2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Adaptive multi-altitude search and sampling of sparsely distributed natural phenomena

Jessica E. Todd[1], Seth McCammon[2], Yogesh Girdhar[2], Nicholas Roy[3] and Dana Yoerger[2]

***Abstract*— In this paper, we propose a novel method for autonomously seeking out sparsely distributed targets in an unknown underwater environment. Our Sparse Adaptive Search and Sample (SASS) algorithm mixes low-altitude observations of discrete targets with high-altitude observations of the surrounding substrates. By using prior information about the distribution of targets across substrate types in combination with belief modelling over these substrates in the environment, high-altitude observations provide information that allows SASS to quickly guide the robot to areas with high target densities. A maximally informative path is autonomously constructed online using Monte Carlo Tree Search with a novel acquisition function to guide the search to maximise observations of unique targets. We demonstrate our approach in a set of simulated trials using a novel generative species model. SASS consistently outperforms the canonical boustrophedon planner by up to 36% in seeking out unique targets in the first 75 - 90% of time it takes for a boustrophedon survey. Additionally, we verify the performance of SASS on two real world coral reef datasets.**

I. INTRODUCTION

Seeking out and sampling environmental phenomena is crucial for modelling geophysical and ecological processes in ocean environmental monitoring. Autonomous underwater vehicles (AUVs) have been widely used in the mapping of continuous scalar oceanic phenomena [1] such as chemical plumes [2], chlorophyll levels [3], temperature, salinity [4] and bathymetric depth [5]. Many classical informative path planning (IPP) approaches assume smooth gradients of the phenomena distribution to plan effective measurement strategies [6], [7], gradients that are not present for sparsely distributed phenomena. The problem of choosing where to take measurements is challenging when the phenomena of interest are discrete and sparsely distributed in space or time, such as when mapping a particular benthic species.

The motivating problem for this paper is monitoring coral reef environments. Scientists are interested in mapping the prevalence of infectious diseases, such as the Stony Coral Tissue Loss disease [8], which affects a subset of sparsely distributed coral species on a reef. Quantifying the prevalence of this disease requires observing as many unique samples of these species as possible. Canonically, searching for sparse discrete phenomenon, a problem we define as the Sparse Search-and-Sample problem, has been achieved using uniform coverage trajectories such as a preprogrammed boustrophedon ('lawnmower') trajectory [9]. Uniform coverage approaches are highly predictable (a desirable feature for expensive sensor assets), but are also inefficient as significant mission time can be expended over low value regions of the environment with no phenomena present. When the total mission duration is constrained by a limited battery life, data gathering requires either multiple deployments, or risks missing important measurements. Targeted planning can instead prioritize search over higher value regions. However the key challenge is determining where to look for these sparse signals in an efficient manner. Our approach draws inspiration from the ecological literature, where species distributions are modelled as correlations between observable and latent abiotic and biotic processes [10]. These processes drive the presence of particular species [11], and shape the broader environmental morphology, or substrates, in which these species are found. We use a correlation model between high-altitude sensing of environmental substrate types (e.g. coral, rock, sand) that provides wider coverage and context of the environment, and low-altitude detection of discrete phenomenon of interest within these substrates.

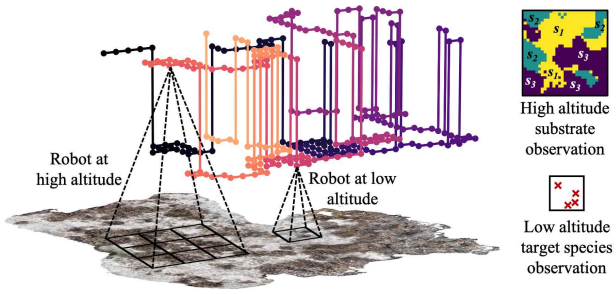


Fig. 1: The Sparse Adaptive Search and Sample planning algorithm uses a multi-altitude trajectory to gain information about both the terrain substrate type and locate target species, for an AUV operating over a coral reef.

Our **S**parse **A**daptive **S**earch and **S**ample (SASS) algorithm (Fig. 1) combines two sensing modes — a broad coverage visual sensor utilised at higher altitude capable of detecting visual substrate types, and a low altitude low coverage visual sensor capable of directly observing the target phenomena — with an anytime probabilistic planning framework, as a method of biasing trajectories towards areas of the map likely to yield observations of our target species. The primary contributions of this work are (1) the formalisation of the Sparse Search-and-Sample problem as

This work was supported in part by NSF Grants 2133029 and 2333604

[1]Jessica E. Todd is with the MIT-WHOI Joint Program in Applied Ocean Science and Engineering at the Woods Hole Oceanographic Institution (WHOI), and with the Department of Aeronautics and Astronautics at the Massachusetts Institute of Technology (MIT) jetodd@mit.edu

[2]Seth McCammon, Yogesh Girdhar and Dana Yoerger are with the Applied Ocean Physics and Engineering Department at the Woods Hole Oceanographic Institution, Woods Hole, MA, USA {smccammon, yogi, dyoerger}@whoi.edu

[3]Nicholas Roy is with the Department of Aeronautics and Astronautics at the Massachusetts Institute of Technology nickroy@csail.mit.edu

a partially observable Markov decision process (POMDP), (2) a correlation model that combines a generative species model and Gaussian process classification substrate model, (3) integration of this correlation model into an anytime non-myopic multi-altitude planner which enables efficient sampling of a target phenomena using heterogeneous sensor modes, and (4) a demonstration of this approach using simulated and real datasets. The remainder of this paper is organised as follows: Section II outlines related work in the field of adaptive environmental mapping and multi-altitude path planning, Section III formalises the problem, Section IV details the components of the planning algorithm, and Section V presents experiments with both simulated and photomosaic datasets.

II. RELATED WORK

The majority of IPP approaches to environmental monitoring focus on sample collection within continuous scalar data fields [12], [13]. Gaussian processes (GP) are a popular technique for modelling the distribution of target phenomenon [14], [15], as the GP regression model provides both a mean estimate of the scalar field state and a formal measure of predictive uncertainty on the accuracy of this model [16]. A robotic planner can then develop strategies to reduce this uncertainty [13], a technique that has been widely applied in the ocean robotics context. Flaspohler et al. [6] described a maximum-seeking algorithm that maintains a GP belief model in mapping a scalar chemical field. Like SASS, they utilised a receding horizon Monte Carlo Tree Search (MCTS) planner, with a maximum-value heuristic reward, to move toward the global maxima. The search problem is similar to ours in that it is characterised by locating a discrete phenomena, i.e., the source of a chemical plume, however this search utilises sampling of a continuous chemical distribution in the surrounding water. Similarly, Manjanna and Dudek [12] used GPs in maximising coverage over a coral reef using visual sensors. They described a variable resolution coverage planner which takes into account the spatial variation in the underlying field, thereby minimising mission time while maximising coverage. The application of multiple GPs to the ocean sampling problem was explored by Das et al. [7]. GP regression models were fitted to environmental covariates such as chlorophyll concentration, salinity, and temperature, to predict plankton abundance and thus maximise plankton sampling. A limitation of these approaches within the context of our problem is that while overall coral coverage or plankton abundance can be approached as a spatially varying continuous field, a sufficiently sparse target distribution like sparse coral targets cannot. We address this limitation by using a GP classification-based belief model to discretize the world into regions of different target likelihood, thereby maximising coverage of discrete targets.

The use of different sensor modes to inform search was examined by Arora et al. [17], who employed a Bayesian network to characterize probabilistic relationships between UV sensing and visual sensing, and a discrete categorical phenomena of interest (rock type) in an IPP setting. Similar to our approach of dividing observations between high-altitude, high-coverage categorical data and low-altitude, low-coverage discrete data, an MCTS planner was used by Arora et al. [17] to determine which sensors to reason most effectively about rock type. A simplistic Gaussian model was used to impose spatial relationships on these probabilistic relationships, allowing limited prediction of unseen areas of the world. In comparison, our proposed use of multiple Gaussian process models allows for non-myopic planning over predicted substrates that have varying lengthscales.

While it has seen limited application in the ocean domain, the concept of planning over three-dimensional space with a single robot to utilise different levels of sensor coverage has been more widely explored in the aerial domain [18]. Popovic et al. [19] used a GP prior and an altitude-dependent sensor model to map weed coverage using a UAV. Coverage of the world was driven by both reducing uncertainty and prioritising highly-infested areas (determined by a thresholding on the visual information), and the UAV traded between gaining noisy broad coverage information of the world with targeted highly accurate information using a greedy lookahead planner. Building on this work, Stache et al. used a semantic classifier to adapt boustrophedon surveys to different altitudes to maximise segmentation accuracy of different agricultural terrain types, utilising a semantic ratio derived from the classifier as the reward function, rather than a probabilistic belief model [20]. The UAV in both cases used a single visual data type, with information from different resolutions fused into a single map. In contrast, our work seeks to use different data types at different altitudes to guide exploration, as when operating underwater the target phenomena are too small to be observed from high altitude, and visual classes are difficult to determine from close range. This requires our decision-making algorithm to simultaneously reason over the different cost of high- and low-altitude sampling as well as the different informational utility of each type of observation.

III. PROBLEM FORMULATION

For the Sparse Search-and-Sample problem, we consider a robot agent with a fixed mission duration B, tasked with seeking out and observing a set of sparsely distributed target phenomena in a discrete world. The Sparse Search-and-Sample problem is formalised as solving a Partially Observable Markov Decision Process (POMDP) [21] represented by a tuple $(\mathcal{S}, \mathcal{A}, T, Z, \mathcal{O}, R, \gamma)$ defined below.

The **state space** $\mathcal{S}$ is composed of the world state u and v, the robot's current location $\mathbf{q}$, the remainder of the cost budget $0 \leq b_\tau \leq B$ for mission time τ, and the set of visited cells in the world ξ. The robot can be at any state $\mathbf{q} = (q_x, q_y, q_z) \in Q = \{m \times n \times 2\}$ where q_x and q_y comprise the robot's 2D location in the $m \times n$ gridded world, and $q_z \in \{z_L, z_H\}$ is its altitude. The world state is static, initially unknown, and partially observable, given by $v \in V = \{1, ..., k\}^{m \times n}$ describing the substrate type present in each cell for k substrate types and $u \in U = \mathbb{Z}+^{m \times n}$ is the number of unique targets in each cell. Since reward is only

gathered when a unique target is viewed, the set of visited cells must also be maintained, $\xi = \{0,1\}^{m \times n}$. Thus the state space is $S = Q \times U \times V \times \xi \times B$. The **action space** $\mathcal{A}$ consists of the set of valid transitions between adjacent states. In this paper we use a naive 4-connected grid with two fixed altitudes, but our planner is agnostic to the particular state transition model, and smooth paths from splines or lattices (e.g. [19]) could be used to better match a real-world AUV motion model. The **transition model** T for the robot state $\mathbf{q}$ and remaining budget b is fully deterministic and u, v and ξ remain static. Each action incurs a time cost which is subtracted from the budget. **Reward** R is attained when the robot visits a grid cell at a low altitude and observes the target state of the cell, given by the reward function

$$R(\mathbf{q}) = \begin{cases} u(q_x, q_y) & q_z = z_L, \quad \xi(q_x, q_y) = 0 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

for $b \geq 0$. This function encourages visiting cells with higher numbers of targets, but the ξ dependence means that the reward per cell can only be collected once. The **discount factor** $\gamma : 0 \leq \gamma \leq 1$ encourages robots to collect rewards sooner. The **observation model** Z and **observation space** $\mathcal{O}$ are described in section III-A. The objective of the Sparse Search-and-Sample problem is to maximise the number of observations of targets over the trajectory $\phi = (\mathbf{q}_1, ..., \mathbf{q}_t)$ of the robot before the cost budget is exhausted

$$R(\phi) = \sum_{\tau=1}^{t} R(\mathbf{q}_\tau) \quad \text{s.t.} \ \sum b_\tau \geq 0. \tag{2}$$

A. Target-substrate observation model

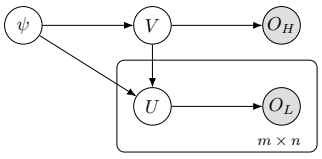


Fig. 2: Simplified generative model for substrate and target species. Observations O_H and O_L are made by the robot's sensors, observing either the substrate set V or the targets U, where both are characterised by some underlying environmental processes ψ, and U by the substrate V

SASS assumes a correlation model between target species presence and overarching substrate of the region as shown in Fig. 2, as well as substrate correlation between neighbouring cells. Interactions between environmental processes drive the physical properties of a region, which can be grouped into visible morphologies or substrates. For example, strong currents in a region combined with a change in depth and consequently light will lessen the habitability of a region, reducing overall coral coverage and resulting in areas of non-coral substrate types. Given that species have specific habitat requirements this will also affect the distribution of target species in this region. SASS assumes a known number of k substrate types, each with an expected target species density $\rho \in \{\rho_1, ...\rho_k\}$ which is known *a priori*. Substrate types can be viewed at high altitude where broader morphology can be identified (Eq. 3), whereas observation of target species is restricted to low altitude (Eq. 4). This mirrors real underwater operating conditions where light attenuation reduces the ability to identify species with increased depth.

$$O_H(\mathbf{q}) \in \{1, ..., k\}^{r_H \times r_H}, \text{ if } q_z = z_H, \tag{3}$$

$$O_L(\mathbf{q}) \in [0, d]^{r_L \times r_L}, \text{ if } q_z = z_L. \tag{4}$$

where r_H and r_L are the sensor footprints at high and low altitude. We assume a fully deterministic **observation model** Z, with the robot able to observe the visible portion of the true world state in its given grid cell. The optimal solution to a general POMDP is a policy ϕ^* that maximises the number of unique targets observed. Computing the solution to a POMDP is PSPACE-hard [22], [23]. The world state is the only unknown component and is static in time, thus SASS uses a receding horizon MCTS planner with an acquisition function in the rollouts to tractably solve the POMDP.

IV. THE SPARSE ADAPTIVE SEARCH AND SAMPLE ALGORITHM

The **S**parse **A**daptive **S**earch and **S**ample (SASS) algorithm addresses the complexities of the Sparse Search-and-Sample problem, namely the sparsity of the target signal and a constrained mission time, through the four steps of a receding horizon planning framework:

1) Predicting low-altitude target densities from high altitude substrate observations
2) Allocating mission time between gathering broad coverage information about the substrate classes present in the world at high altitude, and searching for the target phenomena at low altitude via an acquisition function
3) Planning a reward-maximising path through the environment under time constraints using an anytime MCTS planner with a receeding horizon.
4) Execure planned trajectory and collect new observations and high and low altitudes

A. World State Prediction

The first step of the planning process is to estimate the distribution of targets, u, and substrates, v, from the observations, O, obtained by the robot. We estimate the probability of a given visual substrate at any point in the world using a binary Gaussian Process (GP) classifier. In SASS, a single GP is trained on each of the visual substrates using the *one-versus-all* approach. Probabilities from these k GPs are combined and normalized to produce a unified belief map over k classes. The world state estimate is then based on the *a priori* known density values ρ for each substrate type, weighted by the substrate likelihood

$$\tilde{\rho}(\mathbf{q}) = \Sigma_{i=1}^{k} \rho(i) \cdot P(v(\mathbf{q}) = i) \tag{5}$$

A GP is a distribution over functions that can be fully specified by a mean function and covariance function $f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), K(\mathbf{x}, \mathbf{x}'))$. GP classification (GPC) is a generalization of the more common GP regression [16]. In GPC, given a set of training points $\mathbf{X} = [\mathbf{x_1}, .., \mathbf{x_N}]$ and the corresponding class labels $\mathbf{c} = [c_1, ..., c_N]$, we aim to predict

the probability that a test point $\mathbf{x}^*$ belongs to a given class. The latent function $f(\mathbf{x})$ relating a class label to training points is assumed to have a GP prior, and is mapped to a interval $[0, 1]$ using a sigmoid function, where 1 denotes positive class membership and 0 denotes the negative. For a given test point the mean posterior of the GP models the likelihood of class membership.

B. Acquisition function

The intractability of the Sparse Search-and-Sample problem means we cannot naïvely apply the POMDP reward function given in Eq. 1. Observing substrates has no direct utility with respect to the Sparse Search-and-Sample objective. Instead we use an acquisition function that encourages the robot to balance exploring new areas of the map at high altitude to gain substrate information, and thus the distribution of high target-density regions, while ensuring that it spends sufficient mission time at low altitude in valuable substrates to maximise observations of the target phenomena (Eq. 6). The acquisition function balances reducing uncertainty in its belief, while maximising observations of the target. Doing so better distributes the gradient of reward though the robot's action space, making search more tractable. The acquisition function we use in the MCTS planner estimates reward along the proposed trajectory ϕ as

$$R(\phi) = (1-\alpha)\sum_{\mathbf{q}\in\phi}\tilde{\rho}(q_x, q_y, q_z = z_L) \;+ \\ \alpha\sum_{\mathbf{q}\in\phi} H(A_q(q_x, q_y, q_z = z_H)) \quad (6)$$

The first term captures the exploitation component, where reward is acquired based on the expected target density $\tilde{\rho}$ in a given cell $\mathbf{q}$ from Eq. 5, if the robot is at low altitude z_L. The second term captures the exploration component. Uncertainty in the substrate types at a given location is quantified as the entropy H of the Gaussian process belief model where $H(\mathbf{q}) = \sum_{i=1}^{k} P_i * \log_2(P_i)$, with P being the probability of a substrate. Entropy will be highest in regions where the GP classifier is unsure of the class label, and is reduced when a substrate is observed. This term drives the robot to explore beyond its frontier. $A_q(\mathbf{q})$ describes the higher altitude-dependent sensor footprint centred around the current robot location i.e. $r_H \times r_H$ grid cells, as shown in Figure 1. Entropy is calculated over the entire footprint of the path $A_q(\phi)$, ignoring grid cells that have already been explored at a low altitude ($\xi(\mathbf{q}) = 1$). The exploration constant α balances this explore-exploit trade-off, and both the *explore* and *exploit* terms are normalized to $[0, 1]$.

C. Monte-Carlo Tree Search

To find a near-optimal action sequence ϕ^* that optimizes the acquisition function, we use receding horizon MCTS [24]. MCTS is an anytime planning algorithm which performs a guided search through $\mathcal{A}$, consisting of a *selection*, *expansion*, *simulation* and *back-propagation* phase. A tree of partial action sequences is constructed incrementally by *selecting* and *expanding* leaf nodes using the Upper Confidence Bound heuristic that balances selection of high reward partial sequences with balanced exploration of all possible action sequences [25]. At each iteration, the partial action sequence is forward *simulated* to a fixed horizon using a policy of random action selection over all 5 actions. We use this policy as it is likely to mix both high- and low-altitude observation. Actions that move the robot into a previously unexplored grid cell are randomly selected from first, and repeat cell visits are considered if there are no other viable actions. Reward along the completed trajectory is evaluated according to the acquisition function in Eq. 6. Finally the reward for a given rollout is *back-propagated* up the tree to update the overall value of all parent nodes in the action sequence. After exhaustion of a set computational budget, the partial action sequence in the tree that gives the highest average reward according to Eq. 6 is selected, and the first action along this path executed by the robot. For a complete description of MCTS, we refer the reader to [24]. Fig. 3 gives an example of the progression of the SASS planning algorithm over time for a simulated mission. The AUV remains at a high altitude, gathering larger coverage data of substrates, until it observes a high value substrate and then changes altitude to collect target observations. Once the AUV begins to move beyond the predicted lengthscale of a given substrate, it returns to high altitude to gather more knowledge of the substrate distribution.

V. EXPERIMENTS

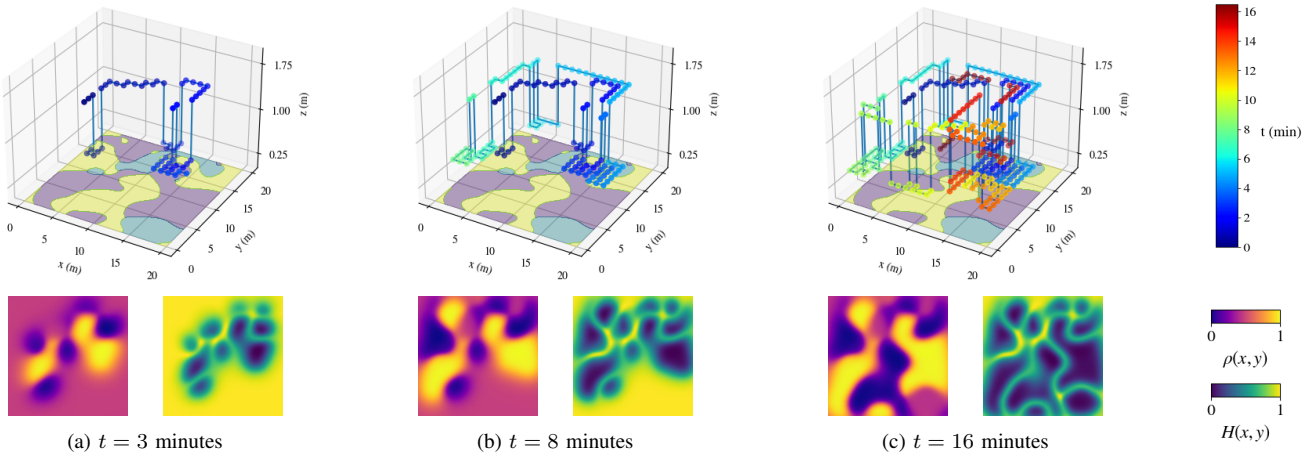
The empirical performance of the SASS algorithm was assessed in simulation using both real-world and synthetic environments. We compared the performance against the single-altitude boustrophedon planner, as this coverage algorithm is standard practice for AUVs mapping coral reefs. To ensure complete coverage of the target phenomena, the robot performed the boustrophedon trajectory at the lower altitude. Performance was evaluated by comparing accumulated reward (Eq. 2) against mission time.

A. Synthetic Environments

We generated synthetic coral reefs using randomly-generated distributions of substrates populated with species from Table I. Each simulated reef is 20 m $\times$ 20 m in size, discretised into 1 m cells for planning.

Substrate Classes: Each of k simulated substrates are drawn from a GP prior with a squared exponential covariance function and zero mean, with hyperparameters $l = 2.0, \sigma^2 = 1.5$. The substrates are passed through a softmax function to produce a simulated distribution of substrates across the world. These values were then taken as the probability of a given visual substrate, and the ground truth environment generated based on the most likely substrate in any given location. For our tests, we used $k = 3$ substrates, however SASS can accept an arbitrary number of $k \geq 2$ substrates.

Simulated Target Phenomena: To simulate realistic sparsely distributed biological phenomena, target coral were simulated using Thomas point processes [26]. Thomas point



(a) $t = 3$ minutes (b) $t = 8$ minutes (c) $t = 16$ minutes

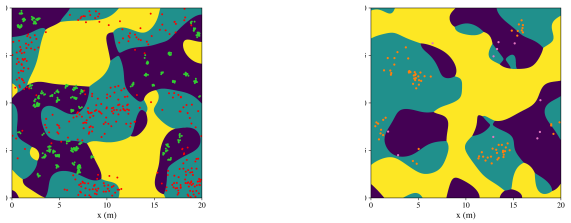
Fig. 3: Example of a simulated mission over three timesteps on a synthetic world. The top three figures illustrate the 3D trajectory of the AUV. The lower figures shows a pair of maps containing a belief map predicting u (left), and entropy map showing uncertainty of the substrate classification (right).

TABLE I: Example instances of target species

λ	μ	σ	**Spatial property**	**Example**
0.1	10	0.1	sparsely distributed small clusters (SS)	
0.5	10	0.1	densely distributed small clusters (DS)	
0.1	10	0.5	sparsely distributed large clusters (SL)	
0.2	50	0.5	densely distributed large clusters (DL)	
0.1	10	1	homogeneously sparse (HS)	
0.5	10	1	homogeneously dense (HD)	

processes are a type of clustered Poisson point process widely used in spatial statistics and ecological modelling to model discrete occurrences of events over a specified spatial window. Thomas point processes are simulated by specifying an intensity value $\lambda > 0$ over a simulation window $W \subseteq \mathbb{R}$. This intensity measure describes the underlying parent Poisson point process. Poisson point processes are collections of random points in a finite space where the number of points is a random variable with a Poisson distribution. For each point in the parent process, a Poisson number of daughter points with mean $\mu > 0$ is simulated within a variance σ^2 of the parent process. Different combinations of parameters were investigated to capture different typical spatial characteristics seen on real coral reefs. Six representative species were selected, with the parameters shown in Table I. Each synthetic world consists of two substrates assigned a randomly-chosen species from Table I, and one empty substrate. Individual coral targets are distributed according to their corresponding point process, weighted by the prevalence of the substrate at each location, resulting in more realistic gradients in distribution intensity across the world. We generated a total of 45 synthetic reefs. Two example worlds are shown in Fig. 4.

For each mission simulation, the robot was randomly initialised in the world and both the boustrophedon and SASS planner were used to plan over a mission budget of 3000 seconds. The MCTS planner planned out to a budget h of 100 seconds, with 1000 rollouts per planning cycle. At low altitude the robot has a sensor footprint of 1 m^2 and travels at 0.2 m/s. At high altitude the footprint is 9 m^2, and travel speed is 0.5 m/s, since the vehicle does not have to go slowly to avoid colliding with coral heads. Transiting between low and high altitude takes 5 seconds. In all cases, the robot has *a priori* knowledge of average intensities (expected targets per 1x1m grid cell) in each of the substrates. The world belief model hyperparameters were trained prior to the mission. 25 robot missions were run on each simulated world for a total of 1125 missions. From experimentation, it was determined that the optimal exploration constant value is $\alpha = 0.15$.



(a) DS and HD target species (b) SL and HS target species

Fig. 4: Two randomly generated substrate worlds populated by instances of two species and thinned according to the probability distribution of the substrates to produce realistic gradients of species across substrate borders.

B. Real-World Reef Orthomosaic Environments

We also evaluated the performance of SASS on real world coral reef datasets, shown in Fig. 8. Images were collected

using the CUREE robot [27] at Tektite Reef and Booby Rock on St John in the U.S. Virgin Islands. 4K images collected by CUREE were rectified and color-corrected using DeepSeeColor [28] and an orthomosaic of the reef was generated in Metashape from these images (Fig. 8a), which served as the ground truth for the mission.

1) Visual Classes: The photomosaic datasets were classified based on visual features using Realtime Online Spatiotemporal Topic modelling (ROST) [29]. For a detailed presentation of the ROST model, refer to [29]. The ROST model is tuned by varying three scalar hyperparameters and the cell size. In this work, the number of topics K was limited to a maximum of three to ensure less patchy topics, and the hyperparameters set to $\alpha = 0.8$, $\beta = 0.4$, $\gamma = 0.000001$, with a cell size of 0.2 m and 0.5 m for the *Tektite* and *Booby Rock* datasets respectively. As can be seen in Figure 8b, the substrates correspond roughly to a *sand*, and two different *coral* substrates.

2) Target phenomena: The target phenomena chosen for the sparse search-and-sample problem was the stony coral species *Porites astreoides*, a common coral species in the Caribbean. Each *Porites astreoides* in the orthomosaic was manually segmented and a binary mask of the segments generated. Locations of the target coral were taken to be the centroid of the coral head, and expected densities of coral calculated for each substrate class as in the simulated worlds.

As with the synthetic worlds, the real world environments were divided into 1 x 1 m grid for the purposes of planning. 25 missions were run on the *Tektite* and *Booby Rock* datasets in the same manner as the simulated world missions, and compared with a boustrophedon trajectory that ensured full coverage of the reef. Full coverage at low altitude took 3600s and 7200s for *Tektite* and *Booby Rock* respectively. The same robot parameters were used as in the simulated case. The exploration constant was set to 0.1 and 0.12 for *Tektite* and *Booby Rock* respectively.

VI. RESULTS

A. Performance on Synthetic Reef Environments

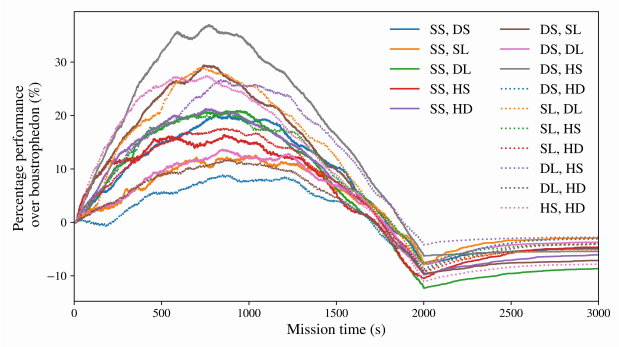
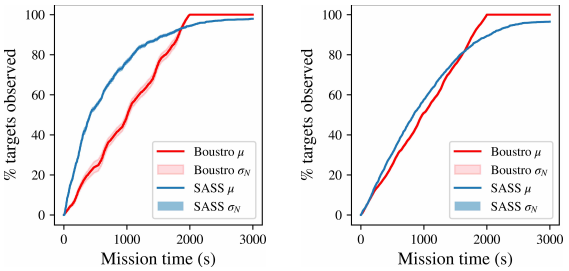


Fig. 5: Performance in terms of difference in percentage of targets observed of SASS compared to boustrophedon planner. Each line is the performance difference averaged across all worlds with a given species combination. $\alpha = 0.15$ for all tests.

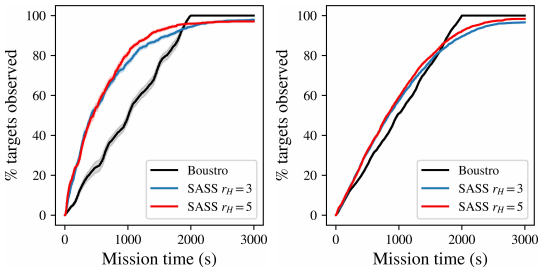
For each of the 45 synthetic worlds, we compare the percentage of targets observed by SASS and a boustrophedon planner, averaged across 25 simulated missions with different random initial starting points. SASS was run with a MCTS horizon of 100s over a 3000s total mission time. The low-altitude boustrophedon is able to achieve complete coverage of a synthetic environment in 2000 seconds. Figure 5 shows the comparative performance of SASS compared to the boustrophedon planner, in terms of the difference in percentage of targets observed over time. Each line on the graph represents the average performance difference of SASS compared to boustrophedon, across all worlds containing a particular combination of two species, to account for variation in substrate distribution. Across all worlds there is a peak performance improvement of between 8% and 36% of SASS over boustrophedon centred around 750-1000s into the mission, with performance difference then decreasing over time. This suggests that given the restricted mission budget of many underwater vehicles and the typical sizes of the environments being explored (on the order of tens or hundreds of square metres) preventing full environment coverage, SASS is consistently a better planning choice for maximising observations of the target species. The highest performance corresponds to worlds that both contain species DS and DL, dense clusters of species. The lowest performing band consists of worlds containing large areas of homogenously distributed (HD) species, as a fairly homogenous world will see little advantage of a targeted planner. Separating out the results by world, in 40 worlds the SASS algorithm provides a significant increase in performance over boustrophedon in the first $\sim$1500-1800s of the mission, depending on the world. An example is shown in Figure 6a, where SASS significantly outperforms boustrophedon in the first 1860s. SASS drives the robot to a high altitude to learn substrate distributions, and thus identifying likely locations of high target density and moving directly to these areas, focusing initial target detection on high-value substrates. Between 1860s and 2000s, boustrophedon matches or exceeds SASS performance since it is more rapidly achieving complete coverage. After 2000s SASS asymptotically approaches the performance of boustrophedon, and given a long enough time it will also achieve full low-altitude coverage of the world. For 5 worlds, SASS only marginally outperforms boustrophedon, an example of which is shown in Figure 6b. These worlds are characterized by fairly uniform distribution of high reward across the world, usually a combination of HD, DL or DS such as that shown in Fig. 4a. As such, a coverage algorithm like boustrophedon can achieve reward at a rate similar to targeted planner like SASS.

Fig 7 shows the results of running SASS with different sensor footprint r_H sizes, holding all other robot parameters constant, where $r_H = j$ gives a footprint of $j \times j$ grid cells. Different exploration constants α were tested for each sensor footprint and 0.15 and 0.08 found to give the highest performance for $r_H = 3$ (9 m^2) and $r_H = 5$ (25 m^2) respectively. A larger footprint results in slighter improved performance, as more information about the world is gained at each high altitude measurement compared to a smaller footprint, thus allowing the MCTS planner to better plan



(a) HS % SL species (b) DS & HD species

Fig. 6: Comparison of SASS to boustrophdeon for example simulated worlds. Mean $\mu \pm \sigma_N$ percentage of targets observed for $N = 25$ missions. Mission time is 3000s with $\alpha = 0.15$.



(a) HS & SL species (b) DS & HD species

Fig. 7: Results of different sensor footprints for example simulated worlds. Mean $\mu \pm \sigma_N$ percentage of targets observed for $N = 25$ missions. Mission time is 3000s with $\alpha = 0.15$ and 0.08 for $r_H = 3$ and $r_H = 5$ respectively.

trajectories toward high value regions.

B. Performance on Real Reef Orthomosaic Environments

Fig. 8c shows example trajectories from simulations of SASS over the *Tektite* and *Booby Rock* dataset. From the substrate maps it can be seen that the majority of the targets are concentrated in the green and purple substrate types. The SASS algorithm is able to utilize the high altitude substrate observations to concentrate slower low altitude exploration in these higher value substrates, and remain at a high altitude when moving over sandy areas that contain much fewer coral. Fig. 8d shows the performance of both SASS and the boustrophedon planner averaged across 25 missions for horizon budget of 50s. Full coverage of the reef at low altitude takes approximately an hour for *Tektite*, and we can see that as in the synthetic cases, SASS is able to outperform the baseline up to a mission time of 2500 s, and more rapidly acquires reward than the baseline boustrophedon for the first half of the mission. This validates the applicability of the SASS algorithm to a real coral environment, where substrates are much patchier than in the synthetic cases, and coral is very sparsely distributed.

We see comparative performance between SASS and boustrophedon for the *Booby Rock* dataset, with boustrophedon eventually outperforming SASS in the later part of the mission. While the Tektite dataset has coral and sand substrates distributed across the world, the *Booby Rock* dataset is more bimodal, with the majority of the coral substrates, concentrated in the left half of the dataset. This means that any robot initialised in the right half of the reef will initially spend significant time over less valuable areas while it learns the substrate distributions. Additionally there is significant mixing of the green and purple substrates in the *Booby Rock* dataset, making it challenging to accurately fit a GP model to these substrates. We also see a more uniform distribution of targets at *Booby Rock* as compared to *Tektite*. These two datasets demonstrate a good use case for the SASS algorithm - patchy reefs with clustered coral targets.

VII. Conclusions and Future Work

In this paper we formalize the sparse search-and-sample problem and present a new method for searching for sparsely distributed discrete phenomena across an environment. Our approach mixes low-altitude observations resolving the target phenomena, with high altitude observations of terrain substrates to enable more efficient reward gathering. We demonstrated the SASS algorithm on a variety of synthetic reef environments, where it observed more unique targets than the baseline methods when full coverage of the world is not possible. Performance was further validated on real-world coral datasets, and SASS observed to outperform the canonical coverage planner for patchy, clustered reefs.

A natural extension to this work is to implement SASS on hardware, such as CUREE, and demonstrate its efficacy in the field. A limitation of SASS is the assumption of uniform coral density within each substrate while ignoring in-substrate clustering of targets. Future work will aim to incorporate this spatially dependent local scale variation into the correlation model. Additionally, the assumption of *a priori* knowledge of high value substrates should be relaxed by incorporating online learning methods that enable the robot to learn the value of substrates as it explores and observes target species. Sparsely distributed phenomena are not unique to the ocean environment. SASS can easily be extended to terrestrial environments, e.g. mapping of animal species across different habitat types, or planetary robotics where the target-substrate correlation model could be applied to recent research showing correlation between subsurface biosignatures and surface geologies [30].

References

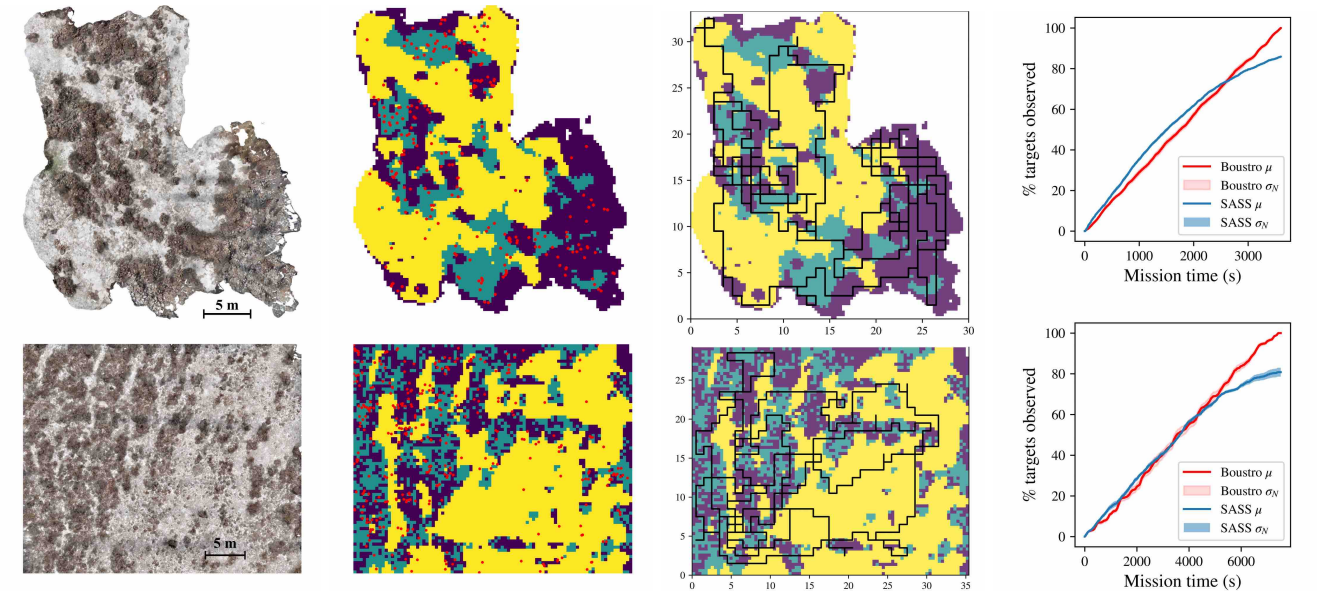
[1] G. Hitz, A. Gotovos, F. Pomerleau, M.-E. Garneau, C. Pradalier, A. Krause, and R. Y. Siegwart, "Fully autonomous focused exploration for robotic environmental monitoring," in *2014 IEEE Intl. Conf. on Robotics & Automation (ICRA)*, May 2014, pp. 2658–2664.

[2] J. C. Mason, A. Branch, G. Xu, M. V. Jakuba, C. R. German, S. Chien, A. D. Bowen, K. P. Hand, and J. S. Seewald, "Evaluation of AUV Search Strategies for the Localization of Hydrothermal Venting," in *30th International Conference on Automated Planning and Scheduling (ICAPS 2020)*, Oct. 2020.

[3] T. O. Fossum, G. M. Fragoso, E. J. Davies, J. E. Ullgren, R. Mendes, G. Johnsen, I. Ellingsen, J. Eidsvik, M. Ludvigsen, and K. Rajan, "Toward adaptive robotic sampling of phytoplankton in the coastal ocean," *Sci. Robot.*, vol. 4, no. 27, p. eaav3041, Feb. 2019.

[4] S. McCammon, G. Marcon dos Santos, M. Frantz, T. Welch, G. Best, R. K. Shearman, J. D. Nash, J. A. Barth, J. A. Adams, and G. A. Hollinger, "Ocean front detection and tracking using a team of heterogeneous marine vehicles," *J. Field Robot.*, vol. 38, no. 6, pp. 854–881, 2021.

[5] 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)*, June 2016, pp. 448–455.



(a) Photographic orthomosaics (b) Substrate and target map (c) Example SASS trajectory (d) Performance of SASS

Fig. 8: Real world results for Tektite *(top)* and Booby Rock *(bottom)* reef environments on St John, USVI. *(a)* shows the orthomosaics generated from underwater imagery collected by CUREE [27]. *(b)* shows the ROST-generated substrate maps with target coral *Porides Astreoides* indicated by black dots. Tektite contains 242 target coral and Booby Rock contains 337 target coral. In both ROST models, *yellow* roughly corresponds to sandy patches, and *green* and *purple* corresponds to coral structures. *(c)* shows examples of simulated trajectories using SASS, with $\alpha = 0.12$. *(d)* shows the performance of SASS compared to boustrophedon planner, based on 25 simulated missions with $\alpha = 0.12$.

[6] 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 Robot. Autom. Lett.*, vol. 4, no. 4, pp. 3782–3789, Oct. 2019.

[7] J. Das, F. Py, J. Harvey, J. Ryan, A. Gellene, R. Graham, D. Caron, K. Rajan, and G. Sukhatme, "Data-driven robotic sampling for marine ecosystem monitoring," *Int. J. Robot. Res.*, vol. 34, no. 12, pp. 1435–1452, Aug. 2015.

[8] L. Alvarez-Filip, F. González-Barrios, E. Pérez-Cervantes, A. Molina-Hernandez, and N. Estrada-Saldívar, "An emerging coral disease outbreak decimated Caribbean coral populations and reshaped reef functionality," In Review," preprint, Nov. 2021.

[9] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robot. Auton. Syst.*, vol. 61, no. 12, pp. 1258–1276, 2013.

[10] W. J. Wright, K. M. Irvine, T. J. Rodhouse, and A. R. Litt, "Spatial Gaussian processes improve multi-species occupancy models when range boundaries are uncertain and nonoverlapping," *Ecol. Evol.*, vol. 11, no. 13, pp. 8516–8527, 2021.

[11] N. Golding and B. V. Purse, "Fast and flexible Bayesian species distribution modelling using Gaussian processes," *Methods Ecol. Evol*, vol. 7, no. 5, pp. 598–608, 2016.

[12] S. Manjanna and G. Dudek, "Data-driven selective sampling for marine vehicles using multi-scale paths," in *2017 IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, Sept. 2017, pp. 6111–6117.

[13] K.-C. Ma, L. Liu, H. K. Heidarsson, and G. S. Sukhatme, "Data-driven learning and planning for environmental sampling," *J. Field. Robot.*, vol. 35, no. 5, pp. 643–661, 2018.

[14] P. Stankiewicz, Y. T. Tan, and M. Kobilarov, "Adaptive sampling with an autonomous underwater vehicle in static marine environments," *J. Field. Robot.*, vol. 38, no. 4, pp. 572–597, 2021.

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

[16] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, ser. Adaptive Computation and Machine Learning series, F. Bach, Ed. Cambridge, MA, USA: MIT Press, Nov. 2005.

[17] A. Arora, R. Fitch, and S. Sukkarieh, "An approach to autonomous science by modeling geological knowledge in a Bayesian framework," in *2017 IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, Sept. 2017, pp. 3803–3810.

[18] C. M. Eaton, E. K. Chong, and A. A. Maciejewski, "Robust UAV path planning using POMDP with limited FOV sensor," in *2017 IEEE Conference on Control Technology and Applications (CCTA)*, Aug. 2017, pp. 1530–1535.

[19] M. Popović, T. Vidal-Calleja, G. Hitz, I. Sa, R. Siegwart, and J. Nieto, "Multiresolution mapping and informative path planning for UAV-based terrain monitoring," in *2017 IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, Sept. 2017, pp. 1382–1388.

[20] F. Stache, J. Westheider, F. Magistri, C. Stachniss, and M. Popović, "Adaptive path planning for UAVs for multi-resolution semantic segmentation," *Robotics Auton. Syst.*, vol. 159, p. 104288, Jan. 2023.

[21] S. Russell and P. Norvig, *Artificial intelligence: a modern approach*. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

[22] O. Madani, S. Hanks, and A. Condon, "On the undecidability of probabilistic planning and infinite-horizon partially observable Markov decision problems," in *Proceedings of National Conference on Artificial Intelligence*, 1999, pp. 541–548.

[23] C. H. Papadimitriou and J. N. Tsitsiklis, "The Complexity of Markov Decision Processes," *Math. Oper. Res*, vol. 12, no. 3, pp. 441–450, 1987.

[24] C. B. Browne et al., "A survey of monte carlo tree search methods," *IEEE Transactions on Computational Intelligence and AI in games*, vol. 4, no. 1, pp. 1–43, 2012.

[25] L. Kocsis and C. Szepesvári, "Bandit based monte-carlo planning," in *Machine Learning: ECML 2006*, J. Fürnkranz, T. Scheffer, and M. Spiliopoulou, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 282–293.

[26] J. Moller and R. P. Waagepetersen, *Statistical Inference and Simulation for Spatial Point Processes*. New York: Chapman & Hall/CRC, 2003.

[27] 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 Intl. Conf. on Robotics & Automation (ICRA)*, May 2023, pp. 11 411–11 417.

[28] S. Jamieson, J. P. How, and Y. Girdhar, "DeepSeeColor: Realtime Adaptive Color Correction for Autonomous Underwater Vehicles via Deep Learning Methods," in *2023 IEEE Intl. Conf. on Robotics & Automation (ICRA)*, Mar. 2023.

[29] Y. Girdhar, P. Giguère, and G. Dudek, "Autonomous adaptive exploration using realtime online spatiotemporal topic modeling," *Int. J. Robot. Res.*, vol. 33, no. 4, pp. 645–657, Apr. 2014.

[30] K. Warren-Rhodes et al., "Orbit-to-ground framework to decode and predict biosignature patterns in terrestrial analogues," *Nature Astronomy*, vol. 7, no. 4, pp. 406–422, Apr. 2023.