Online Multi-Modal Learning and Adaptive Informative Trajectory Planning for Autonomous Exploration

Akash Arora, P. Michael Furlong, Robert Fitch, Terry Fong, Salah Sukkarieh and Richard Elphic

Abstract In robotic information gathering missions, scientists are typically interested in understanding variables which require proxy measurements from specialized sensor suites to estimate. However, energy and time constraints limit how often these sensors can be used in a mission. Robots are also equipped with cheaper to use navigation sensors such as cameras. In this paper, we explore a challenging planning problem in which a robot is required to learn about a scientific variable of interest in an initially unknown environment by planning informative paths and deciding when and where to use its sensors. To tackle this we present two innovations: a Bayesian generative model framework to automatically learn correlations between expensive science sensors and cheaper to use navigation sensors online, and a sampling based approach to plan for multiple sensors while handling long horizons and budget constraints. Our approach does not grow in complexity with data and is anytime making it highly applicable to field robotics. We tested our approach extensively in simulation and validated it with real data collected during the 2014 Mojave Volatiles Prospector Mission. Our planning algorithm performs statistically significantly better than myopic approaches and at least as well as a coverage-based algorithm in an initially unknown environment while having added advantages of being able to exploit prior knowledge and handle other intricacies of the real world without further algorithmic modifications.

A. Arora, S. Sukkarieh
Australian Centre for Field Robotics, The University of Sydney, NSW, Australia,
e-mail: {a.arora, salah}@acfr.usyd.edu.au

P. M. Furlong, T. Fong, R. Elphic
NASA Ames Research Centre, Moffet Field, California, USA
e-mail: {padraig.m.furlong, terry.fong, richard.c.elphic}@nasa.gov

R.Fitch
Centre for Autonomous Systems, University of Technology Sydney, NSW, Australia
e-mail: robert.fitch@uts.edu.au

Fig. 1: KRex2 in the Mojave Desert. KRex carried a Neutron Spectrometer System(NSS) and a Near Infrared Visible Light Reflectance Spectrometer (NIRVIS). The robot autonomously drove and localized itself at the command of remote scientists. The data that it collected was georegistered and presented to the backroom of a team of 30 scientists who adapted their plans in response to data updates.

1 Introduction

In robotic information gathering missions, scientists are often interested in variables or phenomenon which cannot directly be measured but must be observed through correlated proxy measurements. Examples include mapping water abundance in remote environments by measuring neutron flux[1], inferring the health of aquatic life by monitoring chemical concentrations [8], and searching for evidence of life on Mars through biomarkers [16]. These proxy measurements often requires specialized ‘science’ sensor suites such as spectrometers, subsurface drills and sample processing equipment. These are typically either energetically expensive to use, require the robot to remain stationary or have finite capacity limiting how often they can be used given energy constraints and short life spans of many robotic missions.

Robots are also equipped with sensors that are inexpensive in time and energy, such as navigation sensors like cameras or LIDAR. Learning relationships between underlying scientific phenomena of interest and the different inexpensive sensors on-board will allow scientists to better understand phenomena without incurring the prohibitive cost of exhaustively sampling large environments with specialized sensors. Given the locations to be explored are often remote and mostly unknown, this relationship should be learned or updated *in situ*. Robots that can predict latent science variables at a reduced cost will be able to plan paths and sensor usage more effectively which increases science return, mission productivity and allows the robot to operate at higher levels of autonomy.

In this paper we formulate a sensor planning problem in which a robot equipped with multiple sensors has to learn about a latent scientific variable. The robot must plan paths on a graph representation of the environment and decide when and where to use each sensor, constrained by a sensing budget and a goal position. Sensor correlations are modeled by a Bayesian network(BN) generative model, the parameters of which are learned online as observations are made. Reasoning about the network to plan informative sensing sequences is, however, a challenging optimization problem. We calculate approximate solutions by applying Monte Carlo Tree Search (MCTS) techniques [5]. The combination of BNs and MCTS allows the robot to learn and update sensor correlations recursively in a manner which is constant in the number of samples collected and plan informative sensing sequences in an anytime manner. These two properties make our approach highly applicable for online use in robots with limited computational capabilities.

We apply our general approach to a scenario modeled on the Mojave Volatiles Prospector (MVP) project, conducted by NASA Ames Research Center in the Mojave Desert in 2014 [11]. The purpose of the MVP project was to test high tempo remote operations while attempting to estimate abundance of subsurface water. KRex2, the robot used in MVP and pictured in Fig. 1, was equipped with several sensors including a downwards facing camera and a Neutron Spectrometer (NSS) which produces measurements that can be correlated with the abundance of subsurface water. The NSS has a small field of view and measurement requires the robot to drive slowly to avoid spatial blurring of readings. NSS is an inexpensive sensor to use, but we use it as a stand-in for more involved subsurface sampling operations.

At the end of the MVP project, the sensor data was analyzed and it was determined that there was a relationship between the visual properties of terrain and the corresponding NSS readings [9]. If this relationship was learned automatically during the mission, scientists could have made more informed decisions regarding where to direct the robot and deploy sensors to maximize understanding of subsurface water distribution. The MVP project was a precursor to the planned Resource Prospector project which aims to deploy a robot with a similar sensor suite on the moon and map the abundance of surface volatiles [1, 11]. Learning sensor correlations online will make the science return of the RP mission much greater, a significant boon given the project is limited to one lunar day of operations.

We illustrate our key ideas both in simulation and with real data acquired from the MVP project. There the robot deduces the water abundance in an environment by autonomously planning paths, sensor placements and simultaneously learning the relationship between visual properties of terrain and NSS readings. We demonstrate that our approach is statistically significantly better than myopic approaches and comparable to the non-adaptive coverage based planners in initially unknown environments, a result consistent with [13]. Our approach has the added benefit of being generalizable to an arbitrary number of sensors and able to exploit prior knowledge when it's available without further algorithmic modifications.

2 Related Work

Robotic exploration and sensor planning to gain information about the world is an informative path planning problem. Greedy approaches are effective and offer performance guarantees when the problem is submodular [15]. Unfortunately, this property often broken with path dependent rewards often present in field environments. Branch and bound techniques which prune suboptimal branches early in the tree search have shown promise [3, 12] but efficiently calculating tight bounds in problems with unknown environments and multiple sensors becomes non-trivial. There are also various heuristic approaches but they either do not generalize to unknown environments or cannot plan for multiple sensors without significant algorithmic modifications.

In field applications of information gathering, several approaches have been proposed. Thompson *et al.* used a greedy algorithm to design maximally informative trajectories for constructing spatial maps of multi-spectral data [18]. Wettergreen *et al.* extended this in [19] to design trajectories that explore regions of orbital maps that cannot be explained with previous observations – actively solving the spectral unmixing problem. Girdhar *et al.* [10] used a database of observations to detect anomalous data. Similar to our approach, a generative model was learned online by directing the robot towards these anomalies. However these approaches used very short planning horizons and do not make decisions about using expensive secondary sensors to gain information.

Tabib *et al.* [17] explored a search and rescue application where their robot plans trajectories that maximize the information gained by two different sensors which measure the geometry and temperature of the environment. It is assumed that the instruments are constantly collecting data, instead of actively switched on which simplifies the planning problem. Furthermore, it is assumed that the two sensors are conditionally independent while in our problem, being able to learn and exploit relationships between sensing modalities is fundamental.

Arora et al. used a Bayesian network to model relationships between sensing modalities and the phenomena they are trying to measure [2]. The work assumes the relationship between sensors is known *a priori* while in this paper we learn this relationship online. Furthermore, the work uses a greedy planner while here we explore long horizon planning and incorporate goal constraints.

Das *et al.* [7] builds a map of underwater plankton abundance by planning the deployment of a low cost sensor which measures environmental parameters and an expensive ‘plankton’ sensor. To achieve this, two Gaussian Processes (GPs) are used. The first maps spatial co-ordinates to environmental parameters while the second maps environmental parameters to plankton abundance. The robot samples from locations with high plankton uncertainty, where the uncertainty is propagated through both GPs via the Unscented Transform. However the computational complexity of GPs grows with the number of collected samples. Using this framework in online adaptive planning applications like ours is not amenable to long-term operation with the limited computing resources in field robots.

3 Problem Setup

Like MVP we consider a ground vehicle exploring an open environment searching for subsurface water abundance. The operating environment is discretized into a grid where the robot is required to estimate the abundance of water, W, in each grid cell, n. While the robot can be equipped with an arbitrary number of sensors, for ease of illustration, we consider the case with two sensors: a camera which can be used to classify terrain in a cell and a neutron spectrometer (NSS) which returns counts that are positively correlated with water abundance.

The robot plans action sequences, $a_{1:L}$, to maximize the expected information gained, EI, on the water distribution in each cell. The camera always takes measurements but the robot must actively decide when to use the NSS. The robot must also reach a goal position, x_{goal}, before it exhausts the operating (motion and sensing) budget of the mission, B. The optimization objective is:

$$\begin{aligned} &a^*_{1:L} = \underset{a_{1:L}\in A}{\arg\max}\, EI(a_{1:L}) \\ &\textbf{s.t. } cost(a_{1:L}) \leq B \\ &\textbf{s.t. } x_{end}(x_{start}, a_{1:L}) = x_{goal} \end{aligned} \tag{1}$$

A is the action space of the robot which contains the movements the robot can take in the next time step and the decision of whether on not to use the NSS. We define the action space as the four cardinal directions but any motion models can be used here. Similarly, any general cost function can be used and we define ours in Sec. 5. The expected information gain is given by Equation 2 where $H(\cdot)$ is the Shannon entropy, W_n is the water abundance in a cell n and N is the total number of cells in the environment.

$$EI(a_{1:L}) = \sum_{n=1}^{N} [H(W_n) - H(W_n|a_{1:L})] \tag{2}$$

Each action produces some stochastic observation Z_s which reveals information about the water distribution, where $s \in \{Image, NSS\}$. The expected information gain of a sensing sequence is computed over all possible observations that can result from each sensing action in the sequence:

$$EI(a_{1:L}) = \sum_{n=1}^{N} \left[H(W_n) - \sum_{Z_{1:L}} H(W_n|Z_{1:L}) P(Z_{1:L}|a_{1:L}) \right] \tag{3}$$

$P(Z_{1:L}|a_{1:L})$ is the sensor noise model while the $H(W_n|Z_{1:L})$ term is a function of the robot's belief on the environment and the sensor correlations.

4 Approach

The overall proposed architecture is shown in Figure 2. Instead of specifying paths and sensing waypoints directly, our framework allows scientists to simply provide a goal position constraint, sensing budget and the variable they are interested in learning about- a useful capability in remote environments with communication constraints. We now describe the two main components of our approach: a generative model for learning sensor correlations online and an anytime, approximate path planner (MCTS) to find approximate solutions to Equation 1.

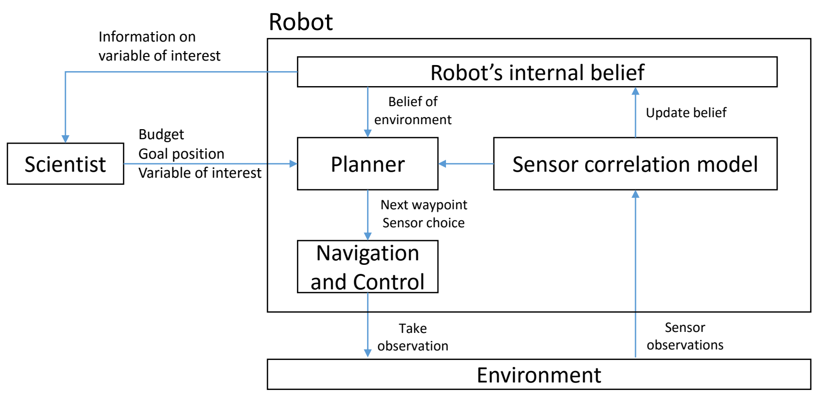

Fig. 2: The overall systems architecture for our approach

4.1 *Modeling and Learning Sensor Correlations*

Loosely inspired from topic modeling literature [4], we structure the dependencies between the NSS observations and the camera with the generative model shown in Fig. 3. The NSS observes the water distribution W in a cell n through observations Z_{NSS}. The camera observation is denoted by Z_I while T is the class of terrain. A conditional probabilistic relationship between terrain and water classes is parametrized by θ and hyperparameters α which are learned during the mission as data is collected. We assume all nodes are discrete variables but the observation nodes can directly handle continuous data as well. The probabilistic mapping from T and W nodes to their corresponding observation nodes is deduced from the sensor/classifier model the robot is using. Unsupervised dimensionality reduction techniques can also be applied here.

In this section we derive the Bayesian update for the beliefs of nodes W_n, T_n and θ as observations are made in a cell. We define:

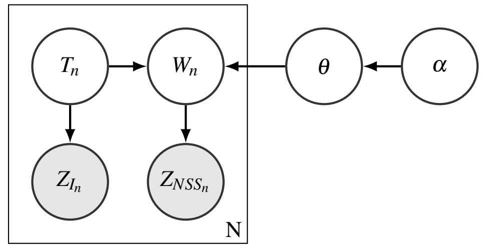

Fig. 3 Bayesian generative sensor correlation model. The model can be extended to an arbitrary number of sensors and adapted to incorporate sensor dependencies that come with specific applications

$$P(W|T=t) \sim Categorical(\theta_t) \tag{4}$$

$$\theta_t \sim Dirichlet(\alpha_{\mathbf{t}}) \tag{5}$$

$$\theta = [\theta_1, \theta_2..\theta_T] \tag{6}$$

By applying Bayes Theorem and exploiting conditional dependencies in the Bayesian network, the beliefs on the water abundance and terrain types can be updated using Equation 7 where η is the normalization constant. For compactness, we drop the subscript n from the terms W_n and T_n.

$$\begin{aligned} P(W|Z_I, Z_{NSS}) &= \eta P(Z_{NSS}|W)P(W|Z_I) \\ &= \eta P(Z_{NSS}|W)\sum_T P(T|Z_I)P(W|Z_I,T) \\ &= \eta P(Z_{NSS}|W)\sum_T P(T)P(Z_I|T)P(W|Z_I,T) \\ &= \eta P(Z_{NSS}|W)\sum_T P(T)P(Z_I|T)\int_\theta P(W|T,\theta)P(\theta)d\theta \\ &= \eta P(Z_{NSS}|W)\sum_T P(T)P(Z_I|T)\int_\theta \theta P(\theta)d\theta \\ &= \eta P(Z_{NSS}|W)\sum_T P(T)P(Z_I|T)\mathbb{E}(\theta) \end{aligned} \tag{7}$$

Similarly, we can iteratively update belief on terrain by evaluating:

$$P(T|Z_I, Z_{NSS}) = \eta P(T)P(Z_I|T)\sum_W P(Z_{NSS}|W)\mathbb{E}(\theta) \tag{8}$$

Since θ is modeled by a Dirichlet distribution, $\mathbb{E}(\theta)$ can be efficiently calculated by normalizing the corresponding hyperparameters. We can update θ using Equation 9. For compactness we define the full observation vector $Z = [Z_I, Z_{NSS}]$.

$$\begin{aligned} P(\theta|\alpha, Z) &= \sum_{T,W} P(\theta|\alpha_{init}, Z, T, W)P(T,W|Z) \\ &= \sum_{T,W} P(\theta|\alpha_{init}, T, W)P(T,W|Z) \end{aligned} \tag{9}$$

Since $P(\theta|\alpha, Z)$ is also a Dirichlet distribution (conjugate prior) we can calculate the posterior by updating the hyperparameters $\alpha_{w,t} = \alpha_{w,t} + P(W = w, T = t|Z)$ for all values of W and T, where $w \in \{1,\ldots,|W|\}$ and $t \in \{1,\ldots,|T|\}$. When $|W|$ and $|T|$ become large, Gibbs sampling approaches in topic modeling literature can be used to approximate this update [10]. When a terrain cell is observed we also update the terrain beliefs in neighboring cells using a Gaussian kernel.

4.2 Planning

Given the generative BN model, the robot needs to plan paths in an initially unknown environment and decide when to use the NSS to maximize the information gained on the water distribution in the map cells, described in Algorithm 1. As per the optimization objective in Equation 1, the planned paths and sensing sequences must meet mission budget constraints and arrive at the goal location, x_{goal}.

Solving Equation 1 for large environments, long mission durations and large observations spaces quickly becomes intractable, especially for field robots with limited computational resources. Therefore we explore approximate online planning approaches where in each time step the robot executes the first action in the calculated plan and adaptively updates plans as new observations are taken. To tackle this sequential decision making problem, we employ the MCTS planning algorithm- a best first, anytime algorithm popular in game playing literature, which like our problem requires reasoning about both long horizons and stochasticity [5].

We formulate the MCTS such that each node in the tree is a potential movement or sensing action that can be made. It is a tuple consisting of the robot's x and y position, a binary variable indicating whether the NSS was used and the remaining sensing budget. MCTS then iteratively builds a tree by selecting leaf nodes to expand using a tree policy, estimating terminal rewards associated with the leaf by conducting simulations or 'rollouts' in the decision space and back-propagating the reward up the tree. The process is repeated until some computational budget is reached, at which point the root child with the highest average reward is selected as the action to be executed.

We use the Upper Confidence Tree policy to select which leaf nodes to expand, which is a popular approach known to produce good results [14]. For the simulation phase, a random action selection policy is used from the leaf node to the goal position. In this problem instance, the reward of the policy rollout is the expected information gained on water distribution across the map after the policy has been executed, where information gain is defined by Equation 2. Exact computation of the reward involves averaging over all possible observations that can result from the rollout sequence which quickly becomes intractable. We approximate information gain by sampling observations from the robot's belief of the map and simulating a belief update. As number of iterations increases, the MCTS converges to the optimal sensing action sequence. This formulation gives us a principled approach to

Algorithm 1 Our algorithm uses MCTS for the $planner(\cdot)$, which is executed after every action.

```
Input: SensingBudget S, BeliefSpace Bel, RemainingBudget R, GoalPosition x_goal
function MAIN
    R ← S
    while R > 0 do
        robotPose ← getLocalisation()
        a_opt ← planner(robotPose, R, Bel, x_goal)
        Z ← takeObservation(a_opt)
        Bel ← updateBeliefSpace(Z, Bel)
        R ← R − cost(a_opt)
```

Algorithm 2 MCTS Algorithm

```
function PLANNER(robotPose, R, Bel, x_goal)
    T ← initialiseTree(robotPose, R)
    currentNode ← T.rootNode
    while within computational budget do
        currentNode ← treePolicy(T)
        simSeq ← defaultPolicy(currentNode, R)
        reward ← getReward(simSeq, Bel)
        T ← updateTree(T, reward)
    return bestChild(T)
```

incorporate multiple sensors in planning and simultaneously handle long horizons and uncertainty in an anytime manner.

5 Analysis

As mentioned in Sec 2 there are several algorithms in literature for informative path planning [3, 12]. However, these approaches are not suitable for tackling situations in which the robot has to simultaneously decide **when** to activate secondary sensors in addition to planning informative paths which adhere to budget and goal constraints in initially unknown environments. We therefore compare the performance of our approach with the following three baseline algorithms:

Random: At each time step the robot determines the set of actions it can execute in the next step without breaking the goal position and sensing budget constraint. A random action is chosen out of this set. The random policy serves as a baseline for algorithm performance.

Greedy: At each time step, out of the reachable action set, the robot selects the action with the highest expected information gain of the water abundance to sensing cost ratio. This is given by:

$$a^*_{next} = \arg\max_{a \in A} \frac{\sum_z I(z)P(z|a)}{cost(a)} \tag{10}$$

Greedy algorithms are popular in similar field applications [10, 18] due to their simplicity and depending on the problem, submodularity.

Lawnmower: We use a 'lawnmower' pattern to get uniform coverage of the environment. Here we arbitrarily allocate 50% of the sensing budget to the path and 50% to using the NSS. A lawnmower-like path which adheres to the initial and final positions and the budget is designed manually and the NSS is used at uniform intervals along the path.

Our approach, **MCTS-50** (50 iterations were used for MCTS) was evaluated against the baseline algorithms on 50 randomly generated 20 by 20 voronoi maps with fixed start and goal positions. Terrain, water, and the observation nodes were categorical variables with three classes. The true correlation between terrain type and water class (initially unknown to the robot) was set to be 0.85. I.e. given the terrain class, the water class could be predicted with 85% accuracy. Sensor noise for the terrain was set to be 10% while the NSS had 5%. All unobserved nodes were given an uniform prior and the α hyperparameters were initialized to a value of 1. The cost of movement was 1 unit per cell while the NSS required 5 units. Two performance metrics were used: information gain and the average posterior probability of the correct class of water in the cells which we call the recognition score.

Mean and standard deviation is reported and statistical significance is shown with the paired t-test p-value and the effect size using Cohen's d. Negative values of d indicate that the performance of the proposed algorithm is greater than the compared algorithm. The magnitude of d gives the size of the effect, with $d > 0.2$, $d > 0.5$ and $d > 0.8$ being thresholds for small, medium and large effects respectively.

The results are shown in Tables 1 and 2. In terms of average information gain, we statistically significantly outperform random and greedy policies with notable effect sizes (bolded). For the recognition score, the performance improvement is less pronounced. This is because the robot only observes a small proportion of the map and the unseen areas dominate the score.

The performance of the lawnmower is comparable to MCTS in these simulated experiments. In completely unknown and open environments, paths which provide good spatial coverage of the environment are indeed a logical and effective way to gain information. In more realistic environments with obstacles, planning lawnmower paths becomes more complicated. When environmental obstacles are known *a priori* Choset's approach can be applied [6]. In unknown or partially known environments, however, additional replanning would need to occur as obstacles are discovered, something our approach already does. Further, adapting the lawnmower approach to an arbitrary number of sensors would require a way to split the sensing budget across the different sensing modalities, which the MCTS optimizes automatically in a principled manner. While the 50-50 budget split between paths and NSS produced good results in the simulation setting, there is no guarantee that performance will continue to be competitive in longer missions and large environments.

In robotic missions, there is usually some prior knowledge available such as orbital maps or scientific beliefs on what the robot is likely to see. A key advantage of our approach is that we can easily encode this knowledge in the form of Bayesian priors. Orbital maps can be encoded by biasing the prior distribution of terrain types

Table 1: Information gain for the different algorithms and their performance relative to MCTS-50

Budget	Greedy				Random				Lawnmower				MCTS-50	
	μ	σ	p	d	μ	σ	p	d	μ	σ	p	d	μ	σ
60	20.7	8.93	**0.05**	**-0.34**	15.6	7.16	**1e-5**	**-0.95**	22.4	8.99	0.40	-0.17	24.0	10.23
80	28.4	11.76	**0.003**	**-0.60**	20.6	10.92	**9e-8**	**-1.20**	31.7	12.62	**0.03**	**-0.35**	36.7	15.54
100	32.3	12.81	**0.004**	**-0.57**	27.1	13.42	**1e-4**	**-0.88**	39.4	14.50	0.52	-0.12	41.4	18.81
120	39.3	13.99	**0.003**	**-0.62**	29.0	12.93	**2e-8**	**-1.31**	46.6	19.24	0.39	-0.14	49.1	17.37
140	43.4	13.12	**3e-5**	**-0.84**	33.3	16.58	**5e-9**	**-1.34**	54.8	21.4	0.62	-0.10	56.8	18.50

Table 2: The average posterior probability of the true water distribution given the maps learned by the different algorithms, larger values are better.

Budget	Greedy				Random				Lawnmower				MCTS-50	
	μ	σ	p	d	μ	σ	p	d	μ	σ	p	d	μ	σ
60	0.37	0.02	0.56	-0.08	0.36	0.02	**0.004**	**-0.43**	0.38	0.02	0.14	0.25	0.38	0.03
80	0.39	0.03	**0.008**	**-0.44**	0.37	0.03	**3e-8**	**-0.94**	0.40	0.03	0.17	-0.19	0.41	0.04
100	0.41	0.03	0.21	-0.19	0.38	0.03	**8e-5**	**-0.81**	0.42	0.03	0.36	0.15	0.41	0.05
120	0.42	0.04	**0.04**	**-0.35**	0.38	0.04	**5e-10**	**-1.37**	0.43	0.03	0.84	-0.03	0.43	0.03
140	0.43	0.04	**0.0008**	**-0.54**	0.39	0.04	**3e-12**	**-1.62**	0.44	0.04	0.15	-0.26	0.45	0.03

while scientific knowledge of known sensor correlations can be incorporated by incrementing the α hyperparameters. Unlike the standard lawnmower, our approach will automatically take advantage of this information without algorithmic modifications. To verify this, we ran 50 trials with a sensing budget of 140 where the robot's belief of the correct terrain type was initialized to 0.5 instead of a uniform distribution. MCTS outperformed the lawnmower with p-values and effect sizes of < 0.001 and ≈ -0.5 respectively for both information gain and recognition scores.

The computational time of MCTS depends on the remaining sensing budget and problem size. For our MATLAB implementation, 50 iterations took between 0 and 5 seconds. With more efficient memory management, optimized implementation and parallelization, significant speed boosts can be achieved which will further boost the performance of MCTS as more iterations can be carried out.

6 Results with Mojave Data

Since much of the data from the Mojave Desert test site was collected in line traverses, we selected 100 pairs of ground camera images and NSS counts from this dataset and redistributed them into a 10 by 10 grid to simulate a field environment. Typical ground camera images are shown in Fig. 4. The images from the MVP dataset are quite noisy with both strong shadows and regions with saturation.

The data now needs to be transformed into a representation that can be fed into the generative model. While any black box classifier model can be used for this,

(a) Pavement Terrain Type (b) Transition Terrain Type (c) Wash Terrain Type

Fig. 4: Different types of terrain in the MVP test area. Pavements were found to be associated with high NSS counts, while washes had low NSS counts. The transition terrain was in between washes and pavements and had moderate NSS counts.

we use a simple example based classifier for illustration. We selected image subsets based on domain knowledge of the terrain classes present and used these to define four cluster centers. Candidate images are then classified based on the closest cluster centre in intensity space. The labels are transformed into soft evidence using a confusion matrix derived from training data. Similarly, k-means clustering with three clusters is used to probabilistically classify NSS counts into water abundance. The probabilistic classifications are fed into the BN as soft evidence. Continuous data can also be directly fed into the proposed generative model as long as the probabilistic mapping from T and W nodes to observations can be determined.

We compare MCTS-50 and the lawnmower algorithms on 20 randomly generated 10 by 10 maps with a sensing budget of 40. We ran two sets of trials with NSS costs of 5 and 2 units. Since the sensing budget of 40 is relatively small, by reducing the cost of NSS, the latter trial artificially increases the planning horizon and intends to show the resulting changes in performance.

The results are shown in Fig. 5. In terms of information gain, MCTS is on average better than lawnmower for this sample and statistically significantly when the NSS cost is 2. There is a larger performance gap compared to the simulations. This is because, doing a 50-50 split in the lawnmower budget allocation is no longer as effective for this map size, sensing budget and sensor model. Like in simulations, we assumed an initially unknown environment and further improvements can be expected with the integration of prior knowledge. In terms of recognition score, MCTS is slightly lower than lawnmower in NSS-5 and similar in NSS-2 but remains statistically indifferent like in simulations.

7 Conclusions and Future Work

Being able to reason about scientific latent variables of interest to plan informative sensing sequences is an important problem in field robotics. We have presented a scalable approach to automatically learn sensor correlations online and a sampling based approach to plan long horizon sensing sequences which is anytime and incor-

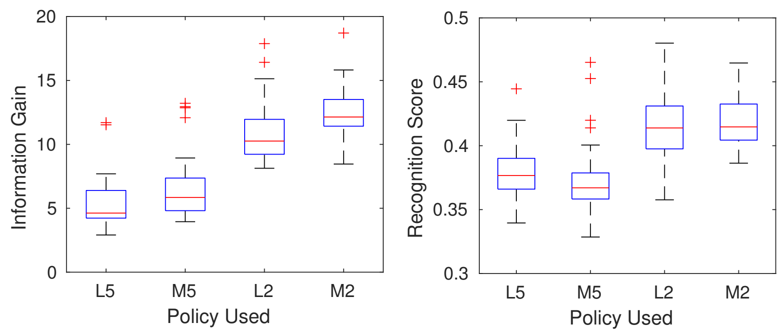

Fig. 5: Comparison of Information Gain and Recognition Scores for Lawnmower and MCTS for different NSS costs

porates budget and goal position constraints. Our simulations and real data experiments show we significantly outperform myopic approaches which are popular in similar applications and compete with maximum coverage paths in unknown environments. Our approach can also exploit prior knowledge when it is available without further algorithmic modifications. In future work we would like to incorporate unsupervised approaches to classification and evaluate our approach on different applications such as remote sensing and agriculture.

Acknowledgements We would like to thank NASA's Resource Prospector project and the Australian Center for Field Robotics for funding this work.

References

[1] Andrews DR, Colaprete A, Quinn J, Chavers D, Picard M (2014) Introducing the resource prospector (RP) mission. In: AIAA SPACE 2014 Conference and Exposition, p 4378

[2] Arora A, Fitch R, Sukkarieh S (2016) Extending autonomy of planetary rovers by encoding geological knowledge in a bayesian framework. In: 10th International Cognitive Robotics Workshop, IEEE International Conference on Intelligent Robotics and Systems, IROS

[3] Binney J, Sukhatme GS (2012) Branch and bound for informative path planning. In: IEEE International Conference on Robotics and Automation (ICRA), IEEE, pp 2147–2154

[4] Blei DM, Ng AY, Jordan MI (2003) Latent dirichlet allocation. Journal of machine Learning research 3(Jan):993–1022

[5] Browne CB, Powley E, Whitehouse D, Lucas SM, Cowling PI, Rohlfshagen P, Tavener S, Perez D, Samothrakis S, Colton S (2012) A survey of monte carlo

tree search methods. IEEE Transactions on Computational Intelligence and AI in games 4(1):1–43
[6] Choset H (2001) Coverage for robotics–a survey of recent results. Annals of mathematics and artificial intelligence 31(1):113–126
[7] Das J, Harvey J, Py F, Vathsangam H, Graham R, Rajan K, Sukhatme GS (2013) Hierarchical probabilistic regression for auv-based adaptive sampling of marine phenomena. In: IEEE International Conference on Robotics and Automation (ICRA), IEEE, pp 5571–5578
[8] Dunbabin M, Marques L (2012) Robots for environmental monitoring: Significant advancements and applications. IEEE Robotics & Automation Magazine 19(1):24–39
[9] Foil G, Fong, Terry DM, Elphic RC, Wettergreen D (2016) Physical process models for improved rover mapping. In: The International Symposium on Artificial Intelligence, Robotics and Automation in Space (iSAIRAS)
[10] Girdhar Y, Dudek G (2016) Modeling curiosity in a mobile robot for long-term autonomous exploration and monitoring. Autonomous Robots 40(7):1267–1278
[11] Heldmann J, Colaprete A, Cook A, Roush T, Deans M, Elphic R, Lim D, Skok J, Button N, Karunatillake S, et al (2015) Mojave volatiles prospector (MVP): Science and operations results from a lunar polar rover analog field campaign. In: Lunar and Planetary Science Conference, vol 46, p 2165
[12] Hollinger GA, Sukhatme GS (2014) Sampling-based robotic information gathering algorithms. The International Journal of Robotics Research 33(9):1271–1287
[13] Hollinger GA, Englot B, Hover FS, Mitra U, Sukhatme GS (2013) Active planning for underwater inspection and the benefit of adaptivity. The International Journal of Robotics Research 32(1):3–18
[14] Kocsis L, Szepesvári C (2006) Bandit based monte-carlo planning. In: Proceedings of European Conference on Machine Learning, pp 282–293
[15] Krause A, Guestrin C (2007) Near-optimal observation selection using submodular functions. In: AAAI, vol 7, pp 1650–1654
[16] Leshin L, Mahaffy P, Webster C, Cabane M, Coll P, Conrad P, Archer P, Atreya S, Brunner A, Buch A, et al (2013) Volatile, isotope, and organic analysis of martian fines with the mars curiosity rover. Science 341(6153):1238,937
[17] Tabib W, Whittaker R, Michael N (2016) Efficient multi-sensor exploration using dependent observations and conditional mutual information. In: IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), IEEE, pp 42–47
[18] Thompson DR, Wettergreen D (2008) Intelligent maps for autonomous kilometer-scale science survey. In: The International Symposium on Artificial Intelligence, Robotics and Automation in Space (iSAIRAS)
[19] Wettergreen D, Foil G, Furlong M, Thompson DR (2014) Science autonomy for rover subsurface exploration of the atacama desert. AI Magazine 35(4):47–60