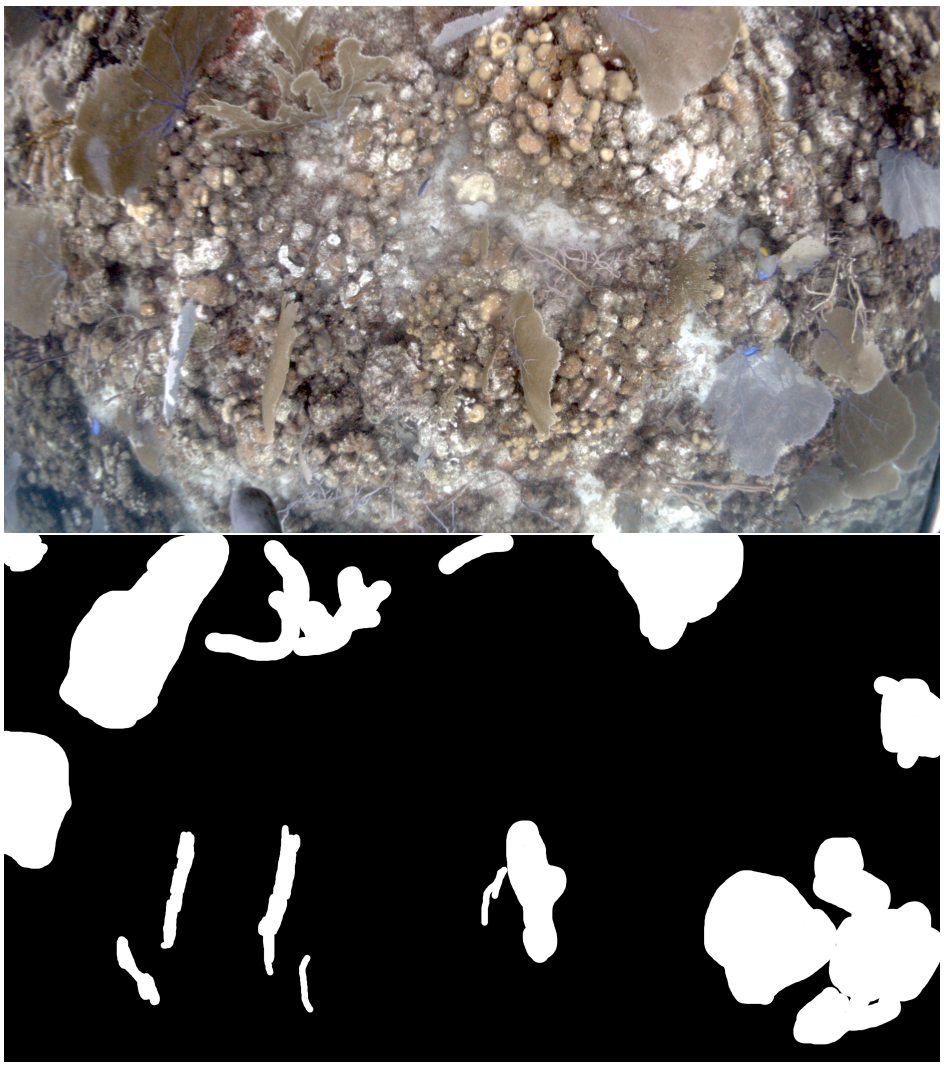


Figure 6.5: Example of an image from the Tektite reef dataset containing several specimens of Gorgonia (above), with a corresponding manual semantic segmentation (below).

Figure 6.6: Tektite Reef. The orthomosaic above is 60 m wide by 45 m tall, and was produced at a resolution of 1 pixel per mm^2 (2.7 gigapixels total). Interpretation of the contents is largely similar to Booby Rock Reef. There is some distortion in the lower right part of the orthomosaic where the robot did not have dense visual coverage.

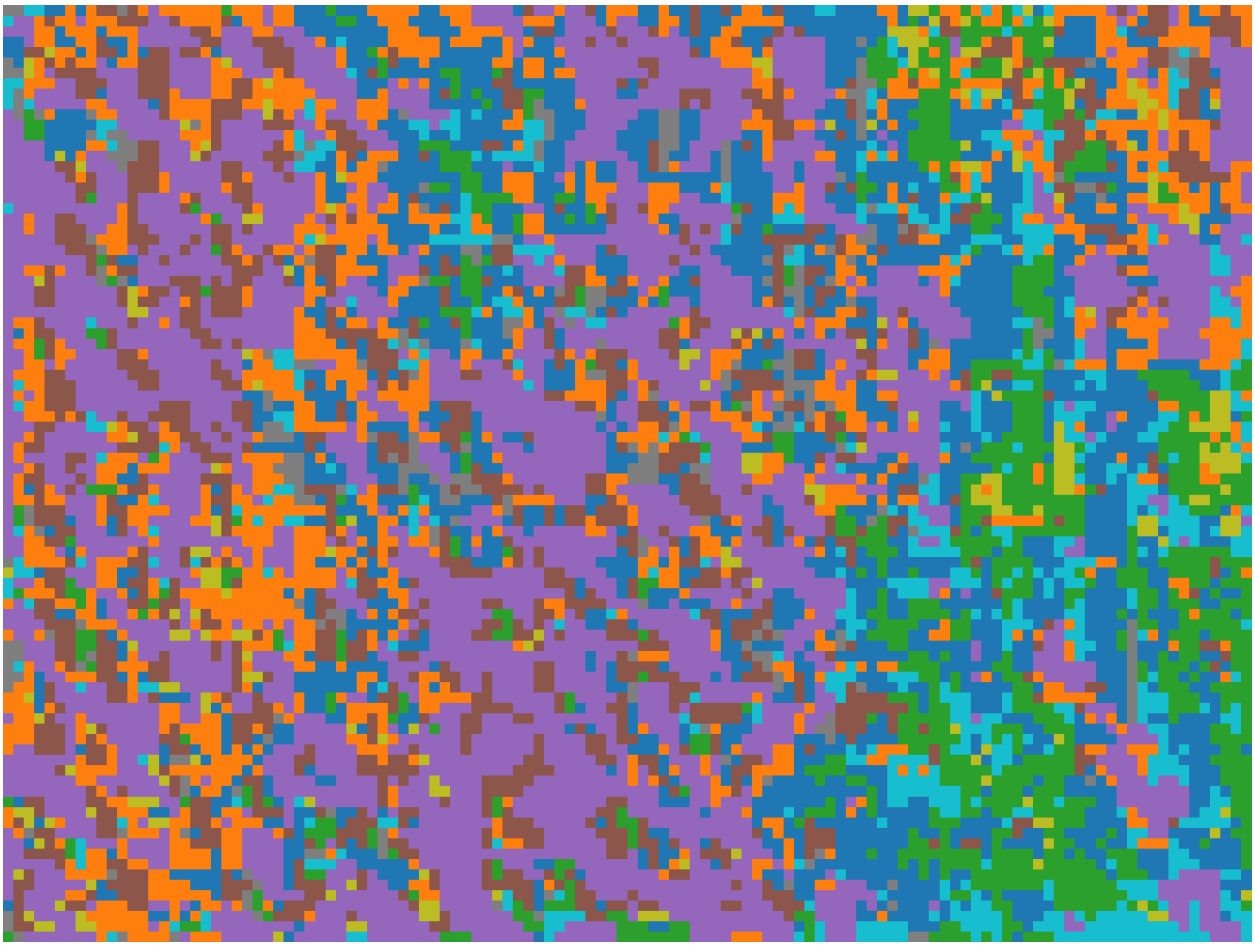


Figure 6.7: Semantic Map of Tektite Reef, using 0.5 by 0.5 m grid cells (120 x 90 cells). The chosen colors are arbitrary and represent only the largest component of the semantic distribution at each location. Purple corresponds best with sand, orange with seagrass, and the green and blue shades with reef. This map was created with the Sunshine package [11] with hyperparameters $K = 8$, $\alpha = 0.01$, $\beta = 1$, and $\gamma = 0$.

on $2 * \kappa_n$ candidate queries. The first κ_n of these queries are chosen as the "exemplars" of each semantic class, as considered by the "Semantic Diversity" query selector. The most recent image observation[9] is also evaluated, while the remaining $\kappa_n - 1$ candidates are chosen uniformly at random from the previously collected observations.

The initial position of the robot was randomized in each experiment. The initial coarse survey was 42.5 meters wide by 32.5 m tall with 6 arms for the Tektite experiments, and 26 m wide by 25 m tall with 10 arms for the Booby Rock experiments. Each of these coarse surveys was designed to take approximate 10 minutes to complete at a speed of 0.5 m s^{-1}. The robot had either 15 or 40 additional minutes to adaptively explore based on its learned reward model, depending on the value of T. The robot was able to send queries throughout the mission, so would always transmit its first observation, and would then pick the second query from among the images collected while first one was being broadcast. Note that the Booby Rock site map contains only 3840 cells, so the extended duration provides sufficient time to visit over 75% of the Bobby Rock site, compared to only 28% of the Tektite site.

6.5.2 Variant Mission: Adaptive Sample Selection

We also considered a different type of mission wherein the robot is restricted to following a pre-planned trajectory but can select specific locations along the way to collect "samples" near targets of interest. In marine exploration, this may represent collecting a water sample to be analyzed after vehicle recovery, or simply slowing down near a target of interest to collect higher-resolution imagery than would typically be collected, which can enable more detailed analysis and 3D reconstruction in later (offline) processing. Importantly, these missions are characterized by a *sample budget* $B \in \mathbb{N}_+$, which limits the number of samples which can be collected.

This type of mission is a compromise between traditional pre-planned surveys, which provide an unbiased estimate of the distribution of various phenomena in a region, and target search, which seeks to bias data collection towards targets of interest. We make the same assumptions as for adaptive target search that there is no additional reward for collecting multiple samples within some sensing radius γ, and that planning will henceforth take place on a grid $\boldsymbol{x} \in \mathbb{Z}^D$ with spacing equal to γ.

We use τ to denote the pre-planned trajectory, and $\delta_n \subset \Omega(\tau)$ to denote the set of locations selected for sample collection at timestep n. This set contains both the locations which have been sampled already, as well as locations that the robot plans to sample later

[9] That is, the one would would be selected by the "Most Recent Obs." query selector.

along the trajectory. The total reward collected by a set of sample selections δ_n is thus

$$R_{\text{SS}}(\delta_n) \coloneqq \sum_{\boldsymbol{x} \in \delta_n} R(\boldsymbol{x}). \tag{6.5.1}$$

We evaluated the proposed system for this sample selection mission, which required only replacing the hotspot-based trajectory planner with a greedy sample selection procedure, which sets δ_n to contain the B locations along the path with the highest expected reward.[10]

We set τ to consist of the initial coarse survey τ_0 used by the hotspot-planner, followed by a denser survey over the same region. For the Booby Rock experiments this denser survey was 28.5 m wide by 26.5 m tall and had 18 arms, while for the Tektite experiments it was 50 m wide by 38 m tall and had 12 arms. Both were chosen to be sufficiently long as to exceed the mission duration of $T = 1500$. The robot is not permitted to allocate any samples until it has completed the initial coarse survey, so that it can first construct the semantic map and send some queries to learn the reward model before making any (irrevocable) sampling decisions.

For the experiments, we used the same reward map as for adaptive target search, meaning that the objective was to collect samples nearby Gorgonia specimens.[11] We set a budget of $B = 150$ samples, and compared the improvement of autonomous sample selection with each of the query selectors against the non-autonomous solution of simply collecting 1 sample every 10 seconds uniformly along the pre-planned trajectory (recall that $T = 1500$).

6.5.3 Results & Discussion

Numerical results from all experiments are presented in Tables 6.1 and 6.2. We discuss each experiment individually below.

Adaptive Target Search The results for the adaptive target search experiments are presented visually in Figure 6.8. They show that the proposed system is able to locate up to 64% more targets of interest than a pre-planned survey (Tektite Reef; $T = 3000$). Across all experiments, the risk-based EBRM query selection approach scales the best as the query response time increases (communication bandwidth decreases). This is because it effectively prioritizes the most useful queries. Note that even at the lowest bandwidth level (825 bps), at which it takes 8 minutes for each query, the EBRM-based adaptive target search system is still able to locate 38% more targets of interest with *only 3 queries* over the course of the

[10] Note that the planner is constrained to keep previously collected samples in δ_n; it cannot choose to ignore or discard them after collection.

[11] This could represent, for example, collecting water samples to inspect for pathogens.

Table 6.1: Mission Reward Collection Rate Increases (%) – Booby Rock

	Spatial Predictor	Oracle						KNN					
	Data Transfer Rate (kbps)	396	13.2	6.6	3.3	1.65	0.825	396	13.2	6.6	3.3	1.65	0.825
Experiment Type	Query Selector												
Adaptive Search	Every Obs.	40.4	—	—	—	—	—	31.3	—	—	—	—	—
	EBRM (*Ours*)	—	***33.8***	***29.3***	***25.4***	***22.9***	***14.2***	—	25.0	***21.9***	***20.2***	***19.5***	***14.2***
	Most Recent Obs.	—	33.3	27.0	20.7	14.8	6.6	—	***25.3***	21.0	15.1	9.9	3.8
	Semantic Diversity	—	32.6	22.3	10.1	3.7	1.3	—	24.8	18.1	8.3	1.4	-0.1
Longer Search	Every Obs.	13.6	—	—	—	—	—	9.7	—	—	—	—	—
	EBRM (*Ours*)	—	***12.8***	***13.0***	***10.8***	***8.4***	***7.0***	—	9.0	***8.7***	***7.1***	***6.1***	***5.7***
	Most Recent Obs.	—	12.7	11.6	9.0	6.7	2.8	—	***9.3***	8.2	6.6	3.7	-0.1
	Semantic Diversity	—	12.5	11.4	6.9	1.1	-3.8	—	8.5	7.5	3.9	-1.8	-5.9
Sample Selection	Every Obs.	116.9	—	—	—	—	—	91.5	—	—	—	—	—
	EBRM (*Ours*)	—	62.8	50.2	***51.0***	***62.3***	***49.8***	—	51.8	40.1	***39.4***	***44.6***	***32.4***
	Most Recent Obs.	—	***81.7***	***66.2***	50.5	36.5	21.4	—	***58.3***	***46.5***	32.8	22.9	12.1
	Semantic Diversity	—	76.4	52.5	28.3	15.6	12.4	—	56.4	38.7	20.9	10.5	6.0

Table 6.2: Mission Reward Collection Rate Increases (%) — Tektite Reef

	Spatial Predictor	Oracle						KNN					
	Data Transfer Rate (kbps)	396	13.2	6.6	3.3	1.65	0.825	396	13.2	6.6	3.3	1.65	0.825
Mission Type	Query Selector												
Adaptive Search	Every Obs.	86.8	—	—	—	—	—	63.1	—	—	—	—	—
	EBRM (*Ours*)	—	***75.0***	***66.1***	***58.5***	***60.1***	***49.5***	—	52.6	47.2	***43.7***	***43.2***	***37.7***
	Most Recent Obs.	—	71.2	64.2	55.4	48.8	38.5	—	***53.9***	***47.9***	42.7	35.3	26.7
	Semantic Diversity	—	68.2	58.4	47.2	35.3	32.5	—	51.2	43.0	35.7	24.1	21.6
Longer Search	Every Obs.	95.1	—	—	—	—	—	67.9	—	—	—	—	—
	EBRM (*Ours*)	—	***84.9***	***82.6***	***78.2***	***79.8***	***70.1***	—	***64.1***	***60.7***	***55.8***	***55.5***	***48.9***
	Most Recent Obs.	—	82.3	76.5	65.4	58.2	47.8	—	63.1	59.1	51.6	44.0	36.1
	Semantic Diversity	—	82.0	73.4	61.1	47.3	33.0	—	61.9	56.9	47.8	37.7	25.4
Sample Selection	Every Obs.	143.6	—	—	—	—	—	103.1	—	—	—	—	—
	EBRM (*Ours*)	—	***109.9***	95.7	***84.2***	***83.7***	***68.5***	—	***74.8***	***65.0***	***58.0***	***55.5***	***40.0***
	Most Recent Obs.	—	109.6	***96.8***	81.2	65.0	46.0	—	72.9	63.6	53.3	42.2	29.6
	Semantic Diversity	—	105.0	87.6	67.1	45.2	38.5	—	69.7	56.5	43.2	32.0	25.2

25 minute mission. As the response to the 3rd query is only received 1 minute before the mission end, most of this improvement comes from careful selection of the first two queries.

Adaptive Sample Selection The results for adaptive sample selection mirror that of adaptive target search, with the proposed system collecting up to 75% more samples at targets of interest (Tektite Reef; $T = 3000$) than the non-adaptive baseline, and with the EBRM risk-based query selector demonstrating superior performance for slow query response times (low bandwidth levels). However, we note a failure case in the Booby Rock experiment where the EBRM query selector performs worse with more queries (short query response times). This is because, referring to Figure B.3, sea fans are relatively highly concentrated within the "yellow" semantic class in the right half of the site, but not on the left. EBRM-based query selection avoid excessive queries for semantic classes that is has already learned concentration parameters for, but this is ineffective when these concentration values are spatially heterogeneous, as in this case. The root of this failure is the beta-concentration reward model, which cannot capture this heterogeneity in its parameters.

Oracle Spatial Prediction We additionally repeat the previous experiments replacing the K-nearest neighbors spatial prediction model with an "oracle" predictor that has complete knowledge of the semantic map *a priori*. This provides insight into how much better the system might be able to perform with more sophisticated spatial prediction. These results are presented in Figures 6.9 and 6.11, respectively. In both mission types, this complete semantic map information provides a moderate improvement in reward collection rates.

Runtime Analyses Lastly, we investigate how the training time of the beta-concentration reward model and runtime of the hotspot-based planner compare to baseline methods. The baselines are logistic regression and a naïve greedy planner which greedily constructs paths out of unit steps in the 8-connected neighborhood of the robot's current location. We choose this greedy planner as reference not because it works well, but because it is very computationally cheap; such greedy planners are often used for rollouts in more complex planners like Monte Carlo Tree Search [20]. The results for each are presented in Figures 6.12 and 6.13, respectively. They demonstrate that the analytical updates to the beta-concentration reward model are orders of magnitude faster than updating a simple logistic regression model, while the hotspot planner runs nearly as fast as a greedy planner. This is because the cost of identifying the optimal reward threshold and creating the hotspot graph is modest, and greedy planning over this graph produces a much better plan and with far fewer steps.

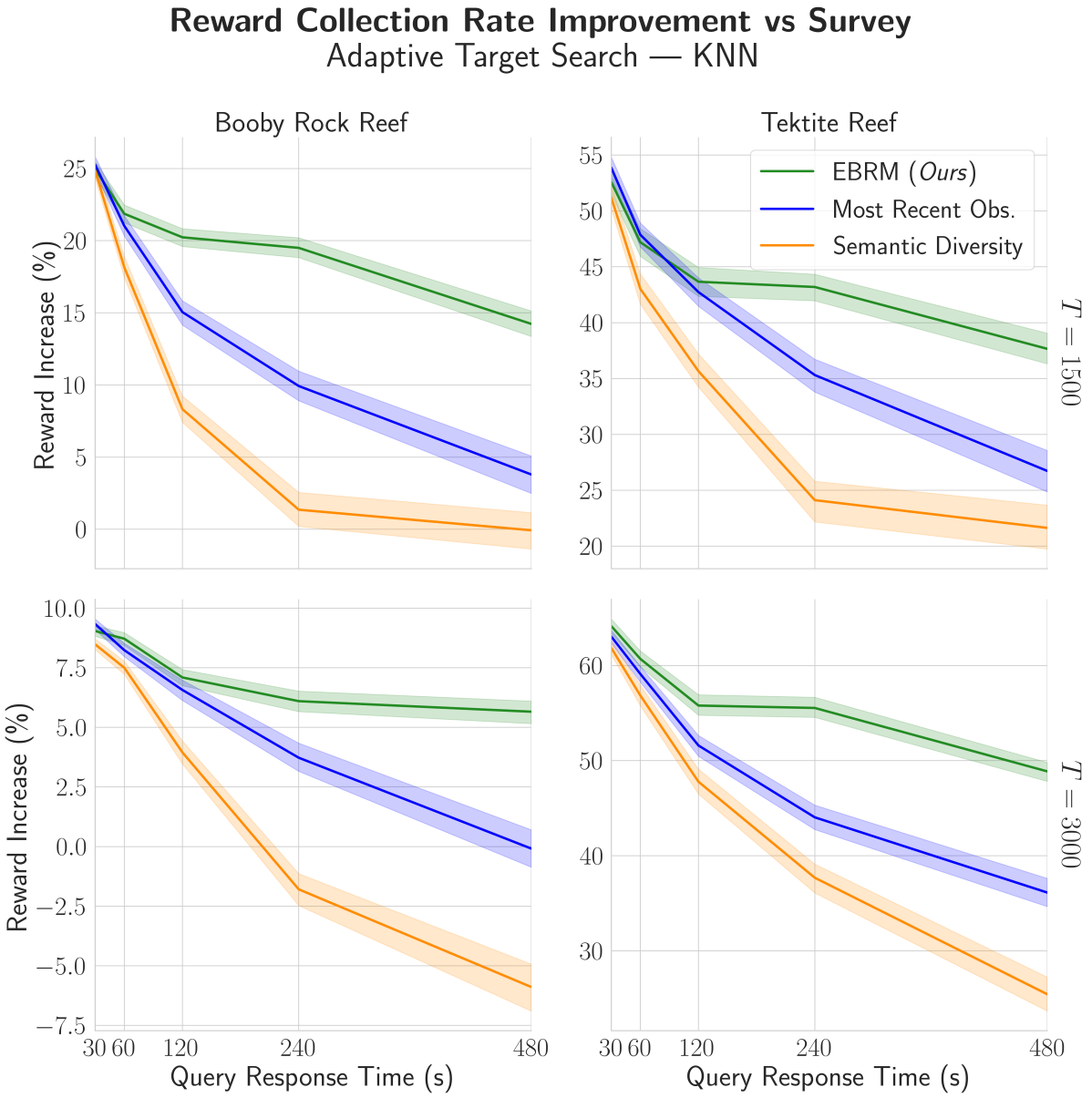


Figure 6.8: These plots show the percentage increase in reward collected by human-robot collaboration in adaptive target search when compared to a fixed survey, using different query selection algorithms. For short query response times (high communications bandwidth), all query selectors provide similar improvements, but the EBRM risk-based query selection approach performs significantly better as the query response time increases (i.e., as bandwidth decreases). Note that the extended duration mission $T = 3000$ provides enough time for a pre-planned survey to visit 75% of the Booby Rock map, so it is unsurprising that there is little advantage to autonomous target search in this case.

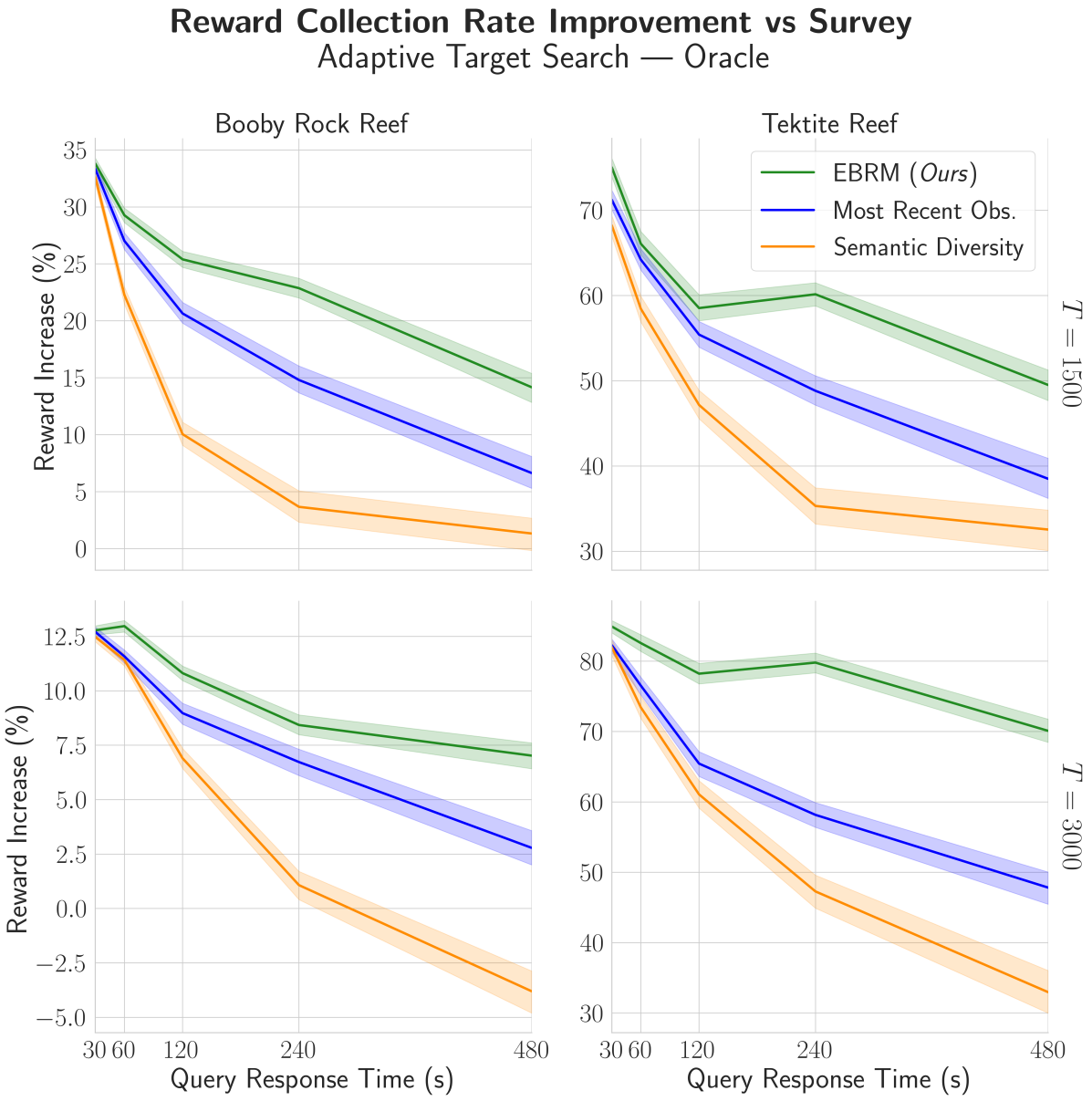


Figure 6.9: When replacing the K-nearest neighbors spatial prediction with an oracle that has complete knowledge of the semantic map, the improvement from using the proposed system increases by around 25% to 50% across all experiments. The relationships between query selectors and across query response times is largely unchanged, although the EBRM query selector's performance for low query response times is slightly improved relative to the baseline selectors.

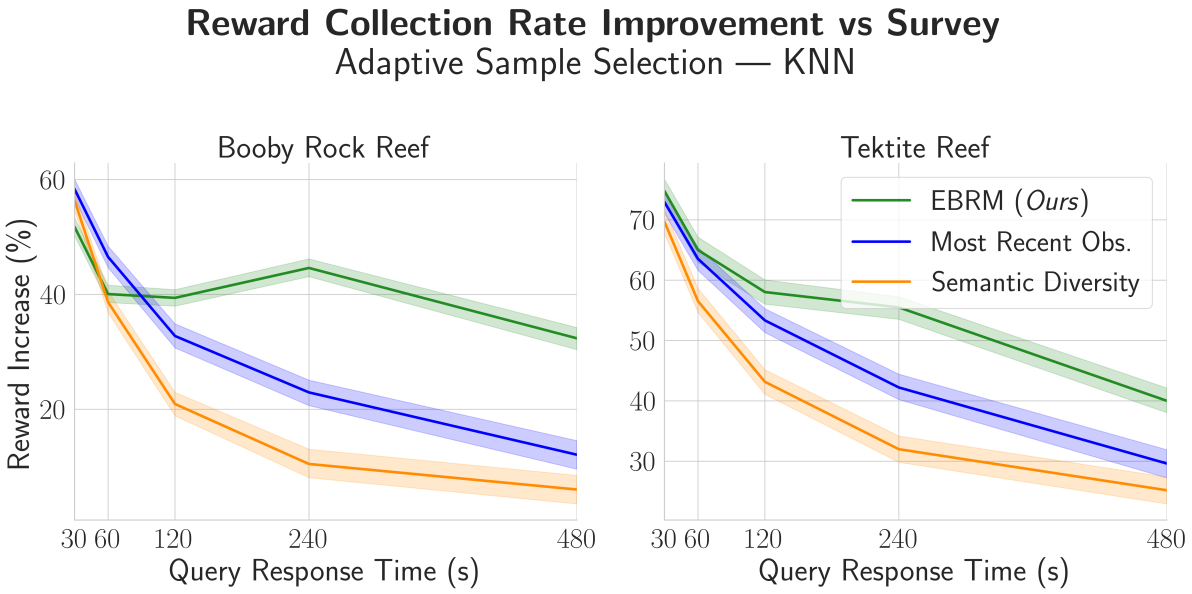


Figure 6.10: These plots show the percentage increase in reward collected by human-robot autonomous exploration in adaptive sample selection when compared to uniform sampling along the pre-planned trajectory, using different query selection algorithms. The risk-based EBRM approach makes better use of the few queries available as query response time increases; interestingly, however, it performs worse than the baselines on the Booby Rock site with many queries, because the concentration of targets of interest within the various semantic classes is spatially heterogeneous; see the main text for details.

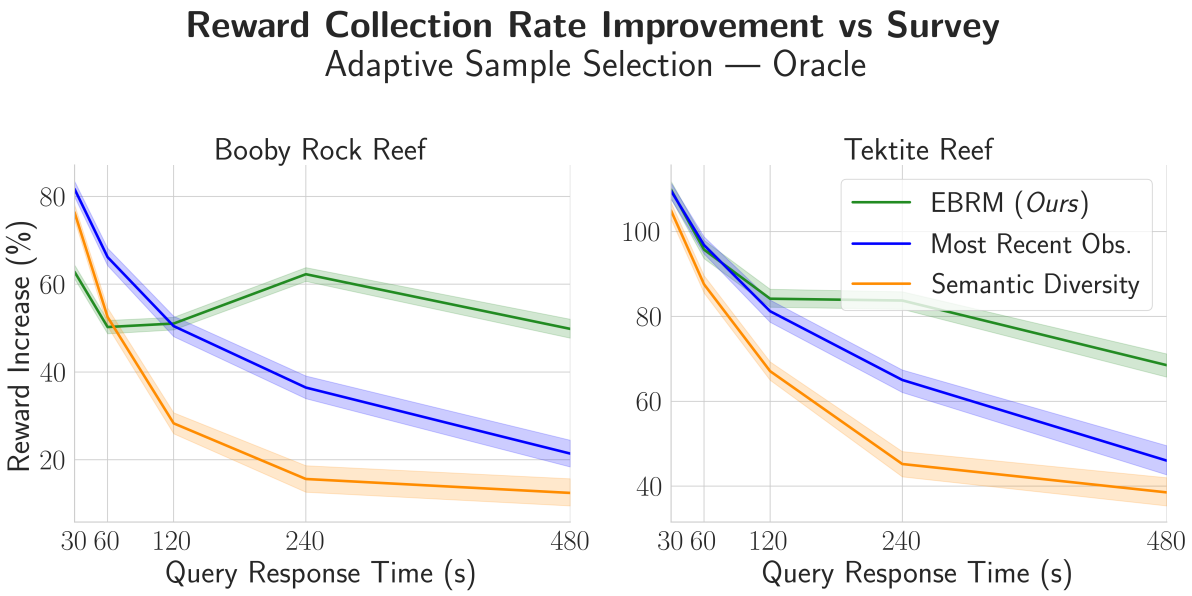


Figure 6.11: As in Figure 6.9, there is a 25 to 50% increase in reward collected when the robots are given complete information about the semantic map in advance, while the relative performance of the various query selectors is the same as in Figure 6.10.

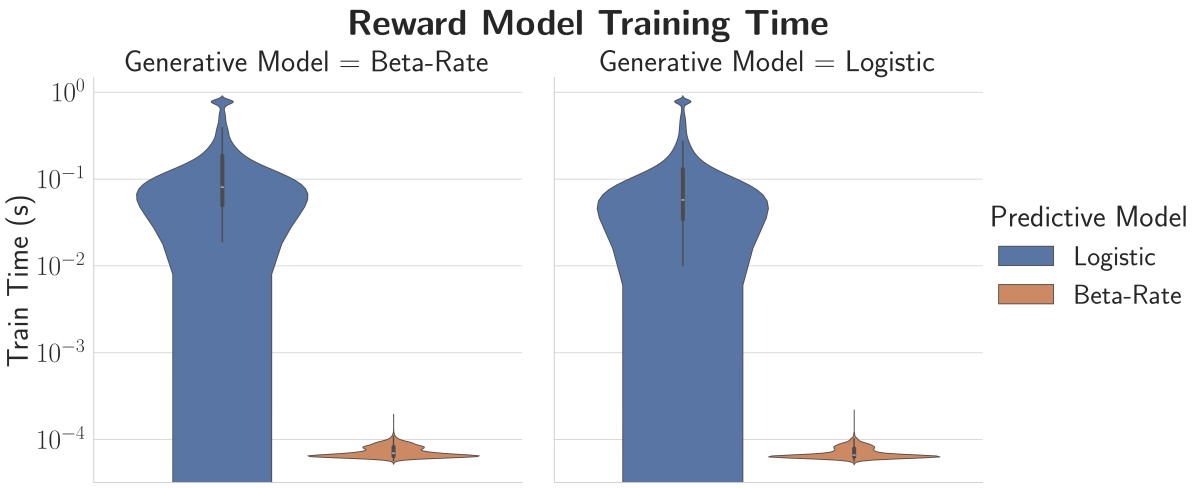


Figure 6.12: Here we compare the training time of a logistic regression model with the proposed beta-concentration model. The logistic regression model was trained as in Figure B.1. Note the log-scale; while the training time of the logistic model is not long for training once, it makes producing dozens of counterfactual reward models quite expensive. Conversely, the beta-concentration reward model performs similarly well but can be trained in orders of magnitude less time.

6.6 Summary

In terms of performance, the proposed system increased the robot's rate of finding/sampling targets of interest by up to 64%/75% over traditional pre-planned surveys or random sampling, respectively. When limited to very few queries, between 25% to nearly 100% of this improvement came from the use of EBRM-based query selection to identify which human-provided labels would have the greatest potential to improve the robot's trajectory. Positive improvements were generally found across all mission types and experiments, however the impacts of adaptive exploration are much more significant when the area of operations is *much* larger than can be covered during the mission duration, and when the patterns between semantic classes and targets of interest are consistent across the map.

As expected, the difference between query selection strategies is insignificant at high bandwidth levels across most experiments, where the large quantity of queries makes up for those lacking in quality. However, the AsympGreedy-EBRM query selector is uniquely capable of skipping queries expected to have very low values. This would save time and effort on behalf of the human supervisor, who would otherwise be expected to answer a proportionally higher number of queries at higher bandwidth levels, and is made possible by the risk-based query utility measures produced by the EBRM approach.

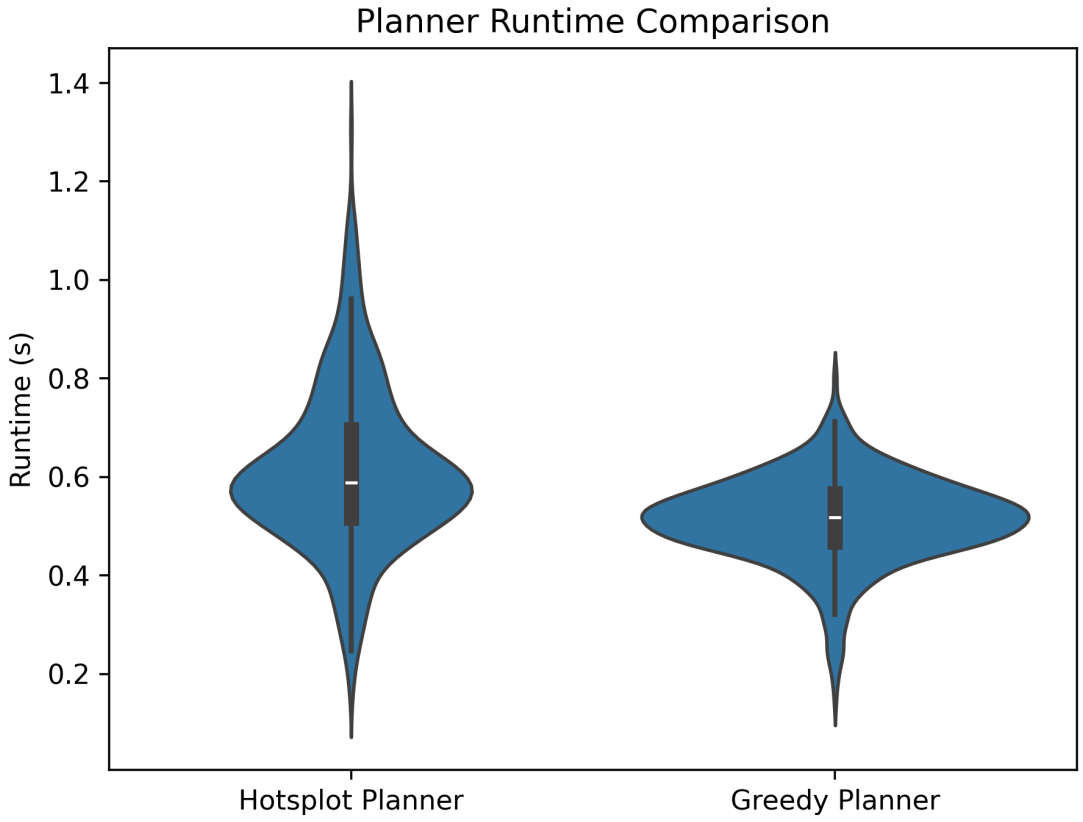


Figure 6.13: The hotspot based planner runtime is on approximately the same scale as a basic greedy planner which simply chains short steps across the map. The longer tail of the hotspot planner represents cases where a large number of small hotspots are identified,

Chapter 7

Conclusions & Future Work

This thesis enables large-scale vision-based exploration with dynamic objectives in completely unfamiliar marine environments and with severely limited communications bandwidth. It does so by presenting contributions that solve the problems associated with human-multi-robot collaboration in such contexts, as presented in Chapter 1. We begin by reviewing these contributions and highlighting their respective roles in improving human-multi-robot collaborative underwater exploration.

Chapter 3 presented a solution for matching the egocentric semantic representation learned by multiple robots, thereby enabling them to share semantic maps describing their environment. The approach was more flexible and robust to communication failures and bandwidth limitations than previous approaches, making it well suited as the foundation for multi-robot collaboration in autonomous vision-based exploration.

Chapter 4 presented the *DeepSeeColor* algorithm for real-time color correction of underwater imagery, which produces color corrected images better suited for both machine and human analysis. In particular, it enables higher quality semantic representations to be produced and, by extension, better semantic maps. This addresses a key practical limitation in the use of unsupervised semantic models for underwater vision, and particularly for autonomous exploration.

Chapter 5 presented a novel risk-based online learning framework, which generalizes previous approaches and is applicable to optimal online learning across a wide range of applications. It also introduced the *AsympGreedy-EBRM* algorithm, which overcomes the myopia of most previous risk-based online learning algorithms while remaining computationally tractable. Furthermore, AsympGreedy-EBRM is capable of handling problem complexities such as communication costs and feasibility constraints, as well as the "cost" of a robot interrupting a human with questions, unlike previous online learning algorithms.

Finally, Chapter 6 introduced a novel beta-concentration reward model and hotspot-based

planner, the former of which effectively models the co-occurence relationship between learned semantic classes and arbitrary targets of interest, while the latter enables producing high-quality plans with minimal computation. These were designed to enable efficient generation of counterfactuals, the basis of risk-based online learning strategies, and we demonstrated that using these together with the AsympGreedy-EBRM algorithm to guide human-robot communication enabled significant increases in mission effectiveness when compared to traditional methods.

The findings of Chapter 6, in particular, suggest that the proposed human-multi-robot collaborative approach to vision-based marine exploration is ready to be tested in practice. They prove that useful semantic maps can be produced online from color corrected images with unsupervised spatiotemporal topic models, and that useful reward models can be learned online from few queries; together, these capabilities enable an adaptive robotic explorer to collect up to 75% more useful data than it would with the kinds of pre-planned surveys that are currently favored in practice. All of the algorithms presented in this thesis are computationally efficient enough to run onboard real AUV platforms (e.g., the CUREE vehicle [166]), and the communications bandwidths tested are representative of the data transfer rates expected when using such platforms. In fact, the results indicate that a robotic explorer can use a relatively small fraction of the maximum feasible underwater communications bandwidth while still achieving significant performance gains, through the careful query selection of AsympGreedy-EBRM.

While I have not yet had the opportunity to validate this system in real-world field trials, I am confident that approaches like the one proposed in this thesis will soon become the norm for marine exploration. I further expect that many of the contributions presented here will remain relevant as further advances are made which replace various components of the proposed system. For example, the semantic representation matching approach from Chapter 3 can be applied to matching representations learned by newer (e.g., deep learning based) unsupervised semantic mapping systems, while the AsympGreedy-EBRM query selection approach described in Chapters 5 and 6 can be used alongside a wide range of alternative reward models, planners, or semantic models. As practitioners build more trust in autonomous and adaptive robotic explorers over the course of repeated successful missions, interest in developing such alternative models and planners will likely grow towards developing more specialized and sophisticated exploration behaviors. Ultimately, more effective human-multi-robot collaborative marine exploration can provide societal benefits through enabling large fleets of robots to more efficiently explore and monitor the Earth's coral reefs and other marine environments. This will bring new discoveries to the attention of human experts as quickly as possible, and likewise enable those experts to intervene, as necessary,

in time to protect and preserve these critical ecosystems.

Lastly, I would like to note that the contributions of this thesis are relevant to many other applications. For example, space exploration and subterranean exploration have many of the same considerations as marine exploration, including dynamic mission objectives with *a priori* unknown phenomena of interest and severely limited communications bandwidth. All of the contributions (aside from DeepSeeColor) can be used to improve human-multi-robot vision-based exploration in these environments, with few, if any, modifications. Meanwhile, DeepSeeColor could be useful in many underwater vision systems, while AsympGreedy-EBRM demonstrated superior performance to baseline methods in applications as ranging from assigning patients to treatments in a multi-objective clinical trial to determining the optimal sales price of a product. In conclusion, I am hopeful that the ideas, contributions, and methods presented in this thesis will inform future progress in vision-based human-multi-robot marine exploration as well as many other diverse applications across similarly diverse fields of research.

7.1 Limitations & Future Work

There is still yet more work required to see the contributions presented in the previous chapters make an impact in real-world autonomous underwater exploration missions. The general lack of field validation of the preceding contributions is evidence of the significant practical challenges in underwater autonomy. Over the course of four trips to the US Virgin Islands for fieldwork, it was only on the most recent one which I was able to begin testing *in situ* human-robot collaboration to enable adaptive mission behaviors. Thus, we have not been able to properly characterize the human-experience component of the human-multi-robot system.

In previous trips, a major limitation was the navigational performance of our AUV platform; it struggled to reliably return to previous coordinates or track trajectories over long distances. This is a major limitation to the autonomous behaviors presented in Chapter 6, which all depend on predicting what the robot will see, which in turn depends on where the robot will be. One way to improve underwater navigation is with more robust underwater vision. Underwater camera calibration was a perennial challenge in our lab, as we experimented with different camera, lenses, and ports, and most camera calibration software is not well-tested for use with underwater vision. Many simultaneous localization and mapping systems perform best with high quality depth imagery, and this is particularly hard to produce with poorly calibrated underwater cameras. This is a significant limitation of DeepSeeColor, which relies on dense depth images in order to learn and apply the color

reconstruction model. Some of our collaborators have recently explored improved camera calibration models which may help to mitigate these issues [227].

Perhaps an even larger barrier than navigation was the substantial investment of time and resources required to do vision-based underwater exploration fieldwork. As my work is most applicable to domains such as coral reefs and the deep ocean, the murky waters off the near-shores of Cape Cod were less than ideal testing grounds; this is one factor which drove us towards fieldwork in the US Virgin Islands. The cost and logistical complexity of such trips limited our time in the field, however, so our risk-minimizing strategy was to focus on collecting datasets which could be used for a variety of simulation experiments. While this worked out well in providing some extraordinary datasets from which we were able to produce high-resolution 2D and 3D reef constructions (e.g., see Figures 6.3 and 6.6), it deterred us from more ambitious autonomous experiments, which would have been more high-risk/high-reward propositions.

In the following subsections, I provide some thoughts on future directions to enable further progress in autonomous marine exploration.

7.1.1 Human-Robot Online Reward Learning Studies

One of the most interesting experiences I had in conducting this research was watching a WHOI biologist use an interface I developed to respond to our AUV's image queries, just as described in Chapter 6. While the data from this experiment had limited utility due to the aforementioned navigational issues,[1] observing the scientist's behavior when interacting with the system made it easily worthwhile. For example, it was clear the robot's request for a query every 30 seconds was somehow both too slow and too fast; the transmission time was long enough that the scientist would occasionally get distracted while waiting for the next query, which in turn would cause them to delay noticing when it did arrive. Constantly responding to the queries also interfered with their conversations and kept their attention away from other matters.

It is clear that future work into human-multi-robot collaborative exploration must take a deeper look at the human experience of the system. Rather than viewing the supervisor as purely a resource, the system must consider how to make effective use of their time while imposing a minimal cognitive and operational burden. For example, we may find that the available bandwidth of a supervisor is even lower than the actual communications rate if they have many competing priorities. Batching queries may be an approach to minimize context-switching on the side of the human supervisor, as labelling many queries at once may be

[1] The survey data collected during this adaptive sampling selection mission was, however, useful for constructing the Tektite reef map presented in Figure 6.6.

a better experience than being constantly interrupted for individual requests. Determining when and how to best engage the human supervisor is an thus a key priority for future work.

7.1.2 Large-Scale Datasets for Marine Exploration

A major challenge in developing autonomous vision-based systems for underwater exploration was the lack of suitable simulation environments and training data. Without realistic training data and simulation environments, it is very difficult to bridge the gap between simulated and real-world autonomous behaviour. This was a key factor in the lack of successful field validation; in fact, the contribution of Chapter 4 (DeepSeeColor) was largely the result of trying to fix issues that only began to appear in real field deployments, as simulators such as AirSim, used in Chapter 3, do not model underwater image formation, and many underwater image datasets have been processed offline to remove such distortions. With better dataset availability and simulators which accurately model underwater vision, more of these issues could be discovered and resolved before disrupting real field experiments.

In Appendix C, I present some notes on existing reef datasets and their limitations. Producing new datasets with the kinds of qualities discussed therein as missing or under-represented would enable far more extensive and realistic simulation experiments, which should translate into more successful real world deployments. Furthermore, large-scale, geolocalized reef datasets with high quality imagery make it much easier for more roboticists to contribute to the development of coral reef monitoring algorithms without needing to have their own AUVs and travel to appropriate deployment areas, much in the same way that datasets like KITTI [228] have enabled researchers to advance self-driving technology without needing to have their own test vehicles.

7.1.3 Multi-Robot Coordinated Online Information Sharing

As the communication required for matching semantic models with the proposed approach scales linearly with the number of robots, it becomes infeasible for very large fleets. This motivates finding an alternative solution that scales sublinearly; as we do in online reward learning with EBRM, a robot could selectively decide whether to share its semantic map and model based on whether this information is expected to have utility for the rest of the fleet. Accordingly, we plan to model sharing and matching semantic models as an explicit communication action available to robots, such that they decide when to take this action using the same risk-based strategies used for online reward learning. This action may be considered in parallel to other types of communication, such as image queries for reward learning.

This leads one to imagine higher-level reasoning about risk-based inter-robot communication; for example, a robot might find that its semantic model produces stronger correlations between semantic classes and targets of interest than those belonging to other robots. In a risk-based communication framework, the robots could reason about the utility of merging or even replacing their semantic models, or individual semantic representations. In effect, this framework would create a "marketplace" of information, wherein each robot can be constantly reasoning about which of its information may be useful to other robots (or to a human supervisor), and about what useful information it may be able to acquire from other robots or humans. The ability of robots to participate in this marketplace is largely enabled by the contributions in Chapters 3 and 5.

A barrier to realizing this vision of an information marketplace is that each robot collects far more data than it can feasibly share with the rest of the team. This motivates exploring how individual robots can achieve one of two tasks:

1. Compactly describe what kinds of information they have available to send, or
2. Compactly describe what kinds of information they would value receiving.

Information in these contexts may include reward labels, images, semantic models, or semantic maps. The *compactness* is critical when operating in marine environments where bandwidth is limited; the information needs in sufficient detail to enable estimating its value, but these tasks are not meaningful if the amount of bandwidth used to describe the information is comparable to the bandwidth actually used for sharing it. Through successfully achieving either of these tasks, however, it would become feasible for robots to identify what kinds of information can be requested, or what information is most valuable to the team as a collective, respectively. These would enable globally optimal information sharing subject to bandwidth constraints, wherein each robot bids on information it desires or can provide, and bids are filled in order of value.

7.1.4 Broader Risk-Based Online Learning Directions

The successes of AsympGreedy-EBRM based algorithms across the experiments in Section 5.5, the clinical trial experiment in Subsection 5.5.2, and the adaptive exploration missions in Chapter 6 motivate developing EBRM algorithms for other impactful real-world applications. Furthermore, designing new EBRM candidate policy sets, and techniques to compute the process risk of policies beyond 1-step lookahead policies, will enable the development of new EBRM-based algorithms that may provide further performance improvements upon Greedy-EBRM, and represents a new direction of research in online learning theory. A

promising initial step in this direction is the incorporation of information-directed sampling techniques [19] and the related theoretical analyses into new techniques to estimate changes in epistemic risk.

There are a wide variety of real-world online learning problems with complexities that could benefit from EBRM approaches. For example, problems with continuous action spaces, typically modelled as linear bandits, should see similar improvements through EBRM-based algorithms. However, they introduce new computational challenges; in particular, the one-step lookahead candidate policy set is infinite given a continuous action space, and so the agent cannot check every policy. Developing techniques to identify the one-step lookahead policy with minimal process risk in a continuous policy set is thus one research priority.

Action costs and cost budgets are additional problem complexities that have been the subject of limited study but have valuable real-world applications. Sensing or communicating information generally has costs, which may be based on energy used or time spent communicating, and real-world systems have limits on these costs. EBRM-based approaches are well suited to capture these costs and budgets in the action-based reward and feasibility criterion, respectively. Relating back to the first proposed area of future work, EBRM-based approaches can directly model costs or rewards associated with interacting with a human supervisor, reflecting factors such as the cognitive load of the request or the frequency of interactions.

Appendix A

EBRM Examples & Proofs

A.1 Contrasting Risk and Regret

A **regret function** measures how much less reward *was* attained, in a posterior sense, by some policy than would have been by the X-optimal policy. That is, if $A_t^\pi = \{a_\tau^\pi\}_{\tau=1}^t$ is the sequence of actions generated by a policy π from an initial belief state S_0, and $S_t^\pi = \{b_t^\pi, \xi_t^\pi\}$ is the corresponding terminal belief state, then for some hidden outcomes X the instance regret is

$$\mathfrak{R}(A_t^\pi; S_0, X) \coloneqq V^{\pi_X^\star}(S_0; X) - \left(\gamma^t R(b_t^\pi) + \sum_{\tau=1}^{t} \gamma^\tau R(x_\tau, a_\tau^\pi)\right). \tag{A.1.1}$$

The expected and Bayesian regrets are similarly defined,

$$\overline{\mathfrak{R}}(A_t^\pi; S_0, \theta) \coloneqq \mathbb{E}_{X|\theta}[\mathfrak{R}(A_t^\pi; S_0, X)], \tag{A.1.2}$$

$$\overline{\overline{\mathfrak{R}}}(A_t^\pi; S_0, b_t) \coloneqq \mathbb{E}_{\theta|b_t}\left[\overline{\mathfrak{R}}(A_t^\pi; S_0, \theta)\right]. \tag{A.1.3}$$

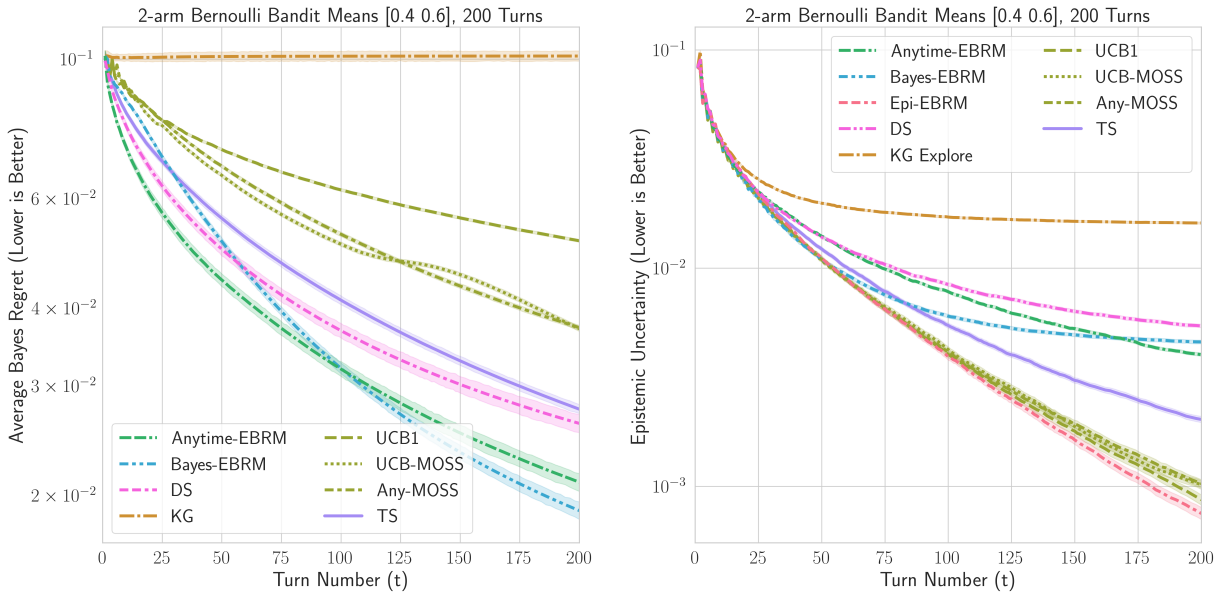
These quantities are defined with respect to a particular sequence of actions and observations made by the agent, unlike risk which is computed based on all possible sequences of actions the agent *might* perform from some initial state. The risk of a policy relative to the X-optimal policy is therefore equal to the *expected* amount of regret that will be incurred by that policy.

Once incurred, regret is "permanent"; by the Tower rule, the Bayesian regret of a sequence of actions does not change, in expectation, due to future actions and observations:

$$\mathbb{E}_{x|b_t}\left[\overline{\overline{\mathfrak{R}}}(A_t^\pi; S_0, b_{t+1}^{x,a})\right] = \overline{\overline{\mathfrak{R}}}(A_t^\pi; S_0, b_t),\ \forall a \in \mathcal{A}. \tag{A.1.4}$$

Conversely, the agent can reduce its risk (i.e., its future regret) by taking an action providing information enabling it to make better decisions in the future.

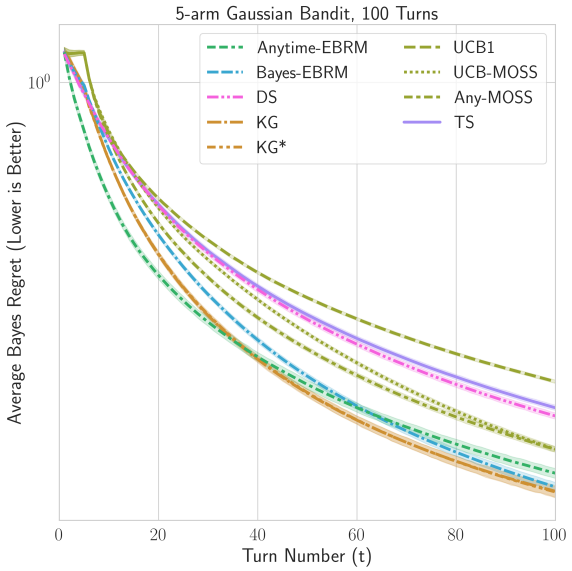
A.2 Additional Experiments



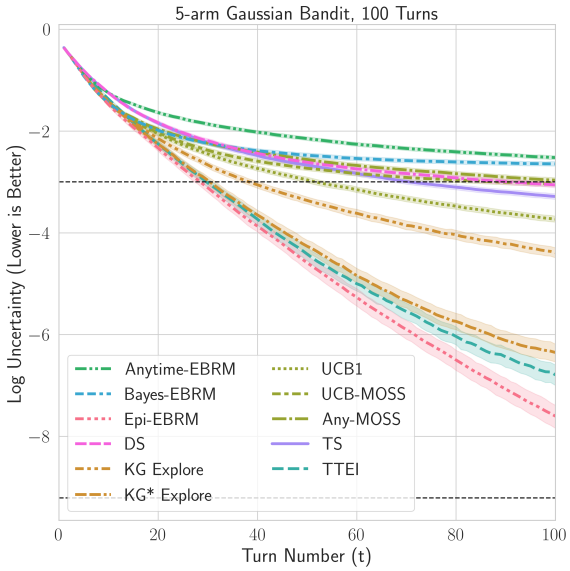
(a) The AsympGreedy-EBRM approaches are highly successful at reducing the average Bayes regret, even over relatively few trials.

(b) In this simple setting, the UCB algorithms are nearly as effective at reducing epistemic uncertainty as Epi-EBRM.

Figure A.1: The results of this 2-arm Bernoulli bandit experiment from [177] clearly demonstrates the exploration-exploitation trade-off, where the algorithms with the lowest average instance regret tend to have the higher epistemic uncertainty, and vice-versa. In particular, observe how Bayes-EBRM rapidly switches from exploration to exploitation around $t = 75$ in order to maximally leverage its accumulated information over the remaining time.

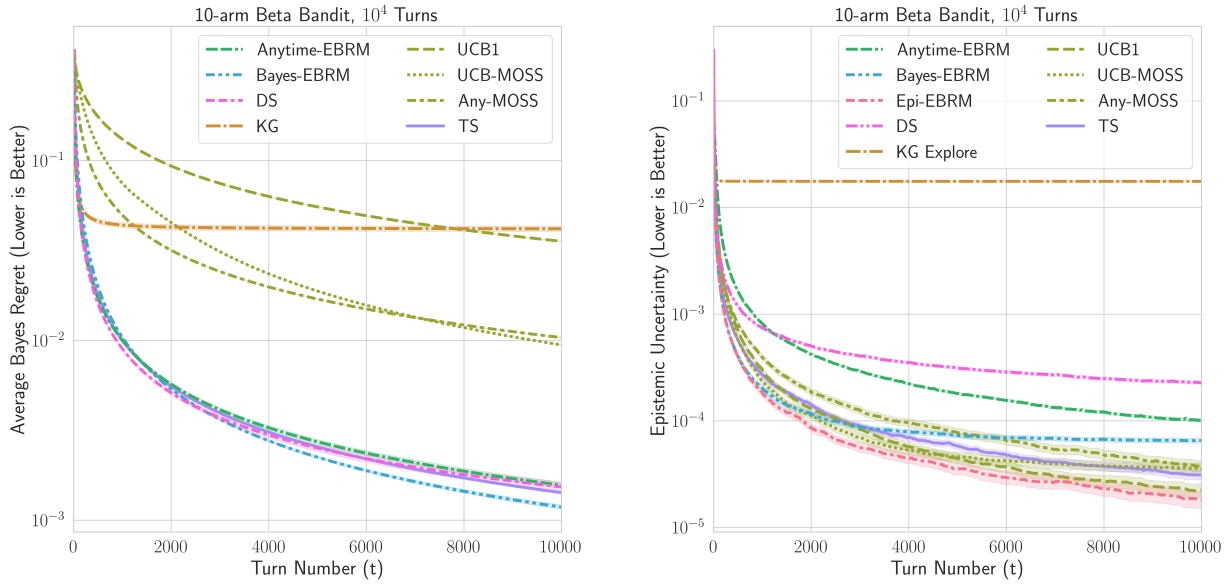


(a) The y-axis shows average sample regret, the bandit optimization metric for which lower is better. Asymp-EBRM has the least average regret initially, before being overtaken by KG and Bayes-EBRM.



(b) The y-axis shows log uncertainty; lower values indicates higher confidence in the best-arm identification. The upper dashed line represents 95% confidence in having found the best arm, while the lower dashed line represents 99.99% confidence.

Figure A.2: Performance of algorithms in bandit optimization and best-arm identification while making decisions in 5-arm Gaussian Bandit problem over 100 turns. Shaded regions indicate 95% confidence intervals over 2000 trials.



(a) TS, DS, Asymp-EBRM and Bayes-EBRM perform nearly identically over the first 4000 turns, after which Bayes-EBRM begins to significantly outperform the rest.

(b) The y-axis shows epistemic uncertainty; lower values indicate better BAI performance. Epi-EBRM achieves the lowest epistemic uncertainty across trials.

Figure A.3: Performance of algorithms in BDO and BAI objectives while making decisions in a 10-arm Beta bandit problem over 10^4 turns. Shaded regions indicate 95% confidence intervals over 2000 trials.

A.3 Example Clinical Research Trial Scenarios

In this section we describe some theoretical clinical research scenarios to highlight the limitations of existing online learning heuristics.

Problem 3 (Villar et al., [71])**.** *Consider 423 patients sequentially assigned to various treatments within a randomized controlled trial. Which assignments best address the competing objectives of (1) maximizing the number of successful patient outcomes by assigning as many as possible to the best treatments, and (2) achieving sufficient statistical power to detect a significant difference between treatments?*

The study which presented Problem 3 found that several leading online learning algorithms each failed to balance between these two objectives [71]. This is due to their inability to account for a second variable, like statistical power in a weighted objective (e.g. maximize a weighted sum of statistical power and successful patient outcomes). The EBRM approach presented in this work can directly model this type of objectives.

Problem 4. *The goal of a similar trial to that in Problem 3 is to determine which treatment to distribute nationally/globally to a much larger population of N patients. Which assignments result in the highest number of successful patient outcomes across all $(N + 423)$ patients who will receive treatments?*

Problem 4 is concerned with ensuring that the treatment recommended for broad distribution is the most effective one, or nearly so. This is a different metric than statistical power or confidence, and one which is not directly targeted by any existing algorithm. Furthermore, this problem introduces an explicit weighting; for $N = 10^4$, identifying a treatment that has a 1% higher success rate will result in 100 more successful outcomes across the patient population. In Section 5.5, we see that the EBRM approach results in the most successful patient outcomes in this type of scenario, for various N. Importantly, the EBRM algorithm does not require any "tuning" to achieve these results; it simply takes N as a parameter.

Problem 5. *As an extension to Problem 4, suppose that there is are costs to enrolling patients in the trial, and a unique cost for administering each type of treatment. What are the optimal trial size and treatment assignments in order to maximize successful patient outcomes given a budget constraint?*

Most existing online learning heuristics lack the ability to model heterogeneous costs across different actions (treatments). Recent work has begun to model such complexities (e.g., [229]), however the approach required developing another heuristic which focuses on maximizing reward per unit cost. The EBRM approach automatically takes into account various feasibility criterion, including those driven by action costs.

A.4 Proof of Section 5.3 Results

Proof of Theorem 1

First observe that, as a result of Eq. (5.3.2), the continuous mapping theorem [230] implies

$$\mathbb{E}_{x|b_t}[f(\cdot)] \xrightarrow[t\to\infty]{P} \mathbb{E}_{x|\theta}[f(\cdot)].$$

As the belief reward is bounded and continuous, then, also by the continuous mapping theorem, there exists some $c \in \mathbb{R}$ such that

$$R(b_t) \xrightarrow[t\to\infty]{P} c.$$

As the belief reward converges in probability to a constant as $t \to \infty$, we have that

$$\begin{aligned}\text{EpistemicRisk}(S_t) &= \mathbb{E}_{\theta|b_t}\left[V^{\pi^\star_\theta}(S_t;\theta)\right] - V^{\hat{\pi}^\star_t}(S_t) \\ &\xrightarrow[t\to\infty]{P} \sum_{n=1}^{\infty} \mathbb{E}_{x|\theta}\left[\gamma^{n-1} R(x, \pi^\star_\theta(S_{t+n-1}))\right] - \sum_{n=1}^{\infty} \mathbb{E}_{x|\theta}\left[\gamma^{n-1} R(x, \hat{\pi}^\star_t(S_{t+n-1}))\right]\end{aligned}$$

As each sum is independent of any future observations or hidden variables, they can be each be maximized by some deterministic open-loop policy $\hat{\pi} \in \hat{\Pi}$ and thus

$$\mathbb{E}_{x|\theta}[R(x, \hat{\pi}^\star_t(S_t))] \xrightarrow[t\to\infty]{P} \mathbb{E}_{x|\theta}[R(x, \pi^\star_\theta(S_t))].$$

Proof of Lemma 2

Let σ be the permutation function that satisfies $A' = \sigma(A)$, and let $X' = \sigma(X)$ be the corresponding permutation of X. First, we observe,

$$x_\tau \overset{\text{iid}}{\sim} P(x \mid b_t) \implies \Pr(X' \mid b_t) = \Pr(X \mid b_t).$$

Furthermore, denoting $y'_\tau = \Phi(x'_\tau, a'_\tau)$ and $y_\tau = \Phi(x_\tau, a_\tau)$ for $\tau = 1, \ldots, |A|$,

$$\begin{aligned}x_\tau \overset{\text{iid}}{\sim} P(x \mid b_t) &\implies \Pr(y'_{1:|A'|} \mid \theta, a'_{1:|A'|}) = \Pr\big(y_{1:|A|} \mid \theta, a_{1:|A|}\big), \\ &\implies b^{X',A'}_{t+|A'|}\big(\theta\big) = b^{X,A}_{t+|A|}\big(\theta\big), \quad \forall \theta \in \Theta.\end{aligned}$$

A.5 Proofs of Section 5.4 Results

Proof of Lemma 3

The adaptive monotonicity of $R(b)$ and the convexity of the expectation operator imply

$$\mathbb{E}_x\left[\max_{N\in\mathbb{N}_\Omega^K(\xi_{t+1}^a)}\hat{V}\big(N;S_{t+1}^{x,a}\big)\right]\geq\max_{N\in\mathbb{N}_\Omega^K(\xi_{t+1}^a)}\mathbb{E}_x\Big[\hat{V}\big(N;S_{t+1}^{x,a}\big)\Big]\geq\max_{N\in\mathbb{N}_\Omega^K(\xi_{t+1}^a)}\hat{V}\big(N;\mathbb{E}_x\big[S_{t+1}^{x,a}\big]\big).$$

The lemma follows from expanding Eq. (5.4.8) with the definition of each term and applying this inequality.

Proof of Proposition 3

We introduce Lemma 5 to simplify the proof.

Lemma 5. *The process risk of the best deterministic* 1*-step lookahead policy is bounded,* $\forall S_t$*,*

$$-\text{EpistemicRisk}(S_t)\leq\min_{a\in\mathcal{A}}\text{ProcessRisk}(\pi_{t,a};S_t)\leq 0. \tag{A.5.1}$$

The lower bound of Lemma 5 follows from, $\forall\pi$ and $\forall S_t$, (see Lemma 1)

$$0\leq\overline{\overline{r}}(\pi\|\pi_\theta^\star;S_t)=\text{EpistemicRisk}(S_t)+\text{ProcessRisk}(\pi;S_t).$$

The upper bound follows from Lemma 3, using the fact that $\text{ImmediateRisk}(S_t,\pi_{t,a})=0$ if $a=\hat{\pi}_{S_t}^\star(S_t)$. According to Lemma 5, the best 1-step lookahead policy has non-positive process risk, but by Lemma 3,

$$\begin{aligned}\text{ProcessRisk}(S_t,a)&=\text{ImmediateRisk}(S_t,a)-\gamma\cdot\text{ExpectedVoI}(S_t,a)\\&\geq\text{ImmediateRisk}(S_t,a)-\gamma\cdot\text{EpistemicRisk}\big(\mathbb{E}_{x|b_t}\big[S_{t+1}^{x,a}\big]\big).\end{aligned}$$

The proposition follows.

Proof of Theorem 2

Consider $\mathbb{E}[\mu_a]=\text{ProcessRisk}(\pi_{t,a};S_t)-\text{ProcessRisk}(\pi_{t,k};S_t)$, and by construction $\mu_a\geq 0$. Therefore,

$$\mathbb{E}[\mu_a]>0\implies\text{ProcessRisk}(\pi_{t,k};S_t)<\text{ProcessRisk}(\pi_{t,a};S_t). \tag{A.5.2}$$

Next, $\text{Var}[\mu_a] = \gamma^2(\sigma_a^2 n_a^{-1} + \sigma_k^2 n_k^{-1})$ follows from the independence of $\nu_a(n_a)$ and $\nu_k(n_k)$, which are generated from IID samples of the hidden outcomes. The theorem follows from Cantelli's inequality [231] applied to $\Pr(\mu_a - \mathbb{E}[\mu_a] < \mu_a)$.

Proof of Theorem 3

Following from the proof of Theorem 2, observe that $\forall n > 0$,

$$\max_{N \in \mathbb{N}_\Omega^K(\xi_{t+1}^a)} \hat{V}\left(N; S_{t+1}^{x,a}\right) - \hat{V}\left(N_{t+1}^a; S_{t+1}^{x,a}\right) \leq M_a \implies \nu_a(n) \leq M_a, \tag{A.5.3}$$

$$\implies \mu_a \leq (\text{ImmediateRisk}(S_t, a) - \text{ImmediateRisk}(S_t, k)) + \gamma M_k, \tag{A.5.4}$$

$$\implies \mu_a \geq (\text{ImmediateRisk}(S_t, a) - \text{ImmediateRisk}(S_t, k)) - \gamma M_a. \tag{A.5.5}$$

Next, we consider that for $n_a = n_k = n$, then $n\mu_a$ is equal to the sum of n IID samples of the difference in process risk between policies $\pi_{t,a}$ and $\pi_{t,k}$,

$$n\mu_a = \sum_{i=1}^{n} \Bigg[\text{ImmediateRisk}(S_t, a) - \gamma \left(\max_{N \in \mathbb{N}_\Omega^K(\xi_{t+1}^a)} \hat{V}\left(N; S_{t+1}^{x_i,a}\right) - \hat{V}\left(N_{t+1}^a; S_{t+1}^{x_i,a}\right) \right) - \left(\text{ImmediateRisk}(S_t, k) - \gamma \left(\max_{N \in \mathbb{N}_\Omega^K(\xi_{t+1}^k)} \hat{V}\left(N; S_{t+1}^{x_i,k}\right) - \hat{V}\left(N_{t+1}^k; S_{t+1}^{x_i,k}\right) \right) \right) \Bigg], \tag{A.5.6}$$

with $\mathbb{E}[n\mu_a] = n(\text{ProcessRisk}(\pi_{t,a}; S_t) - \text{ProcessRisk}(\pi_{t,k}; S_t))$. Then, applying Hoeffding's inequality,

$$\Pr(n\mu_a - \mathbb{E}[n\mu_a] \geq c) \leq \exp\left(\frac{-2c^2}{n(\gamma M_k + \gamma M_a)^2} \right). \tag{A.5.7}$$

The theorem follows from taking $c = n\mu_a$ and recognizing that the largest amount of regret which could be incurred by rejecting policy $\pi_{t,a}$ in favor of $\pi_{t,k}$ is $\gamma(M_a + M_k)$, and the probability of incurring this regret is $\Pr\left(\pi_{t,a} \in \arg\min_{\pi \in \Pi_t} \text{ProcessRisk}(\pi; S_t)\right) \leq \Pr(\mathbb{E}[n\mu_a] < 0)$.

Appendix B

Adaptive Exploration Supplemental

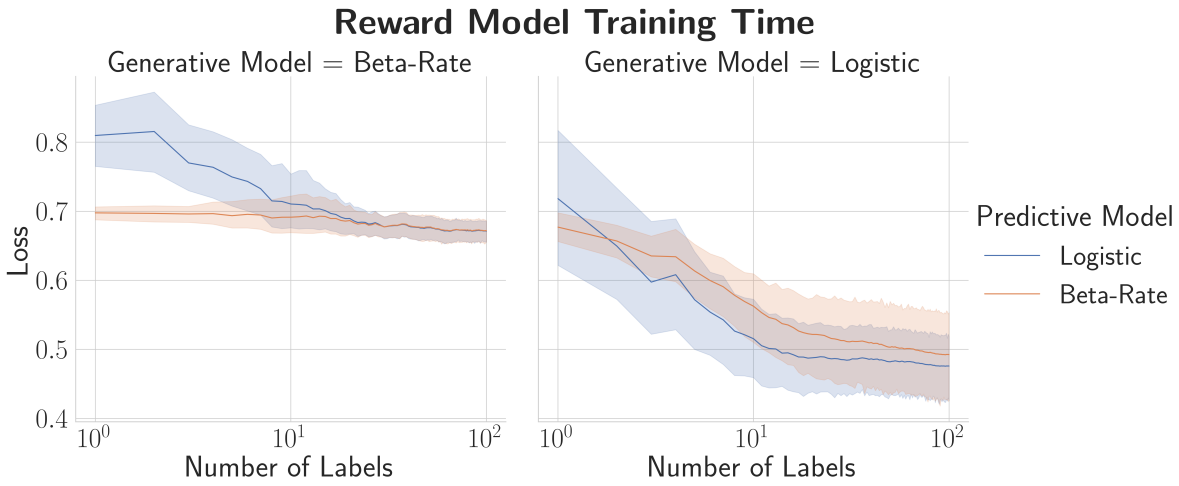


Figure B.1: We evaluate the loss of a logistic regression model and the proposed beta-concentration model on data produced by the corresponding generative model of each. As expected, each model achieves lower loss on data generated by its own generative model. Interestingly, however, the beta-concentration model does not perform significantly worse than the logistic model even on data generated from the logistic prior. The logistic regression model was trained with PyTorch, using the Adam optimizer with a learning rate of 3×10^{-4}, and early stopping once the loss does not decrease by at least 1% over 10 gradient steps.

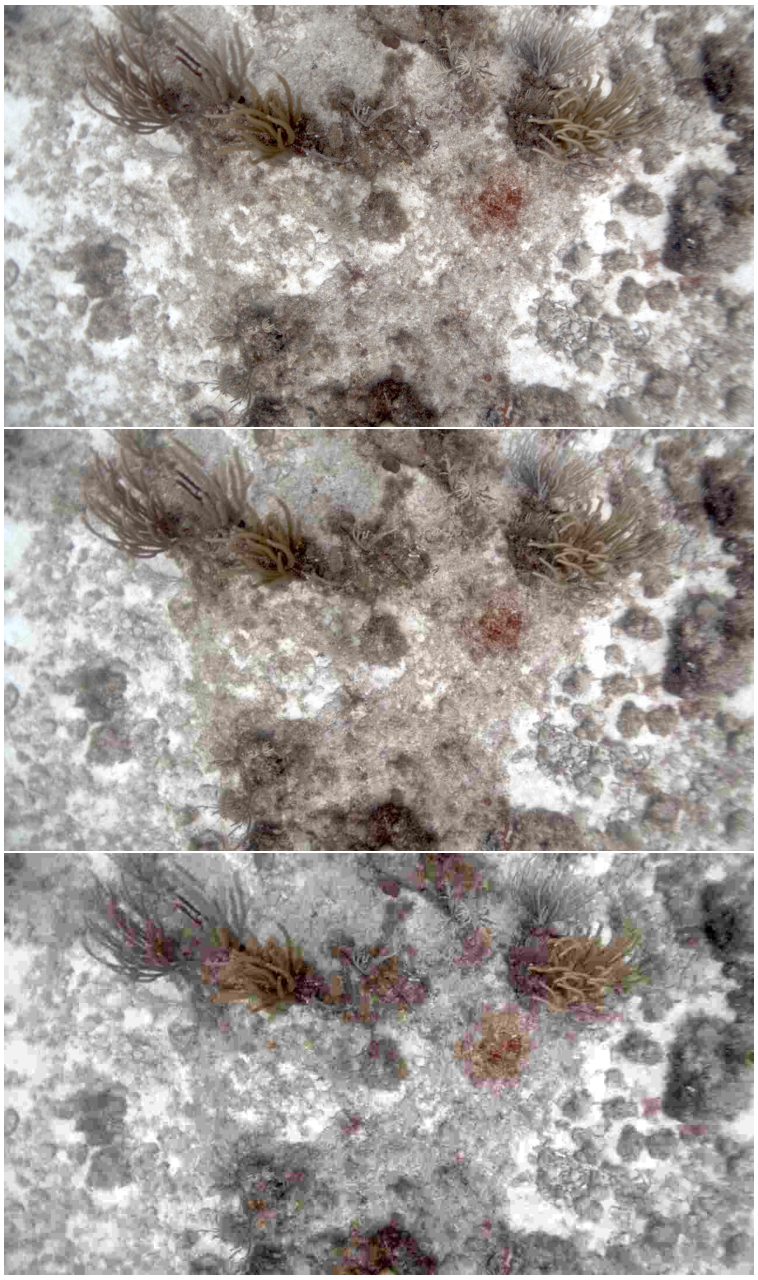


Figure B.2: The same image, in its original quality (8MP) with mild JPEG compression (2MB file size; top), reduced 8x to 1280x720 pixels (1MP) and with moderate JPEG compression (55 kB file size; middle), and at the same 1280x720 resolution with extreme JPEG compression (32.5 kB file size; bottom). The distortions in the final image significantly alter the contents of the image, and would make it very difficult to distinguish any anomalies or targets of interest. Accordingly, we treat 50 kB as the minimum file size for transmitting an image query. The top image was collected by the CUREE AUV [166] at Tektite reef.

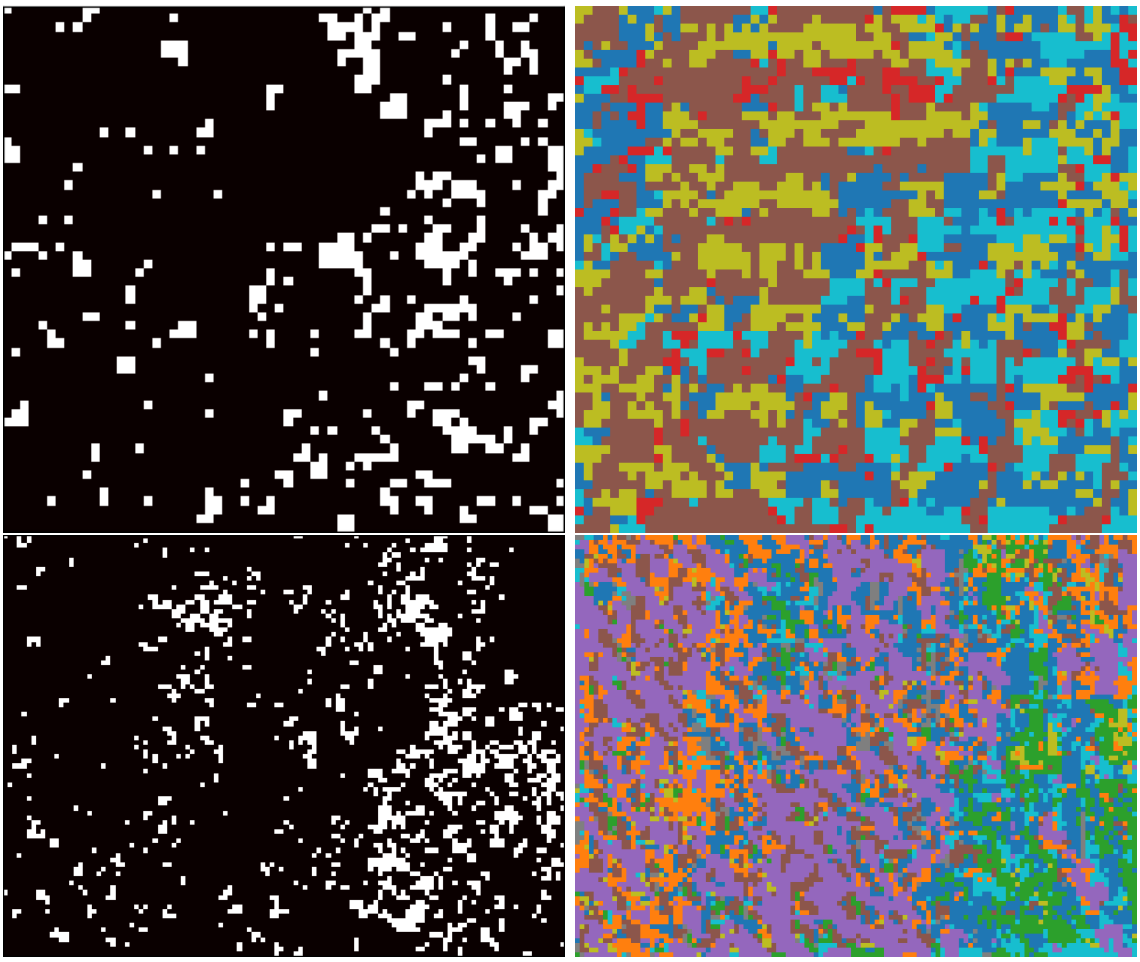


Figure B.3: These maps show locations in each site where a significant number of sea fans were located. Each pixel corresponds to a 0.5 x 0.5 m patch of the benthos. Top: Booby Rock Reef. Bottom: Tektite Reef.

Appendix C

Coral Reef Datasets & Limitations

Large-scale coral reef datasets are invaluable for training reef monitoring robots and related algorithms in simulation. Unfortunately, most coral reef datasets consist only of individual photos taken from various reefs [232]–[234]. These generally provide coarse geographic coordinates for locating the reef at which the images were taken, but do not localize individual photos. While such datasets are useful for training algorithms that identify and assess individual corals, they do not provide a way to understand larger scale reef structures. Some datasets additionally provide videos of linear transects through reefs [234], [235], which provide a limited way to understand larger reef structures; some of these are even timeseries datasets, imaging the same reef at different dates [236]. However, these are similarly not precisely geolocalized, which precludes making comparisons of the same reef structures at different times. On the other hand, remote sensing datasets from satellites are precisely localized, but have very low spatial resolution [237].

There are only a few high-resolution geolocalized coral reef datasets; for example, some have been released by the Australian Centre for Field Robotics (ACFR) [238], [239], and by the Sandin Lab at the University of San Diego [28], [240]. The ACFR datasets cover large regions with moderate spatial resolution, but are not labelled. The 100 Islands Challenge dataset from the Sandin lab images much smaller reefs, but with very high spatial resolution and semantic labels. However, neither dataset provides 3D models or 3D semantic annotations, or timeseries data of the same reefs on different dates.

References

[1] J. M. McDermott, R. Parnell-Turner, T. Barreyre, *et al.*, "Discovery of active off-axis hydrothermal vents at 9° 54′n east pacific rise," *Proceedings of the National Academy of Sciences*, vol. 119, no. 30, e2205602119, Jul. 26, 2022, Publisher: Proceedings of the National Academy of Sciences. DOI: 10.1073/pnas.2205602119.

[2] M. B. Lyons, N. J. Murray, E. V. Kennedy, *et al.*, "New global area estimates for coral reefs from high-resolution mapping," *Cell Reports Sustainability*, vol. 1, no. 2, p. 100 015, Feb. 23, 2024. DOI: 10.1016/j.crsus.2024.100015.

[3] R. D. Carlton, A. C. Dempsey, K. Lubarsky, M. Faisal, and S. Purkis, "Chagos archipelago final report," Khaled bin Sultan Living Oceans Foundation, Vol 13, 2021.

[4] G. Flaspohler, V. Preston, A. P. M. Michel, Y. Girdhar, and N. Roy, "Information-guided robotic maximum seek-and-sample in partially observable continuous environments," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3782–3789, Oct. 2019. DOI: 10/ggb48n.

[5] J. Hwang, N. Bose, and S. Fan, "AUV adaptive sampling methods: A review," *Applied Sciences*, vol. 9, no. 15, p. 3145, Jan. 2019, Number: 15 Publisher: Multidisciplinary Digital Publishing Institute. DOI: 10.3390/app9153145.

[6] J. P. Grotzinger, J. Crisp, A. R. Vasavada, *et al.*, "Mars science laboratory mission and science investigation," *Space Science Reviews*, vol. 170, no. 1, pp. 5–56, Sep. 1, 2012. DOI: 10/f39s2w.

[7] Y. Girdhar, "Unsupervised semantic perception, summarization, and autonomous exploration for robots in unstructured environments," Ph.D. dissertation, 2014, 139 pp.

[8] Y. Girdhar, Walter Cho, M. Campbell, J. Pineda, E. Clarke, and H. Singh, "Anomaly detection in unstructured environments using bayesian nonparametric scene modeling," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 2651–2656. DOI: 10/ggb48m.

[9] Y. Girdhar and G. Dudek, "Modeling curiosity in a mobile robot for long-term autonomous exploration and monitoring," *Autonomous Robots*, vol. 40, no. 7, pp. 1267–1278, 2016, Publisher: Springer US. DOI: 10/744. arXiv: 1509.07975.

[10] G. Flaspohler, N. Roy, and Y. Girdhar, "Feature discovery and visualization of robot mission data using convolutional autoencoders and bayesian nonparametric topic models," in *IEEE International Conference on Intelligent Robots and Systems*, ISSN: 21530866, 2017, pp. 1–8. DOI: 10/ggb47x. arXiv: 1712.00028.

[11] Y. Girdhar, L. Cai, S. Jamieson, N. McGuire, G. Flaspohler, S. Suman, and B. Claus, "Streaming scene maps for co-robotic exploration in bandwidth limited environments," in *2019 International Conference on Robotics and Automation (ICRA)*, tex.ids= Girdhar2019 ISSN: 2577-087X, Montréal, Canada: IEEE, May 30, 2019, pp. 7940–7946. DOI: 10/ggb46q.

[12] J. E. San Soucie, H. M. Sosik, and Y. Girdhar, "Gaussian-dirichlet random fields for inference over high dimensional categorical observations," in *2020 International Conference on Robotics and Automation*, Paris, France: IEEE, May 2020. arXiv: 2003.12120.

[13] S. Jamieson, J. P. How, and Y. Girdhar, "Active reward learning for co-robotic vision based exploration in bandwidth limited environments," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, ISSN: 2577-087X, Paris, France: IEEE, May 30, 2020, pp. 1806–1812. DOI: 10/gjktt7.

[14] X. Wang, X. Fan, P. Shi, J. Ni, and Z. Zhou, "An overview of key SLAM technologies for underwater scenes," *Remote Sensing*, vol. 15, no. 10, p. 2496, Jan. 2023, Number: 10 Publisher: Multidisciplinary Digital Publishing Institute. DOI: 10.3390/rs15102496.

[15] N. Roy and A. McCallum, "Toward optimal active learning through sampling estimation of error reduction," *Proceedings of 18th International Conference on Machine Learning, ICML*, pp. 441–448, 2001.

[16] F. Doshi-Velez, J. Pineau, and N. Roy, "Reinforcement learning with limited reinforcement: Using bayes risk for active learning in POMDPs," *Artificial Intelligence*, vol. 187-188, pp. 115–132, 2012, Publisher: Elsevier B.V. DOI: 10/f997t2.

[17] I. O. Ryzhov, W. B. Powell, and P. I. Frazier, "The knowledge gradient algorithm for a general class of online learning problems," *Operations Research*, vol. 60, no. 1, pp. 180–195, Feb. 2012. DOI: 10/gnxtkp.

[18] I. O. Ryzhov, P. I. Frazier, and W. B. Powell, "On the robustness of a one-period look-ahead policy in multi-armed bandit problems," *Procedia Computer Science*, ICCS 2010, vol. 1, no. 1, pp. 1635–1644, May 1, 2010. DOI: 10/cc8zvx.

[19] D. Russo and B. Van Roy, "Learning to optimize via information-directed sampling," *Operations Research*, vol. 66, no. 1, pp. 230–252, Feb. 2018. DOI: 10/gc6t2n.

[20] T. Dam, G. Chalvatzaki, J. Peters, and J. Pajarinen, "Monte-carlo robot path planning," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11 213–11 220, Oct. 2022. DOI: 10.1109/LRA.2022.3199674.

[21] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, Jun. 1, 2011, Publisher: SAGE Publications Ltd STM. DOI: 10.1177/0278364911406761.

[22] J. D. Gammell, S. S. Srinivasa, and T. D. Barfoot, "Informed RRT*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic," *IEEE International Conference on Intelligent Robots and Systems*, pp. 2997–3004, 2014, ISBN: 9781479969340. DOI: 10/gctq9p. arXiv: 1404.2334v3.

[23] S. Jamieson, K. Fathian, K. Khosoussi, J. P. How, and Y. Girdhar, "Multi-robot distributed semantic mapping in unfamiliar environments through online matching of learned representations," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, ISSN: 2577-087X, Xi'an, China: IEEE, May 30, 2021, pp. 8587–8593. DOI: 10.1109/ICRA48506.2021.9561934.

[24] K. Doherty, G. Flaspohler, N. Roy, and Y. Girdhar, "Approximate distributed spatiotemporal topic models for multi-robot terrain characterization," in *Intelligent Robots and Systems (IROS)*, 2018. DOI: 10/ggb47v.

[25] S. Jamieson, J. P. How, and Y. Girdhar, "DeepSeeColor: Realtime adaptive color correction for autonomous underwater vehicles via deep learning methods," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, London, UK: IEEE, May 27, 2023, pp. 3095–3101. DOI: 10.1109/ICRA48891.2023.10160477. arXiv: 2303.04025[cs].

[26] K. Fathian, K. Khosoussi, Y. Tian, P. Lusk, and J. P. How, "CLEAR: A consistent lifting, embedding, and alignment rectification algorithm for multi-view data association," *arXiv:1902.02256 [cs]*, Jul. 31, 2019. arXiv: 1902.02256.

[27] S. Jamieson, J. P. How, and Y. Girdhar, "Finding the optimal exploration-exploitation trade-off online through bayesian risk estimation and minimization," *Artificial Intelligence*, p. 104 096, Feb. 21, 2024. DOI: 10.1016/j.artint.2024.104096.

[28] J. E. Smith, R. Brainard, A. Carter, *et al.*, "Re-evaluating the health of coral reef communities: Baselines and evidence for human impacts across the central pacific," *Proceedings of the Royal Society B: Biological Sciences*, vol. 283, no. 1822, 2016, ISBN: 0962-8452. DOI: 10/ggb48x.

[29] O. Hoegh-Guldberg, E. S. Poloczanska, W. Skirving, and S. Dove, "Coral reef ecosystems under climate change and ocean acidification," *Frontiers in Marine Science*, vol. 4, 2017. DOI: 10.3389/fmars.2017.00158.

[30] R. A. Magris, A. Grech, and R. L. Pressey, "Cumulative human impacts on coral reefs: Assessing risk and management implications for brazilian coral reefs," *Diversity*, vol. 10, no. 2, p. 26, Jun. 2018, Number: 2 Publisher: Multidisciplinary Digital Publishing Institute. DOI: 10.3390/d10020026.

[31] H. A. El-Naggar, "Human impacts on coral reef ecosystem," in *Natural Resources Management and Biological Sciences*, IntechOpen, Oct. 6, 2020. DOI: 10.5772/intechopen.88841.

[32] National Oceanic and Atmospheric Administration and Harbor Branch Oceanographic Institution. "Medicines from the deep sea: Exploration of the gulf of mexico," NOAA Ocean Exploration. (2003), Available: https://oceanexplorer.noaa.gov/explorations/03bio/welcome.html.

[33] A. E. Wright, National Oceanic and Atmospheric Administration, and Harbor Branch Oceanographic Institution. "Biological diversity equals chemical diversity-the search for better medicines," NOAA Ocean Exploration. (2002), Available: https://oceanexplorer.noaa.gov/explorations/02sab/background/biodiversity/biodiversity.html.

[34] K. McPhail. "Searching for new pharmaceutical drugs from hydrothermal vent animals and microbes," NOAA Ocean Exploration. (2012), Available: https://oceanexplorer.noaa.gov/explorations/12fire/background/pharmacology/pharmacology.html.

[35] R. Sharma, "Deep-sea mining: Economic, technical, technological, and environmental considerations for sustainable development," *Marine Technology Society Journal*, vol. 45, no. 5, pp. 28–41, Sep. 1, 2011. DOI: 10.4031/MTSJ.45.5.2.

[36] L.-K. L. Trellevik, "Exploring exploration—how to look for deep-sea minerals," *Mineral Economics*, May 16, 2023. DOI: 10.1007/s13563-023-00379-x.

[37] A. Brady. "Today's deep-sea explorers are mineral miners and ultrawealthy hobbyists," Scientific American. (Jul. 1, 2023), Available: https://www.scientificamerican.com/article/todays-deep-sea-explorers-are-mineral-miners-and-ultrawealthy-hobbyists/.

[38] M. J. Sonter, "The technical and economic feasibility of mining the near-earth asteroids," *Acta Astronautica*, Developing Business, vol. 41, no. 4, pp. 637–647, Aug. 1, 1997. DOI: 10.1016/S0094-5765(98)00087-3.

[39] R. S. Jakhu, J. N. Pelton, and Y. O. M. Nyampong, *Space Mining and Its Regulation*. Cham: Springer International Publishing, 2017. DOI: 10.1007/978-3-319-39246-2.

[40] A. Khairutdinov, Y. Tyulyaeva, C. Kongar-Syuryun, and A. Rybak, "Extraction of minerals on celestial bodies as a new scientific direction," *IOP Conference Series: Earth and Environmental Science*, vol. 684, no. 1, p. 012 004, Mar. 2021, Publisher: IOP Publishing. DOI: 10.1088/1755-1315/684/1/012004.

[41] S. Pare. "Scientists discover ancient, underwater volcano is still active — and covered in up to a million giant eggs," Live Science. (Jul. 18, 2023), Available: https://www.livescience.com/planet-earth/volcanos/scientists-discover-ancient-underwater-volcano-is-still-active-and-covered-in-up-to-a-million-giant-eggs.

[42] Charles Darwin Foundation. "Scientists discover healthy deep-sea coral reefs and new seamounts in galapagos," Charles Darwin Foundation. (Oct. 26, 2023), Available: https://www.darwinfoundation.org/en/news/all-news-stories/scientists-discover-healthy-deep-sea-coral-reefs-and-new-seamounts-in-the-galapagos/.

[43] L. Mock-Bunting. "Scientists discover new ecosystem underneath hydrothermal vents," Schmidt Ocean Institute. (Aug. 8, 2023), Available: https://schmidtocean.org/scientists-discover-new-ecosystem-underneath-hydrothermal-vents/.

[44] E. Galili, M. Weinstein-Evron, I. Hershkovitz, A. Gopher, M. Kislev, O. Lernau, L. Kolska-Horwitz, and H. Lernau, "Atlit-yam: A prehistoric site on the sea floor off the israeli coast," *Journal of Field Archaeology*, vol. 20, no. 2, pp. 133–157, 1993, Publisher: [Maney Publishing, Trustees of Boston University]. DOI: 10.2307/529950.

[45] G. Papatheodorou, M. Geraga, A. Chalari, D. Christodoulou, M. Iatrou, E. Fakiris, S. Kordella, M. Prevenios, and G. Ferentinos, "Remote sensing for underwater archaeology: Case studies from greece and eastern mediterranean," *Bulletin of the Geological Society of Greece*, vol. 44, pp. 100–115, Feb. 1, 2011. DOI: 10.12681/bgsg.11440.

[46] R. Pacheco-Ruiz, J. Adams, F. Pedrotti, M. Grant, J. Holmlund, and C. Bailey, "Deep sea archaeological survey in the black sea – robotic documentation of 2,500 years of human seafaring," *Deep Sea Research Part I: Oceanographic Research Papers*, vol. 152, p. 103 087, Oct. 1, 2019. DOI: 10.1016/j.dsr.2019.103087.

[47] J. Michel and R. Ballard, "The RMS titanic 1985 discovery expedition," in *Proceedings of OCEANS'94*, vol. 3, Sep. 1994, III/132–III/137 vol.3. DOI: 10.1109/OCEANS.1994.364185.

[48] E. Visontay, "Accidental ocean floor discovery solves 120-year-old mystery of coal ship disappearance," *The Guardian*, Feb. 25, 2024.

[49] J. Pinto, M. Costa, K. Lima, *et al.*, "To boldly dive where no one has gone before: Experiments in coordinated robotic ocean exploration," in *Experimental Robotics*, B. Siciliano, C. Laschi, and O. Khatib, Eds., ser. Springer Proceedings in Advanced Robotics, Cham: Springer International Publishing, 2021, pp. 472–487. DOI: 10.1007/978-3-030-71151-1_42.

[50] M. Faria, J. Pinto, F. Py, J. Fortuna, H. Dias, R. Martins, F. Leira, T. A. Johansen, J. Sousa, and K. Rajan, "Coordinating UAVs and AUVs for oceanographic field experiments: Challenges and lessons learned," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, ISSN: 1050-4729, May 2014, pp. 6606–6611. DOI: 10.1109/ICRA.2014.6907834.

[51] C. Whitt, J. Pearlman, B. Polagye, *et al.*, "Future vision for autonomous ocean observations," *Frontiers in Marine Science*, vol. 7, Sep. 8, 2020, Publisher: Frontiers. DOI: 10.3389/fmars.2020.00697.

[52] D. F. Wolf and G. S. Sukhatme, "Semantic mapping using mobile robots," *IEEE Transactions on Robotics*, vol. 24, no. 2, pp. 245–258, Apr. 2008, Conference Name: IEEE Transactions on Robotics. DOI: 10/b8np55.

[53] S. L. Bowman, N. Atanasov, K. Daniilidis, and G. J. Pappas, "Probabilistic data association for semantic SLAM," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, tex.ids: Bowman2017a, Singapore, Singapore: IEEE, May 2017, pp. 1722–1729. DOI: 10/gf349x.

[54] K. Himri, P. Ridao, N. Gracias, A. Palomer, N. Palomeras, and R. Pi, "Semantic SLAM for an AUV using object recognition from point clouds," *IFAC-PapersOnLine*, vol. 51, no. 29, pp. 360–365, 2018, tex.ids: 2018 publisher: Elsevier. DOI: 10/ghf8wn.

[55] J. Zhang, M. Gui, Q. Wang, R. Liu, J. Xu, and S. Chen, "Hierarchical topic model based object association for semantic SLAM," *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 11, pp. 3052–3062, Nov. 2019. DOI: 10/ggd9zk.

[56] M. Everett, J. Miller, and J. P. How, "Planning beyond the sensing horizon using a learned context," *arXiv:1908.09171 [cs]*, Oct. 26, 2019. arXiv: 1908.09171.

[57] N. Sunderhauf, T. T. Pham, Y. Latif, M. Milford, and I. Reid, "Meaningful maps with object-oriented semantic mapping," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Citation Key Alias: Sunderhauf, Vancouver, BC: IEEE, Sep. 2017, pp. 5079–5085. DOI: 10/ggb474.

[58] Q.-H. Pham, B.-S. Hua, D. T. Nguyen, and S.-K. Yeung, "Real-time progressive 3d semantic segmentation for indoor scene," *arXiv:1804.00257 [cs]*, Apr. 5, 2019.

[59] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, "Kimera: An open-source library for real-time metric-semantic localization and mapping," *arXiv:1910.02490 [cs]*, Oct. 6, 2019. arXiv: 1910.02490.

[60] Y. Nakajima and H. Saito, "Efficient object-oriented semantic mapping with object detector," *IEEE Access*, vol. 7, pp. 3206–3213, 2019. DOI: 10/ggd47n.

[61] M. Grinvald, F. Furrer, T. Novkovic, J. J. Chung, C. Cadena, R. Siegwart, and J. Nieto, "Volumetric instance-aware semantic mapping and 3d object discovery," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 3037–3044, Jul. 2019. DOI: 10/ggdzm5.

[62] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003, ISBN: 9781577352815. DOI: 10/fc8s6g. arXiv: 1111.6189v1.

[63] L. Cao and L. Fei-Fei, "Spatially coherent latent topic model for concurrent segmentation and classification of objects and scenes," *Proceedings of the IEEE International Conference on Computer Vision*, 2007, ISBN: 9781424416318. DOI: 10/cmp782.

[64] X. Wang and E. Grimson, "Spatial latent dirichlet allocation," in *Neural Information Processing Systems*, 2007, pp. 1–8.

[65] Y. Girdhar, P. Giguère, and G. Dudek, "Autonomous adaptive exploration using realtime online spatiotemporal topic modeling," *The International Journal of Robotics Research*, vol. 33, no. 4, pp. 645–657, Apr. 13, 2014. DOI: 10/f539cj.

[66] Y. Girdhar and G. Dudek, "Gibbs sampling strategies for semantic perception of streaming video data," *arXiv:1509.03242 [cs]*, Sep. 10, 2015. arXiv: 1509.03242.

[67] M. Thoma, "A survey of semantic segmentation," pp. 1–16, 2016. DOI: 10/d665fp. arXiv: 1602.06541.

[68] X. Xia and B. Kulis, "W-net: A deep model for fully unsupervised image segmentation," 2017. arXiv: 1711.08506.

[69] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, "ROS: An open-source robot operating system," presented at the ICRA workshop on open source software, vol. 3, 2009, p. 6.

[70] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, no. 3, pp. 285–294, 1933, Publisher: [Oxford University Press, Biometrika Trust]. DOI: 10/dpnhzg.

[71] S. S. Villar, J. Bowden, and J. Wason, "Multi-armed bandit models for the optimal design of clinical trials: Benefits and challenges," *Statistical science : a review journal of the Institute of Mathematical Statistics*, vol. 30, no. 2, pp. 199–215, 2015. DOI: 10/f7gms6.

[72] T. Lattimore and C. Szepesvári, *Bandit Algorithms*, 1st ed. Cambridge University Press, Jul. 31, 2020. DOI: 10.1017/9781108571401.

[73] T. Lattimore and C. Szepesvari, "An information-theoretic approach to minimax regret in partial monitoring," in *Proceedings of Machine Learning Research*, A. Beygelzimer and D. Hsu, Eds., vol. 99, May 29, 2019, pp. 2111–2139. arXiv: 1902.00470.

[74] G. Bartók, D. Pál, and C. Szepesvári, "Minimax regret of finite partial-monitoring games in stochastic environments," in *Proceedings of the 24th Annual Conference on Learning Theory*, ISSN: 1938-7228, JMLR Workshop and Conference Proceedings, Dec. 21, 2011, pp. 133–154.

[75] G. Bartók, D. P. Foster, D. Pál, A. Rakhlin, and C. Szepesvári, "Partial monitoring—classification, regret bounds, and algorithms," *Mathematics of Operations Research*, vol. 39, no. 4, pp. 967–997, Nov. 2014. DOI: 10.1287/moor.2014.0663.

[76] R. Kleinberg and T. Leighton, "The value of knowing a demand curve: Bounds on regret for online posted-price auctions," in *44th Annual IEEE Symposium on Foundations of Computer Science, 2003. Proceedings.*, ISSN: 0272-5428, Cambridge, MA, USA: IEEE Computer Society, Oct. 1, 2003, pp. 594–605. DOI: 10.1109/SFCS.2003.1238232.

[77] G. Bartok, N. Zolghadr, and C. Szepesvari, "An adaptive algorithm for finite stochastic partial monitoring," in *Proceedings of the 29th International Conference on Machine Learning*, Edinburgh, Scotland, UK, Jun. 2012, pp. 1–20.

[78] M. Aziz, E. Kaufmann, and M.-K. Riviere, "On multi-armed bandit designs for dose-finding clinical trials," *Journal of Machine Learning Research*, vol. 22, no. 14, p. 38, 2021. arXiv: 1903.07082.

[79] R. I. Brafman and M. Tennenholtz, "R-max – a general polynomial time algorithm for near-optimal reinforcement learning," *Journal of Machine Learning Research*, vol. 3, pp. 213–231, Oct. 2002.

[80] T. Jaksch, R. Ortner, and P. Auer, "Near-optimal regret bounds for reinforcement learning," *Journal of Machine Learning Research*, vol. 11, S. Kakade, Ed., p. 38, Apr. 2010.

[81] P. Auer, N. Cesa-Bianchi, and P. Fisher, "Finite-time analysis of the multiarmed bandit problem," *Machine Learning*, vol. 47, J. Kivinen, Ed., pp. 235–256, 2002, Publisher: Kluwer Academic Publishers.

[82] E. Kaufmann, O. Cappe, and A. Garivier, "On bayesian upper confidence bounds for bandit problems," in *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, N. D. Lawrence and M. Girolami, Eds., ser. Proceedings of Machine Learning Research, vol. 22, La Palma, Canary Islands: PMLR, Apr. 21, 2012, pp. 592–600.

[83] K. Jamieson, M. Malloy, R. Nowak, and S. Bubeck, "Lil' UCB: An optimal exploration algorithm for multi-armed bandits," *Journal of Machine Learning Research*, vol. 35, no. 1964, pp. 423–439, 2014.

[84] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *Journal of Global Optimization*, vol. 13, no. 4, pp. 455–492, Dec. 1, 1998. DOI: 10/fg68nc.

[85] C. Qin, D. Klabjan, and D. Russo, "Improving the expected improvement algorithm," in *31st Conference on Neural Information Processing Systems*, Long Beach, CA, USA, 2017, p. 11.

[86] Y. Wang and W. B. Powell, "Finite-time analysis for the knowledge-gradient policy," *SIAM Journal on Control and Optimization*, vol. 56, no. 2, pp. 1105–1129, Jan. 2018. DOI: 10.1137/16M1073388. arXiv: 1606.04624.

[87] D. Russo, "Simple bayesian algorithms for best-arm identification," *Operations Research*, vol. 68, no. 6, pp. 1625–1647, Nov. 2020, tex.ids= Russo2018 publisher: INFORMS. DOI: 10.1287/opre.2019.1911. arXiv: 1602.08448.

[88] S. S. Gupta and K. J. Miescke, "Bayesian look ahead one-stage sampling allocations for selection of the best population," *Journal of Statistical Planning and Inference*, 40 Years of Statistical Selection Theory, Part I, vol. 54, no. 2, pp. 229–244, Sep. 16, 1996. DOI: 10.1016/0378-3758(95)00169-7.

[89] P. I. Frazier, W. B. Powell, and S. Dayanik, "A knowledge-gradient policy for sequential information collection," *SIAM Journal on Control and Optimization*, vol. 47, no. 5, pp. 2410–2439, Sep. 1, 2008. DOI: 10.1137/070693424.

[90] R. Bellman, "A markovian decision process," *Journal of Mathematics and Mechanics*, vol. 6, no. 5, pp. 679–684, 1957, Publisher: Indiana University Mathematics Department.

[91] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial Intelligence*, vol. 101, no. 1, pp. 99–134, May 1, 1998. DOI: 10.1016/S0004-3702(98)00023-X.

[92] C. Cai, X. Liao, and L. Carin, "Learning to explore and exploit in POMDPs," in *Advances in Neural Information Processing Systems*, vol. 22, Curran Associates, Inc., 2009.

[93] A. Sharma, J. Harrison, M. Tsao, and M. Pavone, "Robust and adaptive planning under model uncertainty," in *Proceedings of the Twenty-Ninth International Conference on Automated Planning and Scheduling*, Association for the Advancement of Artificial Intelligence, Jan. 8, 2019. DOI: 10.1609/icaps.v29i1.3505. arXiv: 1901.02577[cs].

[94] M. Ghavamzadeh, S. Mannor, J. Pineau, and A. Tamar, "Bayesian reinforcement learning: A survey," *Foundations and Trends in Machine Learning*, vol. 8, no. 5, pp. 359–483, 2015. DOI: 10.1561/2200000049. arXiv: 1609.04436.

[95] Q. Liu, A. Chung, C. Szepesvari, and C. Jin, "When is partially observable reinforcement learning not scary?" In *Proceedings of Thirty Fifth Conference on Learning Theory*, ISSN: 2640-3498, PMLR, Jun. 28, 2022, pp. 5175–5220.

[96] G. Arcieri, C. Hoelzl, O. Schwery, D. Straub, K. G. Papakonstantinou, and E. Chatzi, "Bridging POMDPs and bayesian decision making for robust maintenance planning under model uncertainty: An application to railway systems," *Reliability Engineering & System Safety*, vol. 239, p. 109 496, Nov. 1, 2023. DOI: 10.1016/j.ress.2023.109496.

[97] P. Sharma, B. Kraske, J. Kim, Z. Laouar, Z. Sunberg, and E. Atkins, "Risk-aware markov decision process contingency management autonomy for uncrewed aircraft systems," *Journal of Aerospace Information Systems*, vol. 0, no. 0, pp. 1–15, Jan. 9, 2024, Publisher: American Institute of Aeronautics and Astronautics _eprint: https://doi.org/10.2514/1.I011235. DOI: 10.2514/1.I011235.

[98] R. Meshram, A. Gopalan, and D. Manjunath, "Optimal recommendation to users that react: Online learning for a class of POMDPs," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, Dec. 2016, pp. 7210–7215. DOI: 10.1109/CDC.2016.7799381.

[99] M. Komorowski, L. A. Celi, O. Badawi, A. C. Gordon, and A. A. Faisal, "The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care," *Nature Medicine*, vol. 24, no. 11, pp. 1716–1720, Nov. 2018, Number: 11 Publisher: Nature Publishing Group. DOI: 10.1038/s41591-018-0213-5.

[100] M. O. Duff and A. Barto, "Optimal learning: Computational procedures for bayes-adaptive markov decision processes," Ph.D. dissertation, University of Massachusetts Amherst, 2002.

[101] J. Garcıa and F. Fernández, "A comprehensive survey on safe reinforcement learning," *Journal of Machine Learning Research*, vol. 16, no. 1, pp. 1437–1480, 2015.

[102] Y. Shen, M. J. Tobia, T. Sommer, and K. Obermayer, "Risk-sensitive reinforcement learning," *Neural Computation*, vol. 26, no. 7, pp. 1298–1328, Jul. 1, 2014. DOI: 10.1162/NECO_a_00600.

[103] M. Rigter, B. Lacerda, and N. Hawes, "Risk-averse bayes-adaptive reinforcement learning," in *Advances in Neural Information Processing Systems*, vol. 34, Curran Associates, Inc., 2021, pp. 1142–1154.

[104] S. Al-Hussaini, N. Dhanaraj, J. M. Gregory, R. Jomy Joseph, S. Thakar, B. C. Shah, J. A. Marvel, and S. K. Gupta, "Seeking human help to manage plan failure risks in semi-autonomous mobile manipulation," *Journal of Computing and Information Science in Engineering*, vol. 22, no. 50906, Apr. 13, 2022. DOI: 10.1115/1.4054088.

[105] B. Charpentier, R. Senanayake, M. Kochenderfer, and S. Günnemann, *Disentangling epistemic and aleatoric uncertainty in reinforcement learning*, Jun. 3, 2022. DOI: 10.48550/arXiv.2206.01558. arXiv: 2206.01558[cs].

[106] P. Festor, G. Luise, M. Komorowski, and A. A. Faisal, *Enabling risk-aware reinforcement learning for medical interventions through uncertainty decomposition*, Apr. 27, 2022. DOI: 10.48550/arXiv.2109.07827. arXiv: 2109.07827[cs].

[107] X. Lu, B. Van Roy, V. Dwaracherla, M. Ibrahimi, I. Osband, and Z. Wen, *Reinforcement learning, bit by bit*. Now Foundations and Trends, 2023.

[108] Y. Lin, Y. Ren, and E. Zhou, "Bayesian risk markov decision processes," *Advances in Neural Information Processing Systems*, vol. 35, pp. 17 430–17 442, Dec. 6, 2022.

[109] A. Guez, D. Silver, and P. Dayan, "Scalable and efficient bayes-adaptive reinforcement learning based on monte-carlo tree search," *Journal of Artificial Intelligence Research*, vol. 48, pp. 841–883, Nov. 30, 2013. DOI: 10.1613/jair.4117.

[110] G. Lee, B. Hou, A. Mandalika, J. Lee, S. Choudhury, and S. S. Srinivasa, *Bayesian policy optimization for model uncertainty*, May 8, 2019. DOI: 10.48550/arXiv.1810.01014. arXiv: 1810.01014[cs].

[111] H. Eriksson and C. Dimitrakakis, "Epistemic risk-sensitive reinforcement learning," in *Proceedings of the 28th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, Jun. 14, 2019, pp. 339–344. DOI: 10.48550/arXiv.1906.06273. arXiv: 1906.06273[cs,stat].

[112] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1, pp. 83–97, Mar. 1955. DOI: 10/b2k5tg.

[113] D. Pachauri, R. Kondor, and V. Singh, "Solving the multi-way matching problem by permutation synchronization," in *Advances in Neural Information Processing Systems*, 2013, pp. 1860–1868.

[114] Y. Chen, L. Guibas, and Q. Huang, "Near-optimal joint object matching via convex relaxation," in *Proceedings of the 31 st International Conference on Machine Learning*, vol. 32, Beijing, China: JMLR: W&CP, 2014, p. 9.

[115] N. Hu, Q. Huang, B. Thibert, and L. Guibas, "Distributable consistent multi-object matching," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA: IEEE, Jun. 2018, pp. 2463–2471. DOI: 10/ghf9vf.

[116] J.-G. Yu, G.-S. Xia, A. Samal, and J. Tian, "Globally consistent correspondence of multiple feature sets using proximal gauss–seidel relaxation," *Pattern Recognition*, vol. 51, pp. 255–267, Mar. 2016. DOI: 10/f758hj.

[117] X. Zhou, M. Zhu, and K. Daniilidis, "Multi-image matching via fast alternating minimization," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile: IEEE, Dec. 2015, pp. 4032–4040. DOI: 10/ghf9xt.

[118] E. Maset, F. Arrigoni, and A. Fusiello, "Practical and efficient multi-view matching," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice: IEEE, Oct. 2017, pp. 4578–4586. DOI: 10/ghf9vm.

[119] J. Yan, Z. Ren, H. Zha, and S. Chu, "A constrained clustering based approach for matching a collection of feature sets," presented at the IEEE International Conference on Pattern Recognition, Jun. 12, 2016, pp. 3832–3837. arXiv: 1606.03731.

[120] D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov Chains and Mixing Times, second edition*, 2nd. American Mathematical Society, 2017.

[121] N. Aletras and M. Stevenson, "Measuring the similarity between automatically generated topics," in *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2: Short Papers*, Gothenburg, Sweden: Association for Computational Linguistics, 2014, pp. 22–27. DOI: 10/ghfkpd.

[122] Epic Games, *Unreal engine*, version 4.24, Oct. 22, 2019.

[123] Kelint, *Automatic coral generator*, Publisher: Unreal Engine Marketplace, Apr. 12, 2016.

[124] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "AirSim: High-fidelity visual and physical simulation for autonomous vehicles," *arXiv:1705.05065 [cs]*, Jul. 18, 2017.

[125] A. Cully, K. Chatzilygeroudis, F. Allocati, and J.-B. Mouret, "Limbo: A flexible high-performance library for gaussian processes modeling and data-efficient optimization," *Journal of Open Source Software*, vol. 3, no. 26, p. 545, Jun. 26, 2018. DOI: 10/ggv6c4.

[126] N. X. Vinh, J. Epps, and J. Bailey, "Information theoretic measures for clusterings comparison: Is a correction for chance necessary?" In *Proceedings of the 26th International Conference on Machine Learning*, Montreal, Canada, 2009, p. 8. DOI: 10/fsxtxb.

[127] J. W. Kaeli, J. J. Leonard, and H. Singh, "Visual summaries for low-bandwidth semantic mapping with autonomous underwater vehicles," in *2014 IEEE/OES Autonomous Underwater Vehicles (AUV)*, IEEE, Oct. 2014, pp. 1–7. DOI: 10/ggb452.

[128] L. Cai, N. E. McGuire, R. Hanlon, T. A. Mooney, and Y. Girdhar, "Semi-supervised visual tracking of marine animals using autonomous underwater vehicles," *International Journal of Computer Vision*, Mar. 1, 2023. DOI: 10.1007/s11263-023-01762-5.

[129] D. R. Yoerger, A. F. Govindarajan, J. C. Howland, *et al.*, "A hybrid underwater robot for multidisciplinary investigation of the ocean twilight zone," *Science Robotics*, vol. 6, no. 55, Jun. 16, 2021, Publisher: American Association for the Advancement of Science. DOI: 10/gkqppx.

[130] K. Katija, P. L. D. Roberts, J. Daniels, A. Lapides, K. Barnard, M. Risi, B. Y. Ranaan, B. G. Woodward, and J. Takahashi, "Visual tracking of deepwater animals using machine learning-controlled robotic underwater vehicles," in *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Waikoloa, HI, USA: IEEE, Jan. 2021, pp. 859–868. DOI: 10.1109/WACV48630.2021.00090.

[131] Z. Hao, J. Qiu, H. Zhang, G. Ren, and C. Liu, "UMOTMA: Underwater multiple object tracking with memory aggregation," *Frontiers in Marine Science*, vol. 9, 2022. DOI: 10.3389/fmars.2022.1071618.

[132] E. Guerrero, F. Bonin-Font, and G. Oliver, "Adaptive visual information gathering for autonomous exploration of underwater environments," *IEEE Access*, vol. 9, pp. 136 487–136 506, 2021, Conference Name: IEEE Access. DOI: 10.1109/ACCESS.2021.3117343.

[133] F. Bonin-Font, G. Oliver, S. Wirth, M. Massot, P. Lluis Negre, and J.-P. Beltran, "Visual sensing for autonomous underwater exploration and intervention tasks," *Ocean Engineering*, vol. 93, pp. 25–44, Jan. 1, 2015. DOI: 10.1016/j.oceaneng.2014.11.005.

[134] B. Joshi, M. Xanthidis, M. Roznere, N. J. Burgdorfer, P. Mordohai, A. Q. Li, and I. Rekleitis, "Underwater exploration and mapping," in *2022 IEEE/OES Autonomous Underwater Vehicles Symposium (AUV)*, Singapore: IEEE, Sep. 19, 2022, pp. 1–7. DOI: 10.1109/AUV53081.2022.9965805.

[135] S. Manjanna, N. Kakodkar, M. Meghjani, and G. Dudek, "Efficient terrain driven coral coverage using gaussian processes for mosaic synthesis," in *2016 13th Conference on Computer and Robot Vision (CRV)*, Jun. 2016, pp. 448–455. DOI: 10.1109/CRV.2016.63.

[136] D. Rao, A. Bender, S. B. Williams, and O. Pizarro, "Multimodal information-theoretic measures for autonomous exploration," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 4230–4237. DOI: 10.1109/ICRA.2016.7487618.

[137] P. Singh, E. Gregson, J. Ross, M. Seto, C. Kaminski, and D. Hopkin, "Vision-based AUV docking to an underway dock using convolutional neural networks," in *2020 IEEE/OES Autonomous Underwater Vehicles Symposium (AUV)*, ISSN: 2377-6536, Sep. 2020, pp. 1–6. DOI: 10.1109/AUV50043.2020.9267926.

[138] T. Matsuda, T. Maki, K. Masuda, and T. Sakamaki, "Resident autonomous underwater vehicle: Underwater system for prolonged and continuous monitoring based at a seafloor station," *Robotics and Autonomous Systems*, vol. 120, p. 103 231, Oct. 1, 2019. DOI: 10.1016/j.robot.2019.07.001.

[139] A. M. Yazdani, K. Sammut, O. Yakimenko, and A. Lammas, "A survey of underwater docking guidance systems," *Robotics and Autonomous Systems*, vol. 124, p. 103 382, Feb. 1, 2020. DOI: 10.1016/j.robot.2019.103382.

[140] H.-M. Chou, Y.-C. Chou, and H.-H. Chen, "Development of a monocular vision deep learning-based AUV diver-following control system," in *Global Oceans 2020: Singapore – U.S. Gulf Coast*, ISSN: 0197-7385, Oct. 2020. DOI: 10.1109/IEEECONF38699.2020.9389477.

[141] N. Stilinović, D. Nad, and N. Mišković, "AUV for diver assistance and safety — design and implementation," in *OCEANS 2015 - Genova*, May 2015, pp. 1–4. DOI: 10.1109/OCEANS-Genova.2015.7271670.

[142] A. Gomez Chavez, A. Ranieri, D. Chiarella, E. Zereik, A. Babić, and A. Birk, "CADDY underwater stereo-vision dataset for human–robot interaction (HRI) in the context of diver activities," *Journal of Marine Science and Engineering*, vol. 7, no. 1, p. 16, Jan. 2019, Number: 1 Publisher: Multidisciplinary Digital Publishing Institute. DOI: 10.3390/jmse7010016.

[143] J. R. Pelletier, B. W. O'Neill, J. J. Leonard, L. Freitag, and E. Gallimore, "AUV-assisted diver navigation," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10 208–10 215, Oct. 2022, Conference Name: IEEE Robotics and Automation Letters. DOI: 10.1109/LRA.2022.3191164.

[144] A. Birk, "A survey of underwater human-robot interaction (u-HRI)," *Current Robotics Reports*, vol. 3, no. 4, pp. 199–211, Dec. 1, 2022. DOI: 10.1007/s43154-022-00092-7.

[145] Y. Wang, W. Song, G. Fortino, L.-Z. Qi, W. Zhang, and A. Liotta, "An experimental-based review of image enhancement and image restoration methods for underwater imaging," *IEEE Access*, vol. 7, pp. 140 233–140 251, 2019. DOI: 10.1109/ACCESS.2019.2932130.

[146] Q.-Y. Zhou and V. Koltun, "Color map optimization for 3d reconstruction with consumer depth cameras," *ACM Transactions on Graphics*, vol. 33, no. 4, 155:1–155:10, Jul. 27, 2014. DOI: 10.1145/2601097.2601134.

[147] K. De and M. Pedersen, "Impact of colour on robustness of deep neural networks," in *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, Montreal, BC, Canada: IEEE, Oct. 2021, pp. 21–30. DOI: 10.1109/ICCVW54120.2021.00009.

[148] N. Carlevaris-Bianco, A. Mohan, and R. M. Eustice, "Initial results in underwater single image dehazing," in *OCEANS 2010 MTS/IEEE SEATTLE*, ISSN: 0197-7385, Sep. 2010, pp. 1–8. DOI: 10.1109/OCEANS.2010.5664428.

[149] H.-Y. Yang, P.-Y. Chen, C.-C. Huang, Y.-Z. Zhuang, and Y.-H. Shiau, "Low complexity underwater image enhancement based on dark channel prior," in *2011 Second International Conference on Innovations in Bio-inspired Computing and Applications*, Dec. 2011, pp. 17–20. DOI: 10.1109/IBICA.2011.9.

[150] J. Y. Chiang and Ying-Ching Chen, "Underwater image enhancement by wavelength compensation and dehazing," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1756–1769, Apr. 2012. DOI: 10.1109/TIP.2011.2179666.

[151] D. Berman, T. Treibitz, and S. Avidan, "Diving into haze-lines: Color restoration of underwater images," in *Proceedings of the British Machine Vision Conference*, tex.ids= Berman, BMVA Press, 2017.

[152] M. Bryson, M. Johnson-Roberson, O. Pizarro, and S. B. Williams, "True color correction of autonomous underwater vehicle imagery," *Journal of Field Robotics*, vol. 33, no. 6, pp. 853–874, 2016, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rob.21638. DOI: 10.1002/rob.21638.

[153] D. Akkaynak and T. Treibitz, "Sea-thru: A method for removing water from underwater images," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA: IEEE, Jun. 2019, pp. 1682–1691. DOI: 10/ggvr3z.

[154] W. S. Pegau, D. Gray, and J. R. V. Zaneveld, "Absorption and attenuation of visible and near-infrared light in water: Dependence on temperature and salinity," *Applied Optics*, vol. 36, no. 24, Aug. 20, 1997. DOI: 10.1364/AO.36.006035.

[155] D. Akkaynak and T. Treibitz, "A revised underwater image formation model," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT: IEEE, Jun. 2018, pp. 6723–6732. DOI: 10.1109/CVPR.2018.00703.

[156] D. Berman, D. Levy, S. Avidan, and T. Treibitz, "Underwater single image color restoration using haze-lines and a new quantitative dataset," *arXiv:1811.01343 [cs]*, Mar. 24, 2019. arXiv: 1811.01343.

[157] H. Porav, W. Maddern, and P. Newman, "Adversarial training for adverse conditions: Robust metric localisation using appearance transfer," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, ISSN: 2577-087X, May 2018, pp. 1011–1018. DOI: 10.1109/ICRA.2018.8462894.

[158] J. Li, K. A. Skinner, R. M. Eustice, and M. Johnson-Roberson, "WaterGAN: Unsupervised generative network to enable real-time color correction of monocular underwater images," *IEEE Robotics and Automation Letters*, 2017. DOI: 10.1109/LRA.2017.2730363. arXiv: 1702.07392[cs].

[159] C. Fabbri, M. J. Islam, and J. Sattar, "Enhancing underwater imagery using generative adversarial networks," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, ISSN: 2577-087X, May 2018, pp. 7159–7165. DOI: 10.1109/ICRA.2018.8460552.

[160] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, Dec. 20, 2015, ISBN: 9780521835688. DOI: 10/bmqp. arXiv: 1312.6184v5.

[161] A. Reuther, P. Michaleas, M. Jones, V. Gadepally, S. Samsi, and J. Kepner, "Survey of machine learning accelerators," in *2020 IEEE High Performance Extreme Computing Conference (HPEC)*, ISSN: 2643-1971, Sep. 2020, pp. 1–12. DOI: 10.1109/HPEC43674.2020.9286149.

[162] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2341–2353, Dec. 2011, Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence. DOI: 10.1109/TPAMI.2010.168.

[163] G. Buchsbaum, "A spatial processor model for object colour perception," *Journal of the Franklin Institute*, vol. 310, no. 1, Jul. 1, 1980. DOI: 10.1016/0016-0032(80)90058-7.

[164] M. Ebner and J. Hansen, "Depth map color constancy," *Bio-Algorithms and Med-Systems*, vol. 9, no. 4, Jan. 1, 2013. DOI: 10.1515/bams-2013-0152.

[165] A. Paszke, S. Gross, F. Massa, *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Advances in neural information processing systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett, Eds., Curran Associates, Inc., 2019, pp. 8024–8035.

[166] Y. Girdhar, N. McGuire, L. Cai, S. Jamieson, S. McCammon, B. Claus, J. E. S. Soucie, J. E. Todd, and T. A. Mooney, "CUREE: A curious underwater robot for ecosystem exploration," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, London, UK, May 27, 2023, pp. 11 411–11 417. DOI: 10.1109/ICRA48891.2023.10161282.

[167] J. Kirschner, T. Lattimore, and A. Krause, "Information directed sampling for linear partial monitoring," *Proceedings of Machine Learning Research*, vol. 125, pp. 1–42, 2020.

[168] J. Kirschner, T. Lattimore, C. Vernade, and C. Szepesvari, "Asymptotically optimal information-directed sampling," *Proceedings of Machine Learning Research*, vol. 134, p. 45, 2021.

[169] J. Kirschner, T. Lattimore, and A. Krause, *Linear partial monitoring for sequential decision-making: Algorithms, regret bounds and applications*, Feb. 7, 2023. DOI: 10.48550/arXiv.2302.03683. arXiv: 2302.03683[cs,stat].

[170] S. Bubeck, R. Munos, and G. Stoltz, "Pure exploration in multi-armed bandits problems," in *Algorithmic Learning Theory*, R. Gavaldà, G. Lugosi, T. Zeugmann, and S. Zilles, Eds., ser. Lecture Notes in Computer Science, Berlin, Heidelberg: Springer, 2009, pp. 23–37. DOI: 10/b6456m.

[171] S. L. Scott, "A modern bayesian look at the multi-armed bandit," *Applied Stochastic Models in Business and Industry*, vol. 26, no. 6, pp. 639–658, Nov. 2010. DOI: 10/c8gqt8.

[172] K. Jamieson and R. Nowak, "Best-arm identification algorithms for multi-armed bandits in the fixed confidence setting," in *2014 48th Annual Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, USA: IEEE, Mar. 2014, pp. 1–6. DOI: 10.1109/CISS.2014.6814096.

[173] D. Russo and B. Van Roy, "Learning to optimize via posterior sampling," *Mathematics of Operations Research*, vol. 39, no. 4, pp. 1221–1243, 2014, Publisher: INFORMS. DOI: 10.1287/moor.2014.0650.

[174] X. Huo and F. Fu, "Risk-aware multi-armed bandit problem with application to portfolio selection," *Royal Society Open Science*, vol. 4, no. 11, p. 171 377, Nov. 2017, Publisher: Royal Society. DOI: 10.1098/rsos.171377.

[175] K. Misra, E. M. Schwartz, and J. Abernethy, "Dynamic online pricing with incomplete information using multiarmed bandit experiments," *Marketing Science*, vol. 38, no. 2, pp. 226–252, Mar. 1, 2019, tex.ids= Misra2019. DOI: 10/ghhc8j.

[176] D. Russo, "Technical note—a note on the equivalence of upper confidence bounds and gittins indices for patient agents," *Operations Research*, vol. 69, no. 1, pp. 273–278, Jan. 2021, Publisher: INFORMS. DOI: 10.1287/opre.2020.1987.

[177] C.-W. Hsu, B. Kveton, O. Meshi, M. Mladenov, and C. Szepesvari, "Empirical bayes regret minimization," *arXiv:1904.02664 [cs, stat]*, Jun. 10, 2020. arXiv: 1904.02664.

[178] M. Rigter, B. Lacerda, and N. Hawes, *One risk to rule them all: Addressing distributional shift in offline reinforcement learning via risk-aversion*, Jun. 2, 2023. DOI: 10.48550/arXiv.2212.00124. arXiv: 2212.00124[cs].

[179] S. Depeweg, J.-M. Hernandez-Lobato, F. Doshi-Velez, and S. Udluft, "Decomposition of uncertainty in bayesian deep learning for efficient and risk-sensitive learning," in *International conference on machine learning*, PMLR, 2018, pp. 1184–1193.

[180] D. Golovin and A. Krause, "Adaptive submodularity: Theory and applications in active learning and stochastic optimization," *Journal of Artificial Intelligence Research*, vol. 42, p. 60, Nov. 2011.

[181] K. Murota, *Discrete Convex Analysis* (Discrete Mathematics and Applications). Society for Industrial and Applied Mathematics, Jan. 2003, 406 pp. DOI: 10.1137/1.9780898718508.

[182] K. Murota, "6. m-convex functions," in *Discrete Convex Analysis*, ser. Discrete Mathematics and Applications, Society for Industrial and Applied Mathematics, Jan. 2003, pp. 133–176. DOI: 10.1137/1.9780898718508.ch6.

[183] K. Murota, "10. algorithms," in *Discrete Convex Analysis*, ser. Discrete Mathematics and Applications, Society for Industrial and Applied Mathematics, Jan. 2003, pp. 281–322. DOI: 10.1137/1.9780898718508.ch10.

[184] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions—i," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, Dec. 1, 1978. DOI: 10.1007/BF01588971.

[185] H. Kellerer, U. Pferschy, and D. Pisinger, "The bounded knapsack problem," in *Knapsack Problems*, H. Kellerer, U. Pferschy, and D. Pisinger, Eds., Berlin, Heidelberg: Springer, 2004, pp. 185–209. DOI: 10.1007/978-3-540-24777-7_7.

[186] R. Iyer and J. Bilmes, "Submodular optimization with submodular cover and submodular knapsack constraints," in *Proceedings of the 26th International Conference on Neural Information Processing Systems - volume 2*, ser. NIPS'13, Number of pages: 9 Place: Lake Tahoe, Nevada, Red Hook, NY, USA: Curran Associates Inc., 2013, pp. 2436–2444.

[187] W. Hoeffding, "On sequences of sums of independent random vectors," in *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 2: Contributions to Probability Theory*, vol. 2, Berkeley, CA: University of California Press, Jan. 1, 1961, pp. 213–227.

[188] L. L. Cam, *Asymptotic Methods in Statistical Decision Theory* (Springer Series in Statistics). New York, NY: Springer, 1986. DOI: 10.1007/978-1-4612-4946-7.

[189] I. Urteaga and C. H. Wiggins, "Bayesian bandits: Balancing the exploration-exploitation tradeoff via double sampling," *arXiv:1709.03162 [cs, stat]*, Aug. 8, 2018. arXiv: 1709.03162.

[190] P. Frazier, W. Powell, and S. Dayanik, "The knowledge-gradient policy for correlated normal beliefs," *INFORMS Journal on Computing*, vol. 21, no. 4, pp. 599–613, Nov. 2009, Publisher: INFORMS. DOI: 10.1287/ijoc.1080.0314.

[191] J.-Y. Audibert and S. Bubeck, "Regret bounds and minimax policies under partial monitoring," *Journal of Machine Learning Research*, vol. 11, no. 94, pp. 2785–2836, 2010.

[192] R. Degenne and V. Perchet, "Anytime optimal algorithms in stochastic multi-armed bandits," in *Proceedings of The 33rd International Conference on Machine Learning*, ISSN: 1938-7228, PMLR, Jun. 11, 2016, pp. 1587–1595.

[193] H. P. Vanchinathan, G. Bartók, and A. Krause, "Efficient partial monitoring with prior information," in *Advances in neural information processing systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, Eds., tex.ids= Vanchinathan, vol. 27, Curran Associates, Inc., 2014.

[194] A. Macwan, J. Vilela, G. Nejat, and B. Benhabib, "A multirobot path-planning strategy for autonomous wilderness search and rescue," *IEEE Transactions on Cybernetics*, vol. 45, no. 9, pp. 1784–1797, Sep. 2015, Conference Name: IEEE Transactions on Cybernetics. DOI: 10.1109/TCYB.2014.2360368.

[195] J. Scherer, S. Yahyanejad, S. Hayat, E. Yanmaz, T. Andre, A. Khan, V. Vukadinovic, C. Bettstetter, H. Hellwagner, and B. Rinner, "An autonomous multi-UAV system for search and rescue," in *Proceedings of the First Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use*, ser. DroNet '15, New York, NY, USA: Association for Computing Machinery, May 18, 2015, pp. 33–38. DOI: 10.1145/2750675.2750683.

[196] C. Sampedro, A. Rodriguez-Ramos, H. Bavle, A. Carrio, P. de la Puente, and P. Campoy, "A fully-autonomous aerial robot for search and rescue applications in indoor environments using learning-based techniques," *Journal of Intelligent & Robotic Systems*, vol. 95, no. 2, pp. 601–627, Aug. 2019. DOI: 10.1007/s10846-018-0898-1.

[197] E. Lobecker, A. D. Skarke, M. Nadeau, L. Brothers, B. Bingham, L. Stuart, J. Sheehan, and D. Paxton, "Mapping data acquisition and processing report : Cruise EX1205 leg 1, exploration blake plateau, july 5 - 24, 2012," Nov. 6, 2012, Publisher: NOAA Office of Ocean Exploration and Research. DOI: 10.7289/V5J38QJ0.

[198] A. D. Bowen, M. V. Jakuba, D. R. Yoerger, L. L. Whitcomb, J. C. Kinsey, L. Mayer, and C. R. German, "Nereid UI: A light-tethered remotely operated vehicle for under-ice telepresence," presented at the OTC Arctic Technology Conference, OnePetro, Dec. 3, 2012. DOI: 10.4043/23741-MS.

[199] T. M. Shank, C. Machado, C. R. German, A. Bowen, J. M. Leichty, A. T. Klesh, R. G. Smith, and K. P. Hand, "Development of a new class of autonomous underwater vehicle (AUV), orpheus, for the exploration of ocean world analogues," in *Ocean Worlds 4*, Conference Name: Ocean Worlds 4 ADS Bibcode: 2019LPICo2168.6021S, vol. 2168, May 21, 2019, p. 6021.

[200] V. Verma, M. W. Maimone, D. M. Gaines, *et al.*, "Autonomous robotics is driving perseverance rover's progress on mars," *Science Robotics*, vol. 8, no. 80, eadi3099, Jul. 26, 2023, Publisher: American Association for the Advancement of Science. DOI: 10.1126/scirobotics.adi3099.

[201] M. B. Bejiga, A. Zeggada, and F. Melgani, "Convolutional neural networks for near real-time object detection from UAV imagery in avalanche search and rescue operations," in *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, tex.ids= Bejiga2016a ISSN: 2153-7003, Jul. 2016, pp. 693–696. DOI: 10.1109/IGARSS.2016.7729174.

[202] J. McGee, S. J. Mathew, and F. Gonzalez, "Unmanned aerial vehicle and artificial intelligence for thermal target detection in search and rescue applications," in *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, ISSN: 2575-7296, Sep. 2020, pp. 883–891. DOI: 10.1109/ICUAS48674.2020.9213849.

[203] A. Agha, K. Otsu, B. Morrell, *et al.*, "NeBula: TEAM CoSTAR's robotic autonomy solution that won phase II of DARPA subterranean challenge," *Field Robotics*, vol. 2, no. 1, pp. 1432–1506, Mar. 10, 2022. DOI: 10.55417/fr.2022047.

[204] H. Choset and P. Pignon, "Coverage path planning: The boustrophedon cellular decomposition," *Field and Service Robotics*, pp. 203–209, 1998, ISBN: 978-1-4471-1275-4. DOI: 10/bkwz8h. arXiv: 1011.1669v3.

[205] R. Mannadiar and I. Rekleitis, "Optimal coverage of a known arbitrary environment," in *2010 IEEE International Conference on Robotics and Automation*, ISSN: 1050-4729, May 2010, pp. 5525–5530. DOI: 10.1109/ROBOT.2010.5509860.

[206] L. Paull, C. Thibault, A. Nagaty, M. Seto, and H. Li, "Sensor-driven area coverage for an autonomous fixed-wing unmanned aerial vehicle," *IEEE Transactions on Cybernetics*, vol. 44, no. 9, pp. 1605–1618, Sep. 2014. DOI: 10/f6hrz5.

[207] L. Paull, M. Seto, J. J. Leonard, and H. Li, "Probabilistic cooperative mobile robot area coverage and its application to autonomous seabed mapping," *International Journal of Robotics Research*, vol. 37, no. 1, pp. 21–45, 2018. DOI: 10/gcxmf9.

[208] G. Hitz, E. Galceran, M.-È. Garneau, F. Pomerleau, and R. Siegwart, "Adaptive continuous-space informative path planning for online environmental monitoring," *Journal of Field Robotics*, vol. 34, no. 8, pp. 1427–1449, 2017. DOI: 10/gcj7sp.

[209] B. Ayton, "Risk-bounded autonomous information gathering for localization of phenomena in hazardous environments," Ph.D. dissertation, Massachusetts Institute of Technology, 2017.

[210] A. Arora, R. Fitch, and S. Sukkarieh, "An approach to autonomous science by modeling geological knowledge in a bayesian framework," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Sep. 9, 2017, pp. 3803–3810. DOI: 10/ggb45r.

[211] A. Arora, P. M. Furlong, R. Fitch, S. Sukkarieh, and T. Fong, "Multi-modal active perception for information gathering in science missions," *Autonomous Robots*, vol. 43, no. 7, pp. 1827–1853, 2019, Publisher: Springer US ISBN: 1051401909. DOI: 10/ggb49n.

[212] G. Best and G. A. Hollinger, "Decentralised self-organising maps for multi-robot information gathering," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, USA: IEEE, Oct. 24, 2020, pp. 4790–4797. DOI: 10.1109/IROS45743.2020.9341106.

[213] S. McCammon, D. Jones, and G. Hollinger, "Topology-aware self-organizing maps for robotic information gathering," p. 8, 2020.

[214] S. McCammon and G. A. Hollinger, "Topological path planning for autonomous information gathering," *Autonomous Robots*, vol. 45, no. 6, pp. 821–842, Sep. 1, 2021. DOI: 10.1007/s10514-021-10012-x.

[215] I. Kostavelis, K. Charalampous, A. Gasteratos, and J. K. Tsotsos, "Robot navigation via spatial and temporal coherent semantic maps," *Engineering Applications of Artificial Intelligence*, vol. 48, pp. 173–187, Feb. 1, 2016. DOI: 10.1016/j.engappai.2015.11.004.

[216] C. M. Bishop, *Pattern Recognition and Machine Learning*, M. Jordan, J. Kleinberg, and B. Schölkopf, Eds. Springer-Verlag New York, 2006.

[217] E. Fix and J. L. Hodges, "Discriminatory analysis. nonparametric discrimination: Consistency properties," *International Statistical Review / Revue Internationale de Statistique*, vol. 57, no. 3, pp. 238–247, 1989, Publisher: [Wiley, International Statistical Institute (ISI)]. DOI: 10.2307/1403797.

[218] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Advances in Neural Information Processing Systems*, vol. 30, Curran Associates, Inc., 2017.

[219] J. E. San Soucie, H. M. Sosik, and Y. Girdhar, *Streaming gaussian dirichlet random fields for spatial predictions of high dimensional categorical observations*, Feb. 23, 2024. DOI: 10.48550/arXiv.2402.15359. arXiv: 2402.15359[cs].

[220] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1258–1276, Dec. 1, 2013. DOI: 10.1016/j.robot.2013.09.004.

[221] H. E. Clifton, C. V. W. Mahnken, J. C. Van Derwalker, and R. A. Waller, "Tektite 1, man-in-the-sea project: Marine science program," *Science*, vol. 168, no. 3932, pp. 659–663, May 8, 1970, Publisher: American Association for the Advancement of Science. DOI: 10.1126/science.168.3932.659.

[222] Agisoft, *Metashape professional*, version 2.1.0, 2023.

[223] X. Wang, X. Zhang, Y. Cao, W. Wang, C. Shen, and T. Huang, "SegGPT: Towards segmenting everything in context," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, Paris, France: IEEE, Oct. 1, 2023, pp. 1130–1140. DOI: 10.1109/ICCV51070.2023.00110.

[224] C. A. Burge, M. E. Mouchka, C. D. Harvell, and S. Roberts, "Immune response of the caribbean sea fan, gorgonia ventalina, exposed to an aplanochytrium parasite as revealed by transcriptome sequencing," *Frontiers in Physiology*, vol. 4, Jul. 25, 2013, Publisher: Frontiers. DOI: 10.3389/fphys.2013.00180.

[225] A. M. Tracy, E. Weil, and C. D. Harvell, "Warming and pollutants interact to modulate octocoral immunity and shape disease outcomes," *Ecological Applications*, vol. 30, no. 2, e02024, 2020, _eprint: https://esajournals.onlinelibrary.wiley.com/doi/pdf/10.1002/eap.2024. DOI: 10.1002/eap.2024.

[226] A. A. M. Becker, M. A. Freeman, and M. M. Dennis, "A combined diagnostic approach for the investigation of lesions resembling aspergillosis in caribbean sea fans (gorgonia spp.)," *Veterinary Pathology*, vol. 60, no. 5, pp. 640–651, Sep. 1, 2023, Publisher: SAGE Publications Inc. DOI: 10.1177/03009858231173355.

[227] M. Roznere, A. K. Pediredla, S. E. Lensgraf, Y. Girdhar, and A. Q. Li, "Underwater dome-port camera calibration: Modeling of refraction and offset through n-sphere camera model," in *IEEE International Conference on Robotics and Automation (ICRA)*, tex.copyright: All rights reserved, 2024.

[228] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, ISSN: 10636919, 2012, pp. 3354–3361. DOI: 10/gf7nxj. arXiv: 1612.07695.

[229] S. Cayci, A. Eryilmaz, and R. Srikant, "Budget-constrained bandits over general cost and reward distributions," in *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, S. Chiappa and R. Calandra, Eds., ser. Proceedings of Machine Learning Research, vol. 108, PMLR, Aug. 26, 2020, pp. 4388–4398. arXiv: 2003.00365.

[230] H. B. Mann and A. Wald, "On stochastic limit and order relationships," *The Annals of Mathematical Statistics*, vol. 14, no. 3, pp. 217–226, Sep. 1943, Publisher: Institute of Mathematical Statistics. DOI: 10.1214/aoms/1177731415.

[231] F. P. Cantelli, "Sui confini della probabilità," in *Atti del Congresso Internazionale dei Matematici: Bologna del 3 al 10 de settembre di 1928, Vol. 6, 1929 (Comunicazioni, sezione IV (A)-V-VII), págs. 47-60*, Section: Atti del Congresso Internazionale dei Matematici: Bologna del 3 al 10 de settembre di 1928, 1928, pp. 47–60.

[232] A. S. M. Shihavuddin, N. Gracias, R. Garcia, A. C. R. Gleason, and B. Gintert, "Image-based coral reef classification and thematic mapping," *Remote Sensing*, vol. 5, no. 4, pp. 1809–1841, Apr. 2013, Number: 4 Publisher: Multidisciplinary Digital Publishing Institute. DOI: 10.3390/rs5041809.

[233] O. Beijbom, P. J. Edmunds, C. Roelfsema, *et al.*, "Towards automated annotation of benthic survey images: Variability of human experts and operational modes of automation," *PLOS ONE*, vol. 10, no. 7, e0130312, Jul. 8, 2015, Publisher: Public Library of Science. DOI: 10.1371/journal.pone.0130312.

[234] A. R. Rashid and A. Chennu, "A trillion coral reef colors: Deeply annotated underwater hyperspectral images for automated classification and habitat mapping," *Data*, vol. 5, no. 1, p. 19, Mar. 2020, Number: 1 Publisher: Multidisciplinary Digital Publishing Institute. DOI: 10.3390/data5010019.

[235] M. S. A. C. Marcos, M. N. Soriano, and C. A. Saloma, "Classification of coral reef images from underwater video using neural networks," *Optics Express*, vol. 13, no. 22, pp. 8766–8771, Oct. 31, 2005, tex.ids= Marcos2005a publisher: Optica Publishing Group. DOI: 10.1364/OPEX.13.008766.

[236] Australian Institute of Marine Science (AIMS). "AIMS long-term monitoring program: Video and photo transects (great barrier reef)." (), Available: https://doi.org/10.25845/5c09bc4ff315c.

[237] UNEP-WCMC, S. Andréfouët, F. E. Müller-Karger, and C. Kranenburg. "The millennium coral reef mapping project: Understanding, classifiying and mapping coral reef structures worldwide using high resolution remote sensing spaceborne images," Millennium Coral Reef Mapping Project Seascape. (2012), Available: http://imars.marine.usf.edu/MC/.

[238] M. S. Bewley, N. Nourani-Vatani, D. Rao, B. Douillard, O. Pizarro, and S. B. Williams, "Hierarchical classification in AUV imagery," in *Field and Service Robotics*, L. Mejias, P. Corke, and J. Roberts, Eds., vol. 105, Series Title: Springer Tracts in Advanced Robotics, Cham: Springer International Publishing, 2015, pp. 3–16. DOI: 10.1007/978-3-319-07488-7_1.

[239] D. M. Steinberg, A. Friedman, O. Pizarro, and S. B. Williams, "A bayesian nonparametric approach to clustering data from underwater robotic surveys," presented at the 15th International Symposium on Robotics Research, 2011, pp. 1–16.

[240] S. Sandin, J. Smith, and B. Zgliczynski, "The scripps oceanography 100 island challenge," Mar. 20, 2019.