Intelligent optimization of borehole geophysics planning for critical mineral exploration

Benjamin Barlow

Master of Science
School of Informatics
University of Edinburgh
2024

Abstract

The transition to a carbon-free energy grid and transportation network hinges on accelerating the discovery of the battery metals: copper, nickel, cobalt, and lithium. Mineral explorers infer the presence of subsurface mineral deposits by relying on geophysical data collected above the surface. Airborne geophysical surveys offer expansive coverage and rapid access of vast, remote areas. Aircraft conducting geophysical surveys traditionally fly in straight lines at fixed intervals ranging from 50 m to 2 km. This comprehensive approach permits detailed geophysical maps but accumulates significant flying distances which come at a substantial financial and time cost.

In environmental monitoring, sequential data acquisition techniques that optimize locations for data collection while minimizing resource expenditure have become prevalent. These techniques allow mobile sensors to use observations to form a belief of the world, and in turn, use their current belief of the world to inform the path taken to collect observations in the future. The partially observable Markov decision process (POMDP) framework has demonstrated considerable promise in guiding path planning decisions in robotics and has recently been applied to subsurface applications where noisy observations are used to infer the underlying state. Building on this foundation, we propose a POMDP tailored to optimize the flight paths of fixed-wing aircraft in airborne geophysical surveys. We evaluate our approach using simulated geophysical maps, comparing its performance to traditional grid-based methods in terms of distance flown and resulting profitability.

The results demonstrate that our approach can accurately and confidently estimate the true state of the world in significantly less survey distance. However, its performance is inconsistent across all scenarios and therefore we suggest an alternative model formulation for future work.

Research Ethics Approval

This project was planned in accordance with the Informatics Research Ethics policy. It did not involve any aspects that required approval from the Informatics Research Ethics committee.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Benjamin Barlow*)

Acknowledgements

First, I would like to thank my undergraduate dissertation supervisor, Ioannis Kosmidis, at the University of Warwick. First impressions are crucial, and the collaborative relationship he fostered during our project in 2020 provided me with a remarkable introduction to research, igniting a passion for academic exploration that ultimately led me to seek a master's dissertation project that excited me profoundly.

Next, I wish to thank Nigel Goddard for his guidance during my decision to switch to a part-time master's program. His insights gave me the confidence that this change would not only allow for part-time employment but also deepen my understanding of my academic interests. The free time I gained in the summer of 2023, which would not have been the case had I been a full-time student, allowed me to explore global opportunities for research collaboration. This led to my discovery of KoBold Metals and their partnership with Mineral-X at Stanford University. I am deeply grateful to David Zhen Yin of Mineral-X for responding to my cold email and establishing a relationship that made my project possible. Moreover, David designed a project that perfectly aligned my technical skills in artificial intelligence with my passion for the global clean energy transition. I absolutely loved the project!

My heartfelt thanks go to Robert Moss and Mykel Kochenderfer, whose support transformed the first three weeks of my project. Robert assured me that if I made the effort to visit Stanford, his colleagues at the Stanford Intelligent Systems Laboratory (SISL) would generously share their POMDP expertise in person. This gave me the confidence that traveling across the globe would be well worth it. Upon my arrival, Mykel warmly welcomed me to the SISL family by inviting me to their weekly lab meeting, which helped me quickly integrate with the team and set the stage for a productive and enjoyable stay. Besides the professional gains, I must say, June in California is somewhat superior to June in Edinburgh, so thank you both.

I would like to give special recognition to Dylan Asmar from the SISL family, who was instrumental in guiding me through the world of POMDPs and POMDP solvers. As someone with no prior experience in Julia or POMDPs, I had numerous questions early on, ranging from understanding his workflow in Julia to grasping the complex mathematical foundations of POMDPs and their solvers. Dylan always welcomed my questions and answered them with a smile—a quality I respect massively.

I am grateful to John Mern, a key author of works cited in this dissertation. Despite his demanding role as CEO of an early-stage startup, John, a PhD graduate of SISL, generously took the time to meet with me and discuss my project over coffee. Both

his experience in industrial mineral exploration at KoBold Metals and his advanced POMDP knowledge from his PhD makes him an expert in this field. As a newcomer to both mineral exploration and POMDPs, I valued the opportunity to freely ask him questions extremely highly.

My final thanks to the SISL community go to the authors and maintainers of the POMDP.jl ecosystem. The codebase for this project was built upon your foundational work, and I am truly grateful for your contributions. Furthermore, I extend my gratitude to Jef Caers and, once again, John Mern, for implementing a mineral exploration POMDP in Julia that served as a critical foundation for this project. Without your efforts, this project would not have been possible to complete in 3 months.

I would also like to thank Elliot Fosong, a PhD student at the University of Edinburgh, for generously offering his time to discuss POMDPs and for reassuring me that I had someone in Edinburgh ready to provide supervision if I encountered difficulties with my project's implementation in Julia. Although I quickly became comfortable with Julia, knowing that Elliot was available for support provided me with great peace of mind throughout. Additionally, his suggestions for improving the reward function, particularly through potential-based reward shaping, will be implemented if this work is extended in the future.

Most importantly, I would like to express my deepest gratitude to my supervisors, each of whom have already been acknowledged for their respective contributions beyond direct supervision. Nigel Goddard provided feedback on the structure of my report and took the role of lead supervisor of the project. David Zhen Yin shared his vision for the project in its early stages and kindly guided me in understanding the intricacies of geophysics and mineral exploration. Finally, Robert Moss has been an outstanding supervisor throughout—from in person meetings at Stanford to online discussions after my return to Edinburgh, and even during his travels to conferences. His key suggestion to model my agent as a plane with a continuous flying path significantly enhanced the realism of my project. Robert operated with patience, genuine care, and a supportive tone from start to finish; I hope our professional paths cross again someday. Thank you, Nigel, David, and Robert.

Table of Contents

Chapter 1

Introduction

To permit the global transition to a carbon-free power grid and transportation network, we must rapidly increase the discovery rate of the battery metals: copper, nickel, cobalt, and lithium (Campbell, 2014; Mudd and Jowitt, 2014; Turner, 2022). Extracting ore deposits for metal production is dependent on first understanding geological structures and their relationships (Cox, 2005). Due to mineral explorers traditionally searching for evidence of mineralization above the surface, deposits shallow in the subsurface have progressively depleted in well-explored areas (Davies et al., 2021). Ensuring continued discoveries now requires methods that provide insights into deep subsurface structures lacking surficial evidence. Decisions to mine or abandon regions are ultimately based on drilling campaigns, which validate subsurface composition and, in turn, a prospective ore's grade and volume. Before deciding where to drill boreholes[1], or even whether to drill at all, geophysical data collected above the surface provides preliminary indications of ore presence through subsurface magnetic, resistivity, or density anomalies[2]. Gravity and magnetic methods are particularly popular in mineral exploration since they lend themselves to observation from airborne sensors (Hinze et al., 2013). Traditionally, fixed-wing aircraft fly in predefined straight lines at fixed intervals (see Fig. 2.1) to survey entire areas ranging in scale from individual prospects to continental-sized regions (Hinze et al., 2013). We argue that this conventional, fixed-pattern approach to geophysical data acquisition is not the most efficient strategy—in terms of cost and time—for identifying high-value mineral prospects. We suggest that more adaptive, data-driven methods could significantly ease the financial and time burden of mineral

[1] In mineral exploration, boreholes are holes drilled in the subsurface to extract minerals for further analysis.

[2] For example, a gravity (density) anomaly—a reading that differs from Earth's gravitational field—is caused by lateral variations in rock density in the subsurface.

exploration.

Geophysical data acquisition adheres to a sequential process: sensor readings are recorded in time-ordered sequences with each observation tied to a specific location. The challenge of determining the optimal sequence for collecting data has been studied extensively. These strategies maximize learning opportunities while minimizing resource (time, energy, or financial) expenditure. Applications of these techniques to *environmental sensing*[3] tasks include wind field reconstruction (Yildiz et al., 2023) and ocean monitoring (Zhang et al., 2023). However, there is a notable paucity of studies leveraging sequential data acquisition methods in the field of geophysics.

A partially observable Markov decision process (POMDP; see Section 2.4) is a model formulation for decision making when an agent cannot reliably identify the underlying environment state (Kaelbling et al., 1998). They have proven effective in modelling subsurface *state uncertainty* in applications such as carbon capture and storage (Corso et al., 2022) and groundwater contamination remediation (Wang et al., 2022). *State uncertainty* is applicable in the field of geophysics, where the true state of the subsurface is unknown because anomaly amplitudes decrease with source depth, leading to the challenge that large, deep ore bodies and small, shallow deposits can produce indistinguishable geophysical readings at the surface (Hinze et al., 2013). POMDPs have likewise demonstrated effectiveness in information-driven path planning[4] for both robotics (Lauri et al., 2022) and environmental monitoring (Bai et al., 2021). By integrating the framework's ability to handle subsurface state uncertainty and govern path planning, we define a bespoke POMDP tailored for adapting a fixed-wing aircraft's flight path in real time in response to geophysical observations.

Our central hypothesis is that adaptively planning geophysical survey flight paths using an information-driven approach will reduce the flying distance required and associated costs in mapping geophysical anomalies to inform borehole placement. The primary contribution of this work is to demonstrate to the geophysics community the financial value of applying artificial intelligent techniques, specifically POMDP-based methods, to geophysical data acquisition. Our POMDP-based approach, tested against traditional grid surveys on 150 simulated maps, demonstrates nearly double profitability under fixed survey budgets by significantly reducing the distance flown.

[3]Environmental sensing is a process by which an environmental attribute of interest is collected from different locations so that a continuous map of its levels and variations can be constructed (Bai et al., 2021).

[4]Information-driven path planning is the process by which an agent leverages information perceived from an environment to look ahead and plan actions for the subsequent steps (Bai et al., 2021).

Chapter 2

Background work

2.1 Geophysical data acquisition

As outlined in Chapter 1, discoveries of minerals for metal production are dependent on first understanding geological structures and their relationships. Geophysicists capture the spatial distribution of physical properties of the Earth's subsurface by constructing geophysical maps. Geophysical data, such as magnetic, electrical, electromagnetic, radiometric, and gravity, can be collected on land using hand-held devices, on board ships in offshore exploration, or by conducting airborne surveys with planes and helicopters (Lyatsky, 2010). Following data acquisition in a subset of locations, the entire map can be interpolated through multiple-point geostatistics (Journel and Zhang, 2006; Mariethoz and Caers, 2014) or spatial-covariance based methods (e.g., Gaussian process regression; see App. A). The map is used to decide whether to abandon the region being explored (NO-GO) or to perform a drilling campaign (GO). If the latter, material from the subsurface is extracted—termed "borehole data acquisition"—and the ore's properties, such as grade and volume, are estimated to calculate its monetary value.

Acquiring data for mapping purposes on land presents a variety of challenges: regions of interest can be hundreds of miles from human settlement; northern climates can be covered by snow throughout winter and the potential need for air supply multiplies costs astronomically; and land-use rules can present legal challenges too. Airborne surveys are great solutions since they enable effortless access and rapid coverage. This makes them a popular choice among mineral explorers; by October 2005, a popular airborne gravity system named Falcon had flown 1 million km of surveys (almost entirely for mineral exploration; Dransfield 2007), only six years after their first flight in October 1999 (Dransfield et al., 2001).

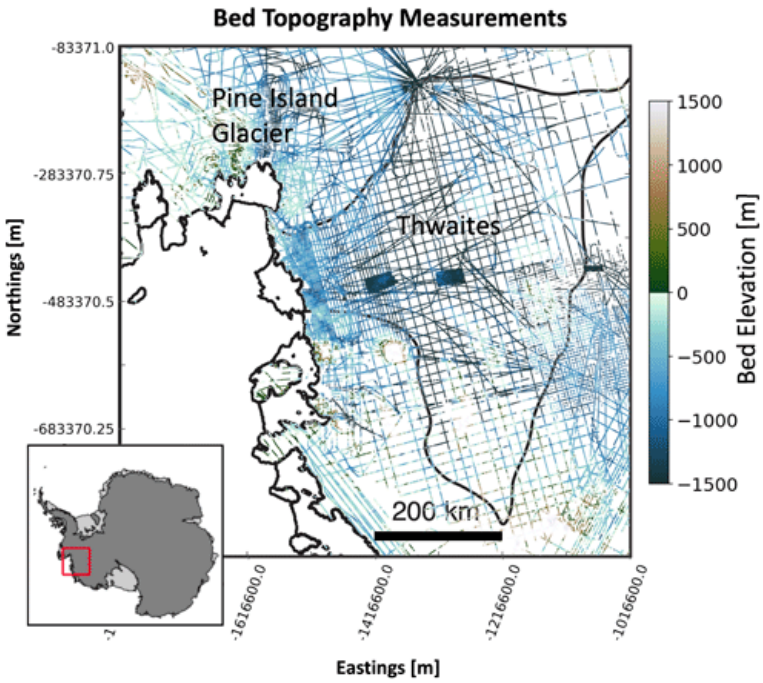


Figure 2.1: Radar survey paths flown over the Thwaites and Pine Island glaciers in West Antarctica. Image credit: Fig. 1, Yin et al. (2022). See Fig. 6 in Oldenburg and Pratt (2007) and Fig. 3 in Kebede and Mammo (2021) for further examples of grid-like flight paths in airborne geophysical surveys.

The airborne survey literature offers advancements in geological modelling (Li and Oldenburg, 2000; Olierook et al., 2020) alongside efforts to enhance data quality through post-processing (Chen and Macnae, 1997; Kass and Li, 2008) and next-generation instrumentation (Tryggvason et al., 2004). However, there has been no attempt to marry path planning advancements in robotics to geophysical survey planning.

Airborne surveys are always conducted by flying in predefined straight lines at fixed intervals (see Fig. 2.1). In fact, Lyatsky (2010) include uniformity of coverage (alongside ease of access and rapid coverage) when listing the advantages of airborne surveys. However, we argue here that uniform coverage is not necessarily desirable. Our work suggests that it's economically naive to continue acquiring data in regions where previously acquired data indicates the probability of mineral presence is extremely low. If the ultimate decision is to make a GO/NO-GO decision, and previously acquired data suffices to quantify uncertainty and make such a decision, it is not wise to continue acquiring more data at a monetary and time cost. We aim to minimize the flying distance (and associated costs) in making such a GO/NO-GO decision by applying sequential data acquisition techniques to optimze data collection.

2.2 Sequential data acquisition

2.2.1 Sensor networks

The issue of sensor networks (Caselton and Zidek, 1984; Krause et al., 2008) pertains to identifying the most efficient positions for data acquisition in the monitoring of spatial phenomena. Problems in this domain may be framed using a predetermined or an adaptable number of sensors. In the fixed case, an objective function is maximized adhering to the constraints on the number of available sensors. In the variable setting, the optimal number of sensors and their locations are determined by balancing the *utility gain* against the cost of deploying and maintaining additional sensors. *Utility gain* can be quantified using techniques such as convex optimization (Joshi and Boyd, 2009), heuristic methods (Jung et al., 2015), or information theory (Pei et al., 2019); the latter being a derivation of uncertainty measures like entropy or mutual information (Trendafilova et al., 2001; Atallah et al., 2010; Hoffmann and Tomlin, 2010).

Global efforts to adapt to climate change consistently present applications with the need to reconstruct entire data fields utilizing a finite number of sensors. An example in environmental modelling is the work in ocean monitoring by (Zhang et al., 2023). They reconstruct global atmosphere and hydrologic data by employing oceanographic buoys and unmanned aerial vehicles (UAVs). In their case, the cost of additional sensors is a function of location, since longer offshore distances correspond to higher deployment and maintenance costs.

An aircraft conducting a geophysical survey can be considered a special case of a sensor network with a single mobile sensor. The intersection of robotics and sensor networks in the form of mobile sensors has been seen in wildfire monitoring (Julian and Kochenderfer, 2020), active volcanoes surveillance (Astuti et al., 2009), and temperature field reconstruction in a small lake (Zhang and Sukhatme, 2007).

2.2.2 Adaptive informative path planning

Whilst static sensors give rise to the question of optimal placement, robotic sensors give rise to the question of optimal path planning. The agent's goal is to maximize utility gain while respecting resource constraints in terms of time or energy. For a comprehensive review of environmental sensing and the topic's intersection with path planning, see Dunbabin and Marques (2012) and Bai et al. (2021), respectively. In general, utility can correspond to uncertainty metrics or another more appropriate choice depending on

the application. Resource constraints depend on the application but usually correspond to time or energy (in our case, we impose financial and time constraints), and in some cases, environmental phenomena too (e.g., Techy et al. (2009) subject UAVs to wind disturbances when monitoring plant pathogen spores). Our aerial dynamics model (see Section 3.2) ignores wind influences.

The literature refers to the above balance of utility gain under resource constraints as the adaptive informative path planning (AIPP) problem (Singh et al., 2009; Lim et al., 2016). Interacting with a *partially observable environment*[1], an agent plans a path that visits a subset of locations that are most "informative" conditioned on all information acquired so far. Addressing the AIPP problem, which is NP-hard (Meliou et al., 2007), has garnered considerable scholarly attention. Singh et al. (2009) and Dunbabin and Marques (2012) extend the problem to the multi-robot case and Ott et al. (2022) plan over multi-modal sensing capabilities through multiple sensor types where more informative sensors come at an additional cost. Both extensions could be relevant in geophysical data acquisition: a geophysical survey could be conducted by a team of planes (multi-robot case) adapting their paths in real-time with multi-modality introduced by combining borehole data acquisition and geophysical data acquisition into a single optimization. However, here we focus on a single agent with a single modality and leave these suggestions as a future consideration.

2.2.3 Myopic and nonmyopic planning

Planning can be defined as finding an optimal way to behave given a complete and correct model of the world dynamics and a reward structure for certain behaviour (Kaelbling et al., 1998). When planning for data acquisition, actions correspond to collecting data at specific locations and times and rewards correspond to *utility gain* examples outlined in Section 2.2. Planning methods that choose actions to maximize the immediate reward according to some metric and disregard the long-term effects of their choices are considered *myopic*. An example of a myopic method is Bayesian optimization (Shahriari et al., 2016). On the other hand, *nonmyopic* methods, for example, Monte Carlo planning and reinforcement learning (Sutton and Barto, 2018), select actions that are believed to facilitate optimal behaviour both in the present and the future. A similarity between both approaches is that they solve for each action only after observing the results of previous actions. In essence, the past influences behaviour

[1] A partially observable environment is one where the agent cannot directly observe the state and must instead make decisions based only on observations that are generated by the state.

in both paradigms, but only nonmyopic methods consider future implications when determining how to behave in the present. In their attempt to optimize borehole placement for mineral exploration, Mern and Caers (2023) discuss the superior performance of nonmyopic methods but acknowledge they typically come at increased computational cost.

2.3 Particle filters

In geophysical data acquisition, *observations* such as magnetic or gravity readings serve as indirect clues to the *true state* (mineral presence in the subsurface), but the true state of the subsurface remains hidden throughout, leading to a setting of *partial observability*. We term this "recovering state variables" from noisy sensor readings; a problem that has received great attention in robotics literature (Barfoot, 2017). A rich class of methods for recovering state variables are Bayes filters (Maybeck, 1982). They are probabilistic in that they don't just guess the state x, they calculate the probability that any state x is correct. While parametric implementations of Bayes filters, such as Kalman filters (Bar-Shalom et al., 2002), are well-established, a more versatile alternative is offered by *particle filters* (Thrun, 2002). They instead estimate state variables through *particles* (samples) in a nonparametric manner. This scheme maintains a probabilistic distribution over the state, thereby accounting for noise and uncertainty seamlessly. Each particle in the filter represents a hypothesis of the true state. The initial particle set is generated by sampling from a prior distribution that encapsulates initial beliefs about the state, though this preliminary representation is typically limited in accuracy. Nevertheless, through a recursive update process, the particle filter progressively refines its estimation as observations are received. The aim is for the particle set to converge to the desired posterior distribution, which is achievable under certain conditions (see. Fig. 3.5 for the approach we adopt to permit convergence). We use particle filters in this study to represent the belief of a POMDP.

2.4 Partially observable Markov decision process

2.4.1 Model definition

Sequential decision-making problems have traditionally been conceptualized and addressed through the Markov decision process (MDP; Kaelbling et al. 1998). Such

methods assume *full observability*; the underlying state of the environment (for example, the configuration of a chess board) is known. In *partially observable* environments, this assumption breaks down. By instead maintaining a probability distribution over possible states, the partially observable Markov decision process (POMDP; Åström 1965) facilitates the resolution of sequential decision-making problems under state uncertainty.

A POMDP is represented by a tuple $(S, A, T, R, O, Z, \gamma)$. Similarly to the well-known MDP tuple (S, A, T, R, γ): S is the set of possible states of the environment; A is the set of possible actions the agent can take within the environment; $T : S \times A \times S \to [0, 1]$ is the transition model[2]; $R : S \times A \times S \to \mathbb{R}$ is the reward function; and $\gamma \in (0, 1]$ is the discount factor applied to the reward at each time step. The POMDP tuple has two additional elements: the observation space O is the set of possible observations and the observation model $Z : O \times A \times S \to [0, 1]$ is a conditional probability distribution over the observation space given a state and action[3].

At each timestep, the agent takes action $a \in A$ to move from state $s \in S$ to state $s' \in S$ with probability defined by the state-transition function $T(s' \mid s, a)$. In the scenario studied in this dissertation, the actions refer to manoeuvring the aircraft and the state represents a geophysical anomaly, the location of the aircraft, and the measurements taken thus far. The optimal action will account for its expected utility gain and its impact on future decisions. Therefore, the intersection of planning with POMDPs is a case of nonmyopic planning (unless $\gamma = 0$ and all future rewards are disregarded). The agent receives an observation $o \in O$ with probability $Z(o \mid a, s')$ and a reward $r = R(s, a, s') \in \mathbb{R}$. The reward evaluates the impact of a given action on the total utility of the action sequence that the agent strives to optimize. The time discount factor γ is used to favour rewards that occur earlier in the process. Observations typically consist of noisy measurements of a subset of the state. In our case, noise arises from other minerals (non-ore minerals) in the subsurface causing minor geophysical anomalies and sensor noise caused by aircraft motion. The action-observation sequence can either continue indefinitely or terminate when a terminal state is reached. Whilst infinite horizon sequential problems exist, our focus is a POMDP that terminates at a variable timestep T, reflecting the GO/NO-GO decision made in mineral exploration.

In practice, a gravimeter mounted to a plane cannot directly observe the subsurface; instead, it gathers observations that offer clues about what lies beneath. In the POMDP

[2]One can also define $T : S \times A \times S \to \mathbb{R}$ to be a density function such that $\int T(s' \mid a, s)\, ds' = 1$.

[3]One can also define $Z : O \times A \times S \to \mathbb{R}$ to be a density function such that $\int Z(o \mid a, s')\, do = 1$.

setting, this is termed *state uncertainty*. More formally, we say the agent is unaware of the true state s, but instead uses the observation o to infer information on the state. The agent maintains a probability distribution b over states (termed a "belief") throughout interaction with the environment. The update function of which is dependent on the current belief, the observation o, and action a. In mathematical terms, b_t is a function of o_t, a_t and b_{t-1}, and due to the recursive nature of belief updates, b_{t-1} itself incorporates information provided by all previous actions and observations $a_1, o_1, \ldots, a_{t-1}, o_{t-1}$.

The agent's goal is to maximize the expected sum of discounted rewards by finding an *optimal policy*. In the POMDP setting, a *policy* π maps beliefs to actions, contrasting with the MDP scheme where policies directly map observable states to actions. Finding an optimal way to behave, in our case, selecting the optimal survey path, is termed "solving" the POMDP. Formally, given a belief b, "solving" the POMDP means finding an optimal policy π^* such that[4]

$$\pi^*(b) = \arg\max_a Q^\pi(b,a),$$

$$Q^\pi(b,a) = \mathbb{E}_\pi \left[\sum_{t=1}^{T} \gamma^t R(s_{t-1}, a_t, s_t) \;\middle|\; b_0 = b, a_1 = a, \pi \right],$$

where the "action-value" function $Q^\pi(b,a)$ is the expected return of taking action a in belief b and following policy π thereafter.

2.4.2 Solving a partially observable Markov decision process

2.4.2.1 Review of solver progress

POMDP solvers find optimal or near-optimal policies for decision making. The challenge of solving POMDPs is well-established, with foundational work dating back to the 20th century, as demonstrated by studies in the field of robotics planning (Monahan, 1982; Lovejoy, 1991; White, 1991; Littman et al., 1995). In recent years, the field has seen significant progress, with new algorithms making it possible to solve POMDPs with up to 10^{56} states (Ye et al., 2016; Sunberg and Kochenderfer, 2018).

Irrespective of when they were developed, solvers fall into two main categories: *offline* algorithms (Hauskrecht, 2000; Browne et al., 2012; Shani et al., 2013) compute a policy for all possible belief states before starting policy execution, while *online*

[4]There exist many conventions for indexing the timestep of elements in a POMDP tuple. See the caption in Fig. 3.1 for an explanation of the notation we use.

algorithms (Ross et al., 2008; Browne et al., 2012; Ye et al., 2016; Sunberg and Kochenderfer, 2018) compute optimal actions only for the current belief state. Online solvers are dominated by tree-based solvers (Browne et al., 2012; Ye et al., 2016; Sunberg and Kochenderfer, 2018; Mern et al., 2021). For the remainder of this section, we reflect on advancements in tree-based online solvers. We begin with the most basic tree construction algorithm and progress to POMCPOW (Sunberg and Kochenderfer, 2018), the algorithm employed in our work. We begin by assuming the state space S, action space A, and observation space O are all discrete.

The foundational tree-based algorithm, forward search (Kochenderfer et al., 2022), constructs a tree with the current belief[5] b_t as the root node and adds a node for each action available to the agent. All possible observations o that could feasibly be observed immediately following action a are then added to the tree. Action-observation layers are added repeatedly until a predefined maximum depth is reached. The philosophy is to evaluate the value of actions by averaging the rewards received in all branches below a given action node.

This simple example highlights the *curse of history*: the number of action-observation histories[6] grows exponentially in the planning horizon. Assuming depth d, the algorithm has complexity $O(|A|^d|O|^d)$. This, coupled with the *curse of dimensionality*, which expresses the number of states also grows exponentially with the number of state variables, makes POMDP planning at scale very challenging. These challenges underscore the need for more advanced algorithms.

Fortunately, *POMCP* (Silver and Veness, 2010) can be used to break both curses. The algorithm employs Monte Carlo sampling to reduce the branching factor dramatically. It is an extension of Monte Carlo tree search (MCTS; Coulom 2007; see Section 2.4.3.1) for MDPs where action-state trajectories are sampled to evaluate optimal actions. Under the POMCP regime, action-observation trajectories are sampled instead, reflecting the nature of using observations to infer information on the unobservable state.

However, in the setting of continuous observation spaces[7], POMCP breaks down. This is due to the inherent difficulty in discretizing an infinite number of possible observations. In fact, since the probability of generating the same real number twice

[5]The current belief, b_t, refers to the initial belief if the agent is yet to take an action ($t = 0$), or more likely, a belief formulated based on the sequence of actions and observations $a_1, o_1, \ldots, a_t, o_t$.

[6]A history is an action-observation sequence $a_1, o_1, \ldots, a_t, o_t$.

[7]This is the case for continuous action spaces too, but we omit their inclusion since we use a discrete action space in our work.

from a continuous random variable is zero, the width of the tree explodes at the first observation layer. While many approaches have been explored for handling continuous observation spaces, we only include here a description of POMCPOW (Sunberg and Kochenderfer, 2018), an extension of POMCP, since it is the algorithm used to solve our geophysical data acquisition POMDP.

2.4.3 POMCPOW

2.4.3.1 POMCP

To grasp the mathematics underlying POMCPOW, it is helpful to first understand POMCP; Monte Carlo tree search method tailored for POMDPs. POMCP is a recursive algorithm that can solve POMDPs approximately. In the tree, the root node represents the current belief. Each observation node o_n has a history $h = a_1, o_1, \ldots, a_n, o_n$, determined by its path from the root node. Action nodes store a value $Q(h,a)$, indicating how promising action a is, and a count $N(h,a)$, which tracks the number of visits to the node. The algorithm proceeds as follows.

Selection. (5)[8] The algorithm selects the action node from the tree that maximizes the upper confidence bound

$$UCB(h,a) = Q(h,a) + c\sqrt{\frac{\log N(h)}{N(h,a)}}, \tag{2.1}$$

where c governs exploration and $N(h) = \sum_a N(h,a)$ corresponds to the total count of visits to the history h. This approach is built on *upper confidence trees* (UCT; Couëtoux et al. 2011), which balances exploration and exploitation through the exploration constant c. The first term encourages exploitation by prioritizing actions with high action value $Q(h,a)$, while the second term incentivizes the selection of poorly explored actions with a small number of visits $N(h,a)$.

Expansion. (6) After selecting action a, the agent interacts with the environment. The agent receives a reward r and observation o, and hidden from the agent, a new state s' is generated according to the transition model $T(s' \mid s,a)$.

Simulation. (20) The tree search algorithm makes a recursive call and parses node (h,a,o) as the tree root and s' as the true state of the environment. The return of this recursive call, which will likely make many recursive calls itself, is combined with the

[8] We use $(\cdot)$ to list the corresponding line in POMCPOW (Alg. 1).

reward r received in the previous step to generate a value R as a measure for the quality of action a.

Backpropogation. (24) The Q-value estimate $Q(h,a)$ is updated using R, thereby ensuring the Q-value estimate is a running average. This running average mechanism makes POMCP an anytime[9] algorithm.

While POMCP (and related MCTS methods) limit tree width through Monte Carlo sampling, the growth of the tree remains exponential with the planning horizon. Additional heuristics are needed to control tree depth and maintain tractability. Action-value estimates at the final layer of the tree are often employed. Such knowledge need not be complete as long as it is able to bias action selection in a favourable fashion. In well-studied domains like Chess, standard heuristics are available (Browne et al., 2012); however, in less typical problems, such as geophysical data acquisition, custom heuristics are needed to achieve desired agent behaviour (see Section 3.3.2).

2.4.3.2 Progressive widening

To overcome the previously introduced explosion in width under continuous observation spaces, *progressive widening* (Couëtoux et al., 2011) can be used. Under such a regime, nodes are created only for a discrete subset of the continuous space. The approach can be applied to the action or observation layer of the tree, with *double progressive widening* corresponding to both simultaneously. As explained in Sec 3.3, our action space is discrete and we therefore omit an explanation of progressive widening for actions here.

Progressive widening for observations artificially limits the number of children $|C(h,a)|$ of node (h,a) depending on the number of visits $N(h,a)$. For hyperparameters $k \geq 0, \alpha \in [0,1]$, an observation node is created for an observation o if the following is satisfied

$$|C(h,a)| \leq k \cdot N(h,a)^{\alpha}. \tag{2.2}$$

For a visual representation of this condition in practice, see Fig. 2.2. If the condition is not met, the subsequent behaviour depends on the specific tree construction algorithm used. In our case, POMCPOW disregards the candidate observation o^i, and instead uses an observation sampled from existing child observation nodes using sample weights

[9] An algorithm that can be stopped at any time during its execution and still provide a valid solution to the problem at hand.

based on the number of visits $M(h,a,o)$ (see lines 7-11 in Alg. 1).

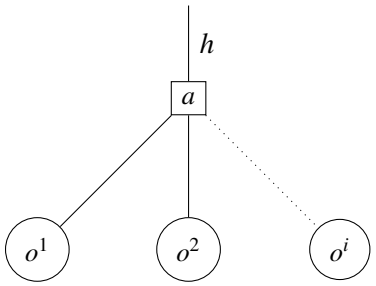


Figure 2.2: A component of a search tree, with history h leading up to the node (h,a). Currently, $C(h,a) = 2$ since (h,a) has two children (h,a,o^1) and (h,a,o^2). The candidate observation o^i is added to the tree at a new node (h,a,o^3) if Equation 2.2 is satisfied. By inspecting Equation 2.2, it is clear the width $|C(h,a)|$ of the observation layer grows with the number of visits $N(h,a)$. This ensures the most promising actions have rich beliefs.

2.4.3.3 POMCPOW: an extension of POMCP

In their survey on Monte Carlo tree search methods, Browne et al. (2012) state "double progressive widening worked well for toy problems, but less so for complex real-world problems". Since then, POMCPOW (Sunberg and Kochenderfer, 2018) has overcome these challenges by using weighted particle filters at observation nodes to create implicit beliefs and progressive widening in observation (and action) layers. The algorithm retains the nonmyopic planning advantages inherent to the POMDP formulation and is adept at managing continuous observations. These features are particularly beneficial in our context, as the algorithm enables seamless integration of continuous geophysical data and considers the plane's resulting location when selecting the next action. We include our implementation of the algorithm in Alg. 1 and include an example of a POMCPOW tree in Fig. D.1.

Algorithm 1 POMCPOW (Sunberg and Kochenderfer, 2018) with minor adaptions to demonstrate the implementation we use in our work. Our discrete action set (see Sec. 3.3) omits the need for action progressive widening, which the original algorithm uses on line 5. Further, at depth $d = 0$ on line 3, we return a heuristic value (see Sec. 3.3) instead of the original algorithm's return value of 0. A history-action-observation path is denoted (h,a,o). A POMCPOW tree is visualized in Fig. D.1.

1: **procedure** SIMULATE(s,h,d)
2: **if** $d = 0$ **then**
3: **return** $H(s)$ ▷ use heuristic at max depth (Eq. 3.3)
4: **end if**
5: $a \leftarrow \arg\max_{a \in C(h)} Q(h,a) + c\sqrt{\frac{\log N(h)}{N(h,a)}}$ ▷ *selction* step (Sec. 2.4.3.1)
6: $s', o, r \leftarrow$ ENVIRONMENT(s,a) ▷ *expansion* step (Sec. 2.4.3.1)
7: **if** $|C(h,a)| \leq k \cdot N(h,a)^{\alpha}$ **then** ▷ observation progressive widening (Eq. 2.2)
8: $M(h,a,o) \leftarrow M(h,a,o) + 1$
9: **else**
10: $o \leftarrow$ select $o \in C(h,a)$ w.p. $\frac{M(h,a,o)}{\sum_{o'} M(h,a,o')}$
11: **end if**
12: append s' to $B(h,a,o)$
13: append $Z(o \mid s,a,s')$ to $W(h,a,o)$
14: **if** $o \notin C(h,a)$ **then**
15: $C(h,a) \leftarrow C(h,a) \cup \{o\}$ ▷ add new observation node (Fig. 2.2)
16: $total \leftarrow r + \gamma$ROLLOUT$(s', (h,a,o), d-1)$
17: **else**
18: $s' \leftarrow$ select $B(h,a,o)[i]$ w.p. $\frac{W(h,a,o)[i]}{\sum_j W(h,a,o)[j]}$
19: $r \leftarrow R(s,a,s')$
20: $total \leftarrow r + \gamma$SIMULATE$(s', (h,a,o), d-1)$ ▷ *simulation* step (Sec. 2.4.3.1)
21: **end if**
22: $N(h) \leftarrow N(h) + 1$
23: $N(h,a) \leftarrow N(h,a) + 1$
24: $Q(h,a) \leftarrow Q(h,a) + \frac{\text{total} - Q(h,a)}{N(h,a)}$ ▷ *backpropogation* step (Sec. 2.4.3.1)
25: **return** $total$
26: **end procedure**

Chapter 3

Methodology

3.1 Problem formulation

Our study aims to demonstrate the potential of the POMDP formulation in the context of geophysical data acquisition. To achieve this, we employ a synthetic case encompassing all components of a sequential decision-making problem. The advantage of using a synthetic scenario lies in our ability to fully determine the true state (ground truth). The synthetic case is created sequentially: a map of a *geophysical derivative* m_s (see Fig. 3.4 *(a)*) is created using the *GeoStats.jl* package (Hoffimann, 2018) and then transformed into a map representing a *geophysical signal* z_s (see Fig. 3.4 *(d)*)[1]. The signal z_s is considered the noisy counterpart of m_s (see Section 3.4). The maps are decoupled because belief updating (see Section 3.5.2) is only computationally feasible on the lower-resolution map; the derivative map.

The domain of our simulated case is an area of 2400 m^2, discretized into a two-dimensional grid over 48×48 and 192×192 grid cells for the derivative and geophysical map, respectively. Each cell has a value in the range $(0, 1)$ representing the magnitude of the geophysical anomaly at that location[2]. The derivative map is constructed by generating a signal with random shape, orientation, and magnitude. Finally, the value $v(s)$ of a synthetic case (state) s, which represents the magnitude of the state's anomaly, is determined by

$$v(s) = \sum_{i=1}^{48} \sum_{j=1}^{48} \mathbf{1}\{m_s(x_{ij}) \geq \theta\}, \tag{3.1}$$

[1] We refer to each as the *derivative* and *geophysical* map for the remainder of the report.

[2] All map plots in this report correspond to the colour bar in Fig. C.1.

where $\mathbf{1}$ is the indicator function, $\theta = 0.7$ is a magnitude threshold, and x_{ij} is grid element (i, j) on the derivative map.

3.2 Agent dynamics model

We model our agent as a single fixed-wing aircraft and assume consistent altitude and constant velocity v ms^{-1}. We define the *agent dynamics tuple* of state s_t at timestep t to be the set $\{x_{s_t}, y_{s_t}, \psi_{s_t}, \phi_{s_t}\}$, where continuous (x_{s_t}, y_{s_t}), continuous ψ_{s_t}, and discrete ϕ_{s_t} define the aircraft's current position, heading, and bank angle, respectively. Given the tuple $\{x_{s_{t-1}}, y_{s_{t-1}}, \psi_{s_{t-1}}, \phi_{s_{t-1}}\}$ at the previous timestep $t-1$, the agent dynamics tuple updates are governed by the current bank angle of the plane ϕ_{s_t} as follows:

$$\bar{\psi}_t = \frac{g\tan(\phi_{s_t})}{v} \qquad \psi_{s_t} = \psi_{s_{t-1}} + \bar{\psi}_t$$
$$x_{s_t} = x_{s_{t-1}} + v\cos(\psi_{s_t}) \qquad y_{s_t} = y_{s_{t-1}} + v\sin(\psi_{s_t}),$$

where g is the sea-level gravitational acceleration constant. We use $g = 9.80665$ ms^{-1} and $v = 40$ ms^{-1}. [3] At each timestep, the bank angle of the aircraft ϕ_{s_t} can changed by $a \in \{-\phi_{\text{CH}}, 0, \phi_{\text{CH}}\}$. To enforce safe flight conditions, a limit such that $|\phi| \leq \phi_{\text{MAX}}$ is imposed by restricting choices for a (see Section 3.3). The magnitude of ϕ_{CH} governs how rapidly the agent can change direction.

3.3 Problem setting

3.3.1 Problem specific formulation

Following the introduction of the general POMDP framework in Section 2.4.1, our geophysical data acquisition POMDP is defined as follows.

States. The state space comprises the simulated ground truth derivative and geophysical maps, the previously defined agent dynamics tuple, and observation history. The simulated ground truth map is consistent for all states $s_0, \ldots, s_T$, where T is the timestep

[3] Most airborne gravity surveys are flown at a ground speed of 60 ms^{-1} (Dransfield, 2007) but in preliminary experiments the agent failed to consistently remain on the map at increased velocities. This is because increasing v limits the agent's ability to turn sharply (see the inverse relationship between ψ_t and v in Section 3.2). A larger planning horizon would allow the agent to plan ahead further when approaching a boundary and therefore permit larger v.

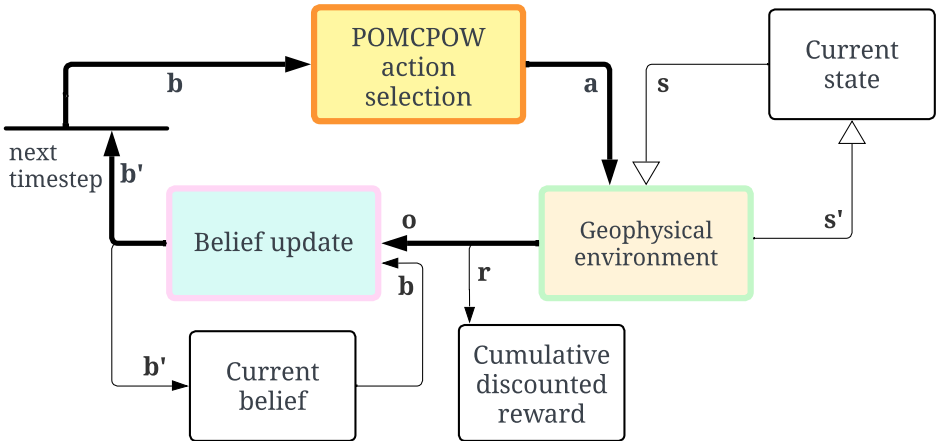


Figure 3.1: The outer loop of the POMDP. *POMCPOW action selection* selects an action $a_t = a$ and receives a reward $r_t = r$ and observation $o_t = o$ generated by the *geophysical environment*. However, the true states $s_{t-1} = s$ and $s_t = s'$ remain hidden from the agent, as indicated by the differing arrows. The loop begins at *POMCPOW action selection* with initial belief b_0 and—following the *geophysical environment* generating terminal $s_T = s'$ at timestep T—terminates at the subsequent *belief update*, which produces the final belief b_T.

at termination. The observation history contains all geophysical readings made by the agent thus far, as well as each reading's coordinate location.

Actions. The actions accessible to the agent comprise GO, NO-GO, and FLY. Consistent with the function of geophysical surveys in mineral exploration—which is to quickly acquire data to guide borehole planning—the GO and NO-GO actions correspond to initiating a drilling campaign or abandoning the region, respectively. The FLY action is further subdivided into three options: MAINTAIN the bank angle from the previous timestep, or choose to INCREASE or DECREASE the bank angle by ϕ_{CH}.

While GO, NO-GO, MAINTAIN, INCREASE, and DECREASE are available to the agent in general, the action set is constrained at each timestep. Formally, the permitted action set at timestep t is a function of the current belief b_t. When the agent begins environment interaction, its goal is to gather information. Hence, we do not allow the agent to select GO or NO-GO until a desired level of confidence is achieved (see Section 3.6). We also restrict the action set by removing INCREASE or DECREASE if performing such an action would violate the constraint $|\phi_{s_t}| \leq \phi_{\text{MAX}}$.

Transition function. The transition model $T(s' \mid a, s)$ is a probability distribution over the next timestep state s' conditioned on the current state s and action a. In our case, two of the three components in the state space change: namely, the observation history and agent dynamics tuple; the synthetic case's map remains constant. The transition

model $T(s' \mid a, s)$ generates terminal s' for all states s if $a \in \{\text{GO,NO-GO}\}$.

Reward function. We employ a profitability-based reward function[4]

$$R(s,a) = \begin{cases} -c_{\text{FLY}} - c_{\text{OOB}} \cdot \mathbf{1}\{\text{OOB}_s\} & \text{if } a = \text{FLY}, \\ \text{Profit}(s) := \alpha_v \cdot v(s) - c_{\text{GO}} & \text{if } a = \text{GO} \\ 0 & \text{if } a = \text{NO-GO}, \end{cases} \tag{3.2}$$

where the out-of-bounds indicator takes value 1 if one of the agent's coordinates x_s or y_s in state s lay outside the interval $[0, 2400]$; c_{OOB} is a penalty for the agent selecting paths outside of the mapped region; $v(s)$ is the state anomaly value; α_v is a multiplier that converts the state anomaly value to monetary value[5]; and c_{GO} is a predefined cost of GO that embodies both drill campaign costs and potential deposit extraction costs. The addition of c_{OOB} introduces a minor departure from a purely profit-based reward; this adjustment can be nullified by setting $c_{\text{OOB}} = 0$.[6]

The agent's aim is to maximize the expected sum of discounted rewards through action selection. Thus, a negative reward at each timestep incentivizes the agent to strategize a path that facilitates timely termination. Minimizing the number of timesteps until termination translates to minimizing the distance flown in an airborne geophysical survey. Further penalization by c_{OOB} for departing the mapped region encourages the agent to explore only the mapped area.

Observations. We generate one observation per timestep at the agent's current position (x_{s_t}, y_{s_t}) on the geophysical map. Although generating multiple observations per timestep might seem beneficial, the marginal gain is minimal since uncertainty near an existing reading is typically low, and this benefit is outweighed by the increased computation cost of an additional observation when performing belief updates (see Section 3.5.2). The process by which the agent observes is visualized in Fig. 3.2 and the addition of noise is further explained in Section 3.4.

Observation model. The probability $Z(o \mid a, s')$—of observing o when transitioning to state s' after taking action a—defines the effect of noise on the data generated by measurements. We use $Z(o \mid a, s') = N(o - m_{s'}(X_o); \mu, \sigma)$, where X_o is the location at which o was observed, $\mu = 0.25$, and $\sigma = 0.005$ is the *sill* as defined in Section 3.4. This

[4] In Section 2.4.1, we defined the general POMDP reward function as $R : S \times A \times S \to \mathbb{R}$, but our choice or reward $R(s,a)$ rather than $R(s,a,s')$ gives a function defined by $R : S \times A \to \mathbb{R}$.

[5] We use $\alpha_v = 1$ in our experiments for simplicity.

[6] We use $c_{\text{OOB}} = 0$ in all experiments because preliminary tests found our heuristic (see Section 3.3.2) is more effective at encouraging the agent to remain on the map.

model is used by POMCPOW to generate weights for simulated observations and these weights are subsequently used during resampling if the progressive widening condition in Equation 2.2 is not satisfied (see Section 2.4.3.2 and lines 13 & 18 in Alg. 1). The observation model is further used by the belief updating mechanism (see line 4 in Alg. 2).

Discount factor. We use a discount factor of $\gamma = 0.99$ to encourage the agent to plan with a nonmyopic focus. Choosing a value close to 1, which would place equal weight between the immediate reward r_t and future rewards $r_{t+1}, r_{t+2}, \ldots, r_T$, ensures the agent considers the impact of an action on its future location and actions available thereafter during action selection.

3.3.2 Problem specific heuristics

POMCPOW (and other MCTS algorithms) do not permit efficient learning based on the reward system alone because—in the *Simulation* step in 2.4.3.1—it is computationally unfeasible to simulate the POMDP until a terminal state is reached. Instead, simulations are limited to a maximum depth and once this depth is reached, a heuristic is used to identify high-quality actions. Our heuristic function is defined on non-terminal states s as

$$H(s) = \begin{cases} w_{\geq 0} \cdot \mathbf{1}\{0 < x_s, y_s < 2400\} \cdot \text{Profit}(s) & \text{if Profit}(s) \geq 0, \\ w_{<0} \cdot \text{Profit}(s) & \text{if Profit}(s) < 0, \end{cases} \tag{3.3}$$

where the indicator function takes value 1 if the agent's coordinates x_s and y_s in state s lay within the mapped region. Similarly to c_{OOB} in Equation 3.2, this manipulates action-value functions to encourage the agent to remain in the mapped region. Parameters $w_{\geq 0}$ and $w_{<0}$ are used to control how much weight is applied to profitable and unprofitable cases, respectively. We place significantly more weight on profitable cases by choosing $w_{\geq 0} = 0.9$ and $w_{<0} = 0.1$ This is necessary since equal weighting could lead to NO-GO being chosen earlier than desired. If the weights were equal, say $w_{\geq 0} = w_{<0} = 1$, and a particle filter at a leaf node in the tree contains states $\{s_1, \ldots, s_n\}$ such that $\sum_{i=1}^{n} \text{Profit}(s_i) \approx 0$, then the agent would consider $a = \text{NO-GO}$ to be approximately as promising as $a = FLY$ since $Q(h, \text{NO-GO}) = 0$ for all histories h.[7] By placing more weight on profitable states in the filter, we encourage the agent to choose $a = \text{FLY}$

[7] When $a = \text{NO-GO}$, $Q(h,a) = 0$ since $R(s,a) = 0$ and all states s' generated by $T(s' \mid s,a)$ (line 6 in Alg. 1) are terminal.

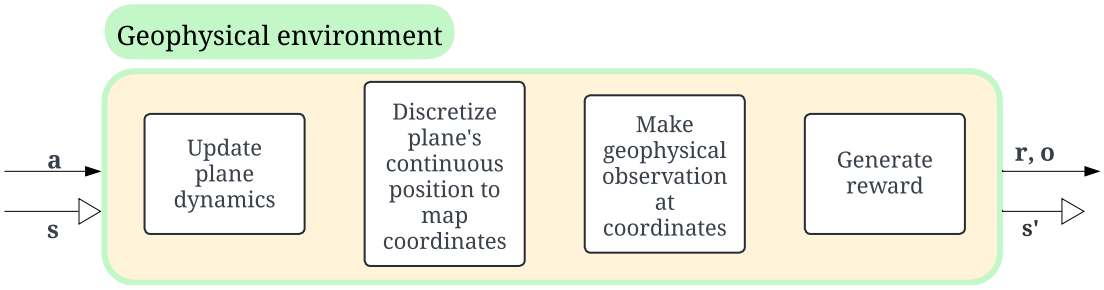


Figure 3.2: A summary of the agent's interaction with the environment. The agent selects an action a and receives a reward r and observation o. However, the true states s and s' remain hidden from the agent, as indicated by the differing arrows.

unless there is significant evidence the particle filters at leaf nodes suggest the true state s has $\text{Profit}(s) < 0$.

3.4 A robust noise mechanism

We impose noise on the derivative m_s to generate the geophysical map with signal z_s—from which the agent makes observations—for two key reasons. First, it is crucial that our synthetic case realistically reflects the inherent noise in real-world geophysical data acquisition. The noise sources we consider include geological background variation, regional noise, and sensor noise. Geological background variation accounts for minor geophysical anomalies caused by subsurface minerals directly beneath the sensor. Regional noise captures the influence of nearby minerals on geophysical readings, while sensor noise stems from aircraft motion. The second key reason for incorporating noise is to prevent particle filter collapse. Insufficient noise can cause the filter to converge prematurely to a single estimate, resulting in overconfidence and a loss of diversity within the particle set.

Background variation noise is introduced using a Gaussian process (see App. A) with a known mean and covariance structure. By setting a positively-valued mean $\mu_{\text{GP}} = 0.25$, we assume background noise is present across the entire map, reflecting the influence of other minerals on geophysical readings at all locations. The noise's dependence structure is modelled with a spherical variogram, which is well-suited for geological data exhibiting gradual spatial changes up to a certain range, beyond which changes become random and uncorrelated. For an overview of spherical variogram models and their key parameters—sill s, range r, and nugget n—refer to Appendix B. In this work, we adopt $(s, r, n) = (0.005, 30, 0.0001)$, following the values used in the

borehole data acquisition POMDP implemented by Mern and Caers (2023). While we acknowledge that more optimal values may exist given the transition from borehole data in their work to geophysical data in our study, we prioritize addressing more critical challenges with a greater impact on the project's overall objectives. The traditional agent's near-perfect accuracy in our numerical experiments (see Ch. 4) supports the validity of this assumption.

A Gaussian filter (Bergholm, 1987) with smoothing parameter $\sigma_{\text{GF}} = 3$ is applied to generate the geophysical map from the derivative map, effectively computing a weighted average of nearby grid cells and thereby introducing regional noise. Sensor noise, caused by aircraft motion, is modelled by adding a noise term $\varepsilon \sim N(0, \sigma_{\text{SN}})$ to all observations, where $\sigma_{\text{SN}} = 0.005$ is used in all experiments. An additional consideration is modelling sensor noise as a function of aircraft bank angle, capturing the improvement in sensor performance when the aircraft is level. We leave this as a suggestion for future work.

3.5 Belief representation and updating

3.5.1 Belief representation

The belief, which is a probability distribution over states, is an unweighted ensemble of all particles in a particle filter. Increasing the number of particles in a particle filter enhances the accuracy and robustness of posterior state estimates while reducing the risk of premature particle convergence (particle depletion), though at the cost of higher computational expense during belief updates (see Section 3.5.2). We use $N = 1000$ particles for our problem which proved sufficient for obtaining accurate estimates for $v(s)$ during numerical experiments (see Ch. 4). For clarity, we use m_s and z_s for the ground truth s; $\bar{m}_{p_i}$ and $\bar{z}_{p_i}$ to refer to the derivative and geophysical signal of particle p_i; and $\hat{m}(b)$ and $\hat{z}(b)$ to refer to belief b's mean estimate for m_s and z_s. The standard deviation of estimates $\hat{m}(b)$ and $\hat{z}(b)$ is captured by σ_b^m and σ_b^z. See Fig. 3.3 *(d)* and 3.4 *(b)* for a visualization of the mean estimate $\hat{z}(b_0)$ for initial belief b_0 with $N = 3$ and $N = 1000$ particles, respectively, as well as a visualization of initial uncertainty σ_b^z in Fig. 3.4 *(e)* for the $N = 1000$ case.

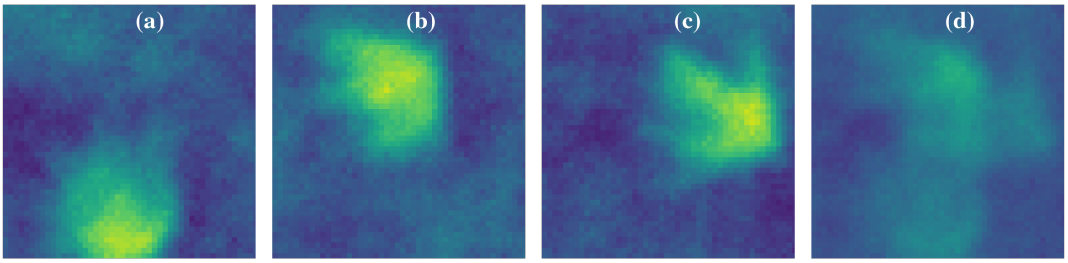


Figure 3.3: The belief mean $\hat{z}(b)$ *(d)* is an unweighted average of particles $\{\bar{z}_{p_1}, \bar{z}_{p_2}, \bar{z}_{p_3}\}$ *(a-c)*. In this visualization, we use $N = 3$ particles but our experiments use $N = 1000$ which gives a uniform prior belief.

As outlined in Section 2.3, particle filters are comprised of a set of candidate states for the ground truth state. We therefore generate each particle according to the same process used to produce the ground truth (see Section 3.1). When performing mineral exploration in the real world, where the generating distribution for the ground truth is unknown, the particles would instead be estimated by experts using data already available. At initialization, all particles in the set differ greatly. This is achieved by generating derivatives $\bar{m}_{p_i}$ with significant variations in terms of shape, orientation, and magnitude. Consequently, an almost uniform initial belief is established in Fig. 3.4 *(b)*, with a slight concentration of weight towards the centre of the map due to the centring imposed when generating derivatives $\bar{m}_{p_i}$ $(i = 1, \ldots, N)$. Through our *belief update* mechanism, which leverages observations made by the agent to infer information on the true state, our particle set converges to the ground truth (see Fig. 3.4 *(c)*). The success of this convergence hinges on the fulfillment of specific conditions, which we review now.

3.5.2 Belief updating

The outer loop of the POMDP in Fig. 3.1 shows that, between two belief updates, an action a is selected and observation o is generated. Hence, at each iteration of the belief update, the prior belief b must be updated to the posterior belief b' by incorporating a and o. Firstly, we define $b(s) = p(s \mid b)$ as the probability of being in state s given the belief b. Under the general POMDP model formulation, after transitioning to state s' by taking action a in state s and receiving observation o, the prior belief b is updated to the posterior belief b' as follows[8] (Kochenderfer et al., 2022)

[8]To be explicit regarding timestep t, here $s = s_{t-1}, a = a_t, o = o_t, s' = s_t$, and $b' = b(s') = b(s_t)$.

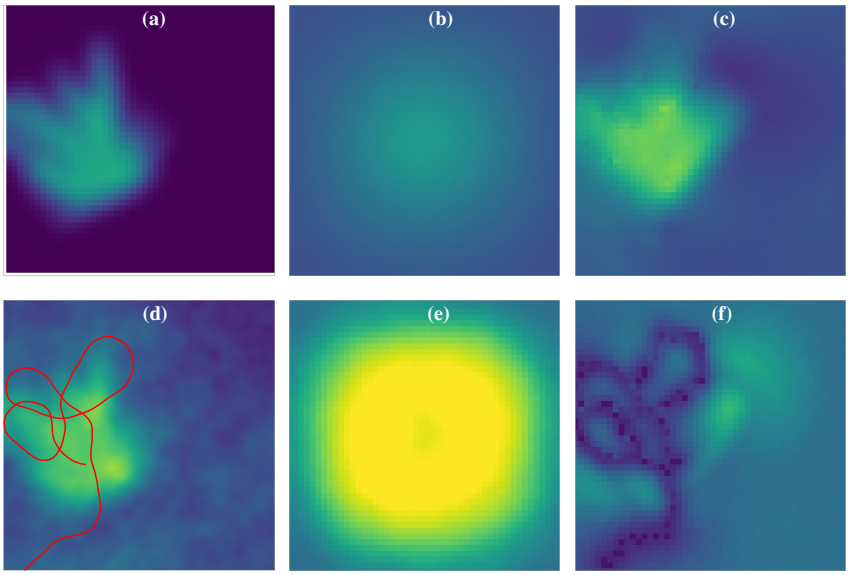


Figure 3.4: An example of a derivative map m_s *(a)* and its corresponding geophysical map z_s with the agent's path visualized *(c)*. The mean *(b)* and standard deviation *(e)* of the initial belief are visualized beside the mean *(c)* and standard deviation *(f)* of the final belief at termination.

$$
\begin{aligned}
b' &:= p(s' \mid b,a,o) \\
&\propto p(o \mid b,a,s')p(s' \mid b,a) && \text{(Bayes' rule)} \\
&= O(o \mid a,s') \int p(s' \mid a,b,s) \cdot p(s \mid b,a)\, ds && (o \perp b) \\
&= O(o \mid a,s') \int T(s' \mid a,s) b(s)\, ds. && (s' \perp b) \qquad (3.4)
\end{aligned}
$$

The posterior b' represents the probability of being in state s' given the prior belief b, action a selected by the *POMCPOW action selection* component, and observation o generated by the *geophysical environment*. At the next timestep, the process repeats; only terminating once the *geophysical environment* generates a terminal state $s' = s_T$ at timestep T, at which point, a final belief update is performed to produce the final belief b_T.

However, continuous observation spaces make the update too complex to perform analytically. Particle filters demonstrate their effectiveness by offering a computationally feasible approximation of Equation 3.4. The approach works by calculating weights w_i for each particle p_i according to the observation received. Using Bayes rule,

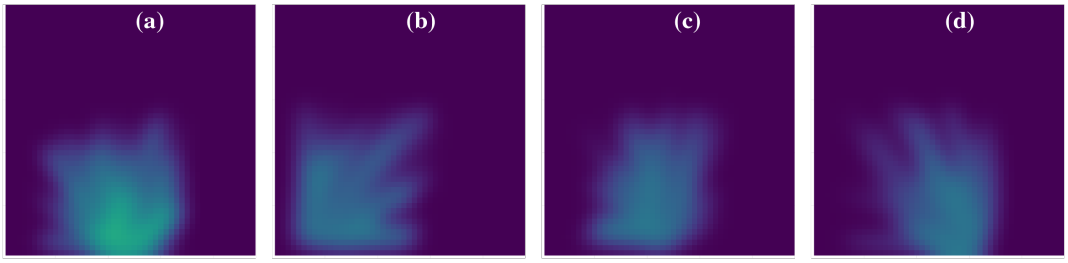


Figure 3.5: An illustration of the perturb functionality in line 13 of Alg. 2. The derivative field m_s *(a)* is perturbed in terms of shape, orientation, and magnitude *(b-d)*. In this example, noise magnitude is governed by $\omega = 1$.

$$
\begin{aligned}
w_i &\propto f(o_t \mid m_{p_i}, o_{1:t-1}) \\
&\propto f\left(o_t - m_{p_i}(X_{o_t}) \middle| \left\{o_j - m_{p_i}(X_{o_j})\right\}_{j=1,\ldots,t-1}\right),
\end{aligned} \tag{3.5}
$$

where o_t is the recently made observation; X_{o_j} is the location at which the agent observed observation o_j; and $m_{p_i}(X_{o_j})$ is the value of derivative m_{p_i} at location X_{o_j}. The function f corresponds to a GP model with mean $\mu_{\text{GP}} = 0.25$ and covariance matrix governed by the parameters of our spherical variogram model (see Section 3.4) and the observation locations $\{X_{o_1}, \ldots, X_{o_t}\}$. Next, the particle set undergoes resampling with replacement, guided by the sample weights $\{w_i\}_{i=1,\ldots,1000}$. At this point, careful attention must be paid to ensure the particle filter remains effective. It is possible for extreme weights to cause the particle filter to collapse into a limited set of duplicated particles, thereby negating the benefits of its probabilistic representation of the posterior distribution. We navigate this threat by making adjustments to each particle in terms of shape, orientation, and magnitude. We say the particle is *perturbed*, the magnitude of which is governed by noise parameter ω (see Fig. 3.5). While it is designed to prevent particle collapse (a loss in diversity), ω could cause divergence of the particle filter (too much diversity) if chosen to be too large, and hence, it must be tuned during experiments. The entire belief update mechanism is described explicitly in Algorithm 2.

3.6 Decision making

The agent's ultimate aim is to make a GO/NO-GO decision. With reference to Fig 3.1, if $a \in \{\text{GO, NO-GO}\}$, then the *geophysical environment* returns a terminal state $s' = s_T$ and the *belief update* is performed once more to produce a final belief. However, for

this to occur, the *POMCPOW action selection* component must allow the agent to select such actions by constructing trees like those shown in Fig. 3.6. Initially, the action set is constrained to flying actions: INCREASE, DECREASE, and MAINTAIN. Only once a stop criterion—representing an arbitrary level of confidence in belief b about $v(s)$—is satisfied, does the *POMCPOW action selection* component construct a tree like the examples in Fig 3.6. This confidence is achieved through convergence of the particle filter. Formally, we extend Equation 3.1 to define the value of a belief b as follows

$$v_b = \sum_{i=1}^{48} \sum_{j=1}^{48} \mathbf{1}\left\{ \hat{m}_b(x_{ij}) \geq \theta \right\}, \tag{3.6}$$

where $\hat{m}_b(x_{ij}) = \frac{1}{N} \sum_{k=1}^{N} \bar{m}_{p_k}(x_{ij})$ is the grid element level average of particle derivatives $\bar{m}_{p_k}$ at grid location x_{ij} $(k = 1, \ldots, 1000)$. Then the stop criterion is met if one of the equations below are satisfied:

$$\text{UCB} = v_b + \alpha_{\text{U}} \sigma_b^m \leq c_{\text{GO}}, \quad \text{or} \quad \text{LCB} = v_b - \alpha_{\text{L}} \sigma_b^m \geq c_{\text{GO}}, \tag{3.7}$$

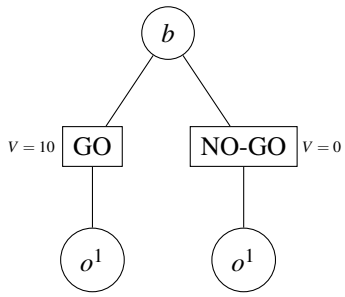
where α_{LCB} and α_{UCB} are parameters governing the required level of confidence, and c_{GO} is the previously defined cost of GO. The agent's decisions are evaluated to be correct or incorrect depending on $\text{Profit}(s) = v(s) - c_{\text{GO}}$, as defined in Equation 3.2. A simulated case is said to be "profitable" if $\text{Profit}(s) > 0$ and "unprofitable" otherwise. Decisions are classified to be "correct" or "incorrect" depending on the case's profitability and final action $a_T \in \{\text{GO,NO-GO}\}$, as shown in Tab. 3.1.

	a_T	
	GO	NO-GO
Profitable	Correct	Incorrect
Unprofitable	Incorrect	Correct

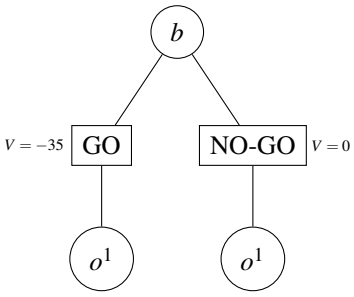
Table 3.1: Decision logic based on $\text{Profit}(s)$ of simulated case s and final action a_T.

3.7 Illustrative examples

This section provides a practical demonstration of the proposed framework using an illustrative example of sequential exploration in a synthetic environment. The geophysical map is constructed according to the framework presented in Section 3.1. The plane enters the mapped region at $x = 190$ m and $y = 0$ m heading northeast. The



(a) A positive action-value for GO will lead to $a =$ GO being selected.



(b) A negative action-value for GO will lead to $a =$ NO-GO being selected.

Figure 3.6: Trees constructed by the *POMCPOW action selection* component after the stop criterion is satisfied. Since the *geophysical environment* returns empty observations for $a \in \{\text{GO,NO-GO}\}$, all state trajectories sampled return the same observation, meaning there is only a single observation node following each action. The action-values of $a =$ GO and $a =$ NO-GO are computed according to the reward function (see Sec 3.3). Our heuristic $H(s)$ in Equation 3.3 does not influence action-values since all states in the particle filter at observation nodes are terminal.

value $v(s)$ of the case is 226 units and c_{GO} is 150 units, meaning $\text{Profit}(s) = 76$. This means the correct final decision is GO. We use $\alpha_{\text{U}} = \alpha_{\text{L}} = 0.9$ for the confidence bound parameters to provide a stringent stop criterion. All other parameters correspond to the values used for experiments, as explained in Section 4.1.

3.7.1 Intelligent agent

The discussion in this section is with reference to Fig. 3.7. At $t = 0$, the random generation of the particle filter creates a symmetric histogram with mean $\mu = 157.14$ and standard deviation $\sigma = 64.16$.[9] According to Equation 3.7, this gives an uninformative confidence bound with $(\text{LCB}, \text{UCB}) = (99.4, 214.9)$ given our choice of confidence

[9]We reduce notation of $\hat{z}(b)$ and σ_b^z to μ and σ, respectively, to align with the titles in the histograms in Fig. 3.7.

Algorithm 2 Particle filter update after taking action a and observing o

procedure UPDATEBELIEF($\mathbf{b} = \{p_i\}_{i=1}^{N=1000}, a, o$)
  $X_o \leftarrow$ discretized location of o
  for $i \in 1, \ldots, N$ **do**
    $w_i \leftarrow \textsc{Evaluate}\big(o - \bar{m}_{p_i}(X_o)\big)$ ▷ see Equation 3.5
  end for
  $\mathbf{w} \leftarrow \textsc{Normalize}(\mathbf{w})$
  $\bar{\mathbf{b}} \leftarrow \textsc{Sample}(\mathbf{b}, \mathbf{w})$
  for $p_i \in \bar{\mathbf{b}} = \{p_1, \ldots, p_N\}$ **do**
    if p_i is a duplicate particle **then**
      $\bar{m}_{p_i'} \leftarrow \textsc{Perturb}(\bar{m}_{p_i}, \omega)$ ▷ see Fig. 3.5
    end if
    $\bar{z}_{p_i'} \leftarrow \textsc{GenerateNoise}(\bar{m}_{p_i'})$ ▷ see Sec. 3.4
  end for
  $p_i' \leftarrow \{\bar{m}_{p_i'}, \bar{z}_{p_i'}\} \quad (i = 1, \ldots, N)$
  $\mathbf{b}' \leftarrow \{p_i'\}_{i=1}^{N}$
  return $\mathbf{b}'$
end procedure

bound parameters $\alpha_U = \alpha_L = 0.9$. At $t = 20$ (row 2 in Fig. 3.7), the red line entering from the southwest corner shows the path the agent has explored thus far. As a result of actions and observations $a_1, o_1, \ldots, a_{20}, o_{20}$, the agent has performed 20 iterations of belief updates and has therefore reduced uncertainty in the southwest corner of the map (shown by the change in colour in the standard deviation map). The extremely dark line on the standard deviation map has not been manually plotted, the dark shade corresponds to an extremely low uncertainty, which we expect at locations the agent has observed directly. The histogram is similar to that at $t = 0$ since the agent has not observed strong geophysical anomalies.

However, at $t = 25$ (row 3 in Fig. 3.7), the agent observes a strong geophysical signal and the particle filter shifts to have mean $\mu = 216.07$. Uncertainty is still significant with $\sigma = 78.11$. This leads to a confidence bound of $(145.8, 286.37)$, which still does not satisfy the criterion in Equation 3.7. However, after the agent has made another 10 observations at $t = 35$, the confidence bound becomes $(160.7, 267.5)$ and we have $\text{LCB} = 160.7 > 150 = c_{\text{GO}}$ thereby fulfilling the stop criterion. A tree is constructed by the *POMCPOW action selection* component that corresponds to those shown in

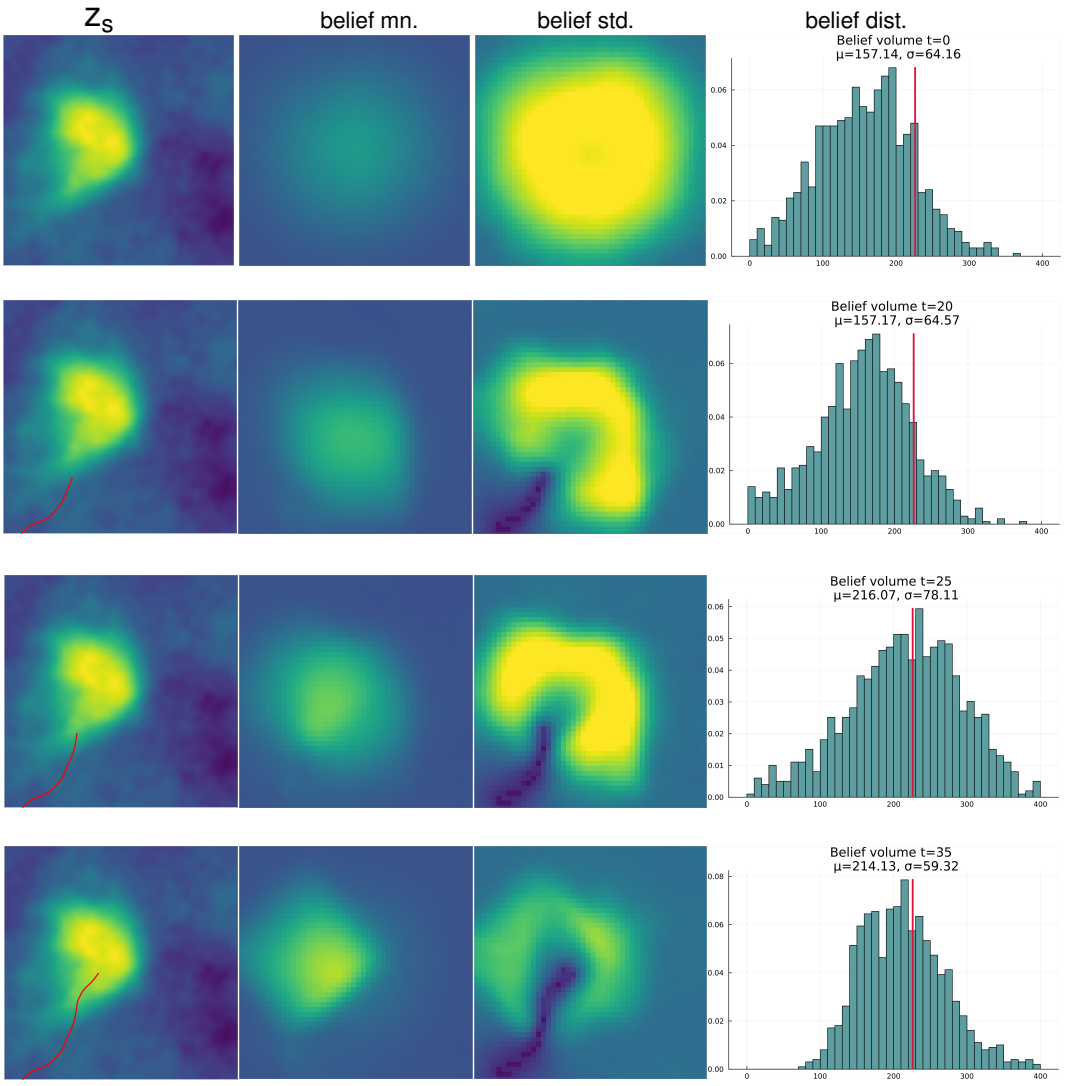


Figure 3.7: An illustration of the intelligent agent's behaviour. The geophysical map z_s with the agent's trajectory represented by the red line (left). The other visualizations show the belief's mean and standard deviation, as well as the particle filter's distribution. The rows are ordered and correspond to key timesteps $t = 0, 20, 25$ and terminal $T = 35$.

Fig. 3.6 and the agent selects $a = \text{GO}$. As a result, a terminal state is generated by the *geophysical environment* component and the algorithm terminates.

3.7.2 Traditional agent

This section refers to Fig. 3.8 throughout. The traditional agent mirrors current geophysical survey methods by following a grid structure (shown by red lines in Fig. 3.8), flying 10×2400 m survey lines at 300 m intervals (5 east-west and 5 north-south). At a speed of $v = 40\text{ms}^{-1}$, it takes 600 timesteps to complete the 24 km survey per case.

The time for manoeuvring between lines is ignored, as it would be negligible in larger real-world surveys.

The traditional agent terminates with mean $\mu = 226.8$ and $\sigma = 12.3$, leading to the correct decision $a_T = \text{GO}$. Comparing the standard deviation maps and histograms in Figs. 3.7 & 3.8, the traditional agent's extensive effort results in significantly more confidence in its conclusion, evidenced by the sharper peak in the histogram and the darker shade of the standard deviation plot. The final standard deviation of $\sigma = 12.32$ for the traditional agent is far smaller than $\sigma = 50.32$ for the intelligent agent. Additionally, the error of the value estimate $\mu = 226.81$ units is 0.81 units for the traditional case versus 11.9 for the intelligent agent (given $v(s) = 226$ units). This boost in confidence and accuracy is expected, given that the traditional agent explores for an additional 565 timesteps.

3.7.3 Financial comparison

Our hypothesis in Chapter 1 posits that adaptively planning geophysical flight paths with an information-driven approach could reduce survey distances and costs. The intelligent agent's correct decision after just 35 timesteps supports this idea. However, validating the hypothesis requires testing our approach across a set of cases with variations in anomaly shape, orientation and magnitude, which we do now.

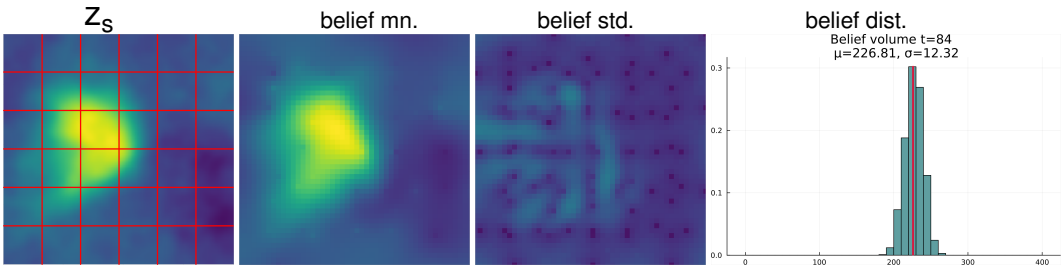


Figure 3.8: An illustration of the traditional agent's behaviour. The geophysical map z_s with the agent's trajectory represented by the red line (left). The other visualizations show the belief's mean and standard deviation, as well as the particle filter's distribution at $T = 600$.

Chapter 4

Experiments

4.1 Experiment setup

In this study[1], a total of 150 synthetic cases are used to evaluate the performance of our proposed method under varying conditions in terms of shape, magnitude, and orientation of the geophysical derivative. Each synthetic case is generated as described in Section 3.1.

The agent's path is governed by $\phi_{\text{CH}} = 18^\circ$ and $\phi_{\text{MAX}} = 54^\circ$. The agent dynamics tuple is initially set to $(x_{s_0}, y_{s_0}, \psi_{s_0}, \phi_{s_0}) = (190\text{ m}, 0\text{ m}, 45^\circ, 0)$, exactly as shown by the agent's starting position in Fig. 3.7. The heading $\psi_{s_0} = 45^\circ$ corresponds to a northeast heading. For the particle filter, we use $N = 1000$ particles and $\omega = 0.8$ for perturbation, the value for which we determined via grid search.

For decision-making purposes, the parameters α_{LCB} and α_{UCB} are both set to 0.8. The cost c_{GO} is chosen to be 150 units which, given our data-generating distribution, provides approximately equal proportions of profitable and unprofitable simulated cases[2]. Further, c_{OOB} is set to 0 to maintain a pure profit-based reward, and we instead rely on the heuristics at leaf nodes to penalize the agent for leaving the mapped region. Finally, $c_{\text{FLY}} = 0.01$ units.

Unlike the example in Section 3.7.1, which terminated at $T = 35$, a minimum of 100 readings are required before decision making is possible. This constraint was imposed as preliminary experiments indicated that the standard deviation of the belief is subject to volatility in the early stages of interaction with the environment (see Fig. 4.2). Since

[1] Codes available at `https://github.com/ben-j-barlow/geophy-min-ex`.

[2] This is necessary as an agent bias towards GO could artificially improve our results if we had more profitable cases than unprofitable. Similarly, a NO-GO bias and more unprofitable cases would do the same.

the aim of this study is to evaluate whether intelligent path planning can accelerate decision making, we impose an information gathering cut-off in terms of time. A maximum of 250 timesteps is used by forcing the agent to choose $a_{251} \in \{\text{GO,NO-GO}\}$ if a decision has not already been made previously.

For the intelligent agent, the *POMCPOW action selection* component uses 15,000 queries for each decision, with progressive widening parameters $k = 2$ and $\alpha = 0.3$ selected via grid search. Both the state-value $Q(h,a)$ and visit count $N(h,a)$ in Alg. 1 are initialized at 0. We use UCT trees with exploration parameter $c = 125$ for the MCTS *selection* step and the output of *POMCPOW action selection* after tree construction is decided by selecting $argmax_a Q(h,a)$.

Finally, we enable rigorous evaluation in the borderline cases by separating our synthetic cases $\{s_i\}_{i=1}^{150}$ into categories based on anomaly value $v(s_i)$. This categorisation is as follows

$$\text{Category}_i = \begin{cases} \text{Unprofitable (highly)} & \text{if Profit}(s_i) \leq -20, \\ \text{Unprofitable (borderline)} & \text{if } -20 < \text{Profit}(s_i) \leq 0, \\ \text{Profitable (borderline)} & \text{if } 0 < \text{Profit}(s_i) \leq 20, \\ \text{Profitable (highly)} & \text{if } 20 < \text{Profit}(s_i). \end{cases}$$

4.2 Evaluation metrics

At each timestep t, we evaluate the quality of the current belief's estimate using the mean absolute percentage error MAPE_t. Further, to evaluate the change in confidence as the particle filter converges to the true state, we use SDRatio_t; the ratio of the standard deviation σ_t at time t over the initial belief's standard deviation σ_0. They are defined as follows

$$\text{MAPE}_t = \frac{1}{150}\sum_{i=1}^{150}\left|\frac{\mu_i^{(t)} - v(s_i)}{v(s_i)}\right| \times 100, \qquad \text{SDRatio}_t = \frac{1}{150}\sum_{i=1}^{150}\frac{\sigma_t^{(i)}}{\sigma_0^{(i)}}, \tag{4.1}$$

where $\mu_i^{(t)}$ represents the anomaly value estimate of trial i at time t; $v(s_i)$ is the true anomaly value of the state corresponding to trial i; $\sigma_t^{(i)}$ is the standard deviation of the value estimate for trial i at time t; $\sigma_0^{(i)}$ is the initial standard deviation of the value estimate for trial i.

4.3 Results

4.3.1 Traditional agent

In Tab. 4.1, we observe the traditional agent has near-perfect correctness (see correctness logic in Tab. 3.1) on the *profitable (highly)* and *unprofitable (highly)* cases. Out of the 123 non-borderline cases, the approach yielded a correct GO/NO-GO decision in 122 instances, resulting in a 99.2% accuracy rate. Accuracy decreased to an average of 77.8% for the borderline cases; however, this is of lesser concern given the smaller monetary sums at stake. Given that this investigation focuses on the financial benefits of employing an intelligent agent for survey path planning, it is crucial to assess the financial performance: the traditional agent achieves a total profit of 5002 out of a possible 5110, resulting in a 97.9% utilization of available profit.

		Go	No-go	Total	Accuracy	Available profit	Actual profit
Profitable	Highly	67	1	68	98.5%	4967	4917
	Borderline	8	3	11	72.7%	143	97
Unprofitable	Borderline	3	13	16	81.3%	0	-12
	Highly	0	55	55	100.0%	0	0
					95.3%	5110	5002

Table 4.1: The results of 150 synthetic cases with the traditional agent following a grid-based path. A GO/NO-GO decision is made once the agent has completed their predefined path. The total accuracy of 95.3% is given by the proportion of correct decisions (143) over the total number of cases (150). See correctness logic in Tab. 3.1.

We observe the error and standard deviation ratio decrease rapidly during the first 300 timesteps in Figs. 4.1 and 4.2. However, after the traditional agent completes the north-south flight lines, there is little change in the agent's belief. This provides evidence that our suggestion of responding to observations in real-time by terminating an airborne survey has value, rather than completing an entire survey in full.

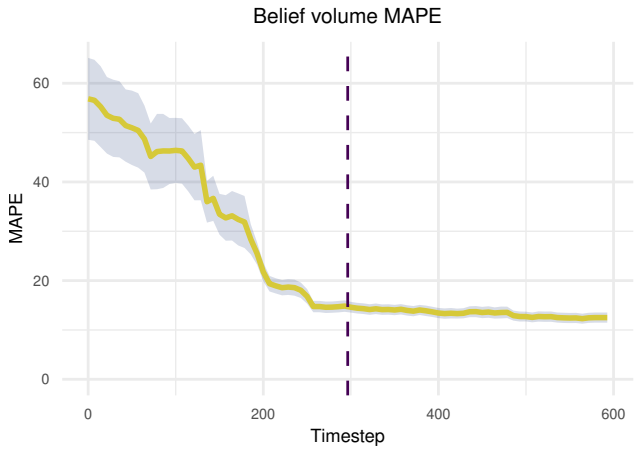


Figure 4.1: The MAPE_t as defined in Equation 4.1, for the traditional agent. One standard error of the average is shown by the shaded region and the vertical line shows when north-south flying lines are complete and east-west flying lines commence.

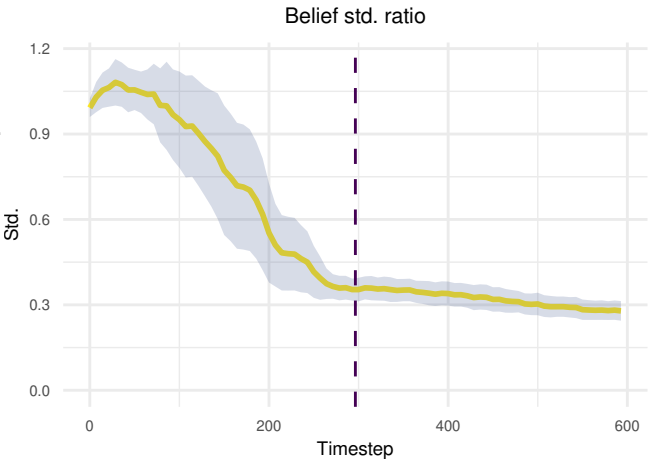


Figure 4.2: The SDRatio_t as defined in Equation 4.1 for the traditional agent. One standard error of the average is shown by the shaded region and the vertical line shows when north-south flying lines are complete and east-west flying lines commence.

4.3.2 Intelligent agent

Figure 4.3 provides an example of the intelligent agent performing as intended. The agent locates the anomaly in a timely manner and circles it until its uncertainty is sufficiently reduced to $\sigma_v = 16.2$ at $T = 197$. This level of confidence is similar to that consistently achieved by the traditional agent (see Fig. 4.4), but with 16.12 km less distance flown (403 fewer timesteps). Furthermore, the value estimate of 162.9 produces an error of 0.9 units. The agent attained a mean absolute error of less than 1 unit in 3 additional cases, totalling 4—only 2 fewer than the traditional agent.

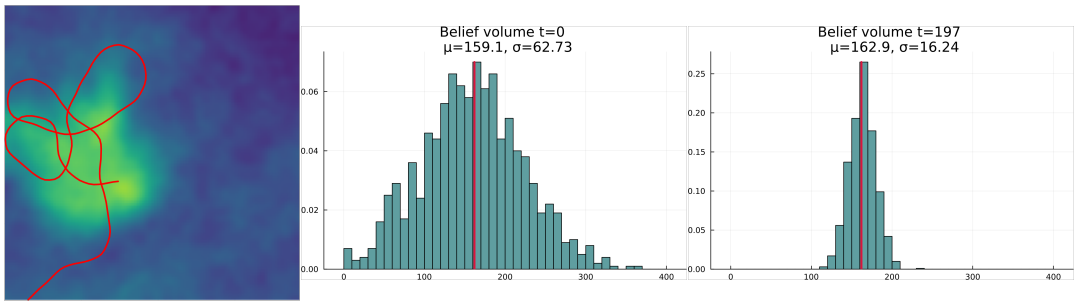


Figure 4.3: An example case where the intelligent agent performs as desired. The geophysical map with agent's trajectory (left) shows the agent circling the geophysical anomaly, thereby encouraging the particle filter's distribution at $t = 0$ (centre) to converge to the true value of 162 units at $T = 197$ (right) before the agent makes a correct decision with $a_T =$ GO.

However, the agent did not behave in this manner consistently across all 150 cases, as evidenced by the average accuracy of 68.7% in Tab. 4.2. The agent utilizes 54.0% of the available profit, influenced heavily by obtaining negative profit in unprofitable cases due to incorrect GO decisions.

		Go	No-go	Total	Accuracy	Available profit	Actual profit
Profitable	Highly	44	24	68	64.7%	4967	3449
	Borderline	5	6	11	45.5%	143	69
Unprofitable	Borderline	5	11	16	68.8%	0	-28
	Highly	12	43	55	78.2%	0	-733
					68.7%	5110	2757

Table 4.2: The results of 150 synthetic cases with the intelligent agent adopting an adaptive path during exploration. The total accuracy of 68.7% is given by the proportion of correct decisions (103) over the total number of cases (150). See logic in Tab. 3.1.

The large proportion of incorrect decisions in the profitable (highly) and unprofitable (highly) cases prompt further investigation. In Fig. 4.4, we observe the standard deviation of the final belief is larger for the intelligent agent. However, minimizing uncertainty is not the agent's ultimate aim, it is a tool in the wider decision making process. Of more urgency than the standard deviation alone is the proportion of incorrect decisions that this correlates with (the dark shade corresponds to incorrect decisions).

The possible causes for termination are: the stop criterion is met or the agent reaches the maximum timestep of 250. In Fig. 4.5, we observe the stop criterion

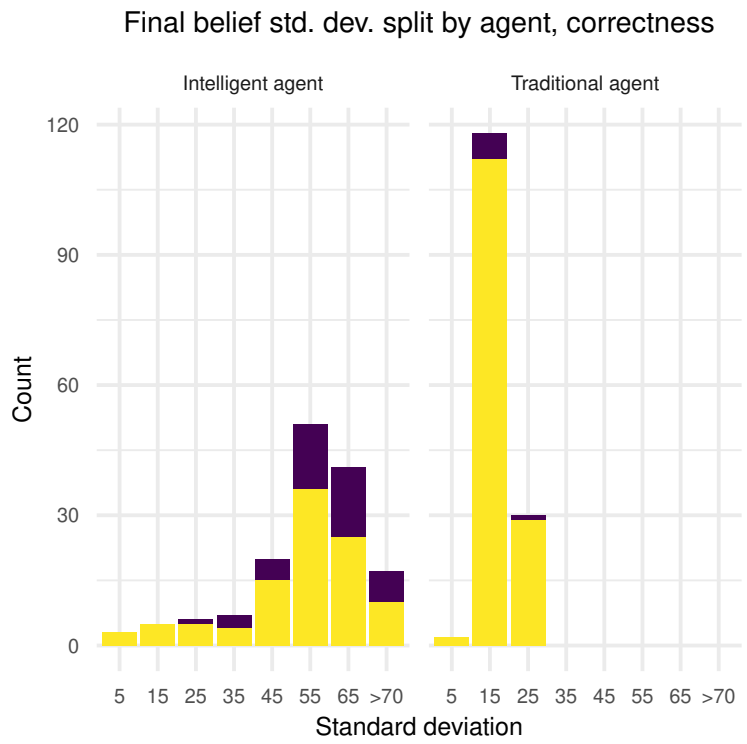


Figure 4.4: The final belief standard deviation split by agent and correctness. The yellow and dark shades correspond to a correct and incorrect GO/NO-GO decision, respectively.

and max timestep are the causes for termination in approximately equal proportions. We visualize the agent's path in an example of each case in Figs. 4.6 *(a)* & *(c)*. The premature termination issue in Fig. 4.6 *(c)* can be fixed easily by changing values α_U and α_L to construct a more restrictive criterion. We make suggestions for the issues presented by Figs. 4.6 *(a)* & *(b)* in Section 5.2.

4.4 Financial analysis

Given the cost of $c_{FLY} = 0.01$ and $v = 40$ ms^{-1}, we have the cost per km of survey flying to be $c_{KM} = 0.025$. The traditional agent flew for 600 timesteps in all 150 simulated cases, which totals 3600 km with 24 km per simulated case. The intelligent agent flew a total of 1114.9 km at an average of 7.4 km (185.8 timesteps) per simulated case. The financial summary in Tab. 4.3 shows that despite the intelligent agent flying a significantly smaller distance, its decision making causes a reduction in profit compared to the traditional agent.

However, it is important to interpret our results in the context of the real world. Geophysical surveys are conducted with fixed budgets. Assume a fixed budget of $c_{SURVEY} = 270$ units to conduct a survey with the goal of achieving maximal regional coverage and maximizing profit. Each case i has a profit Profit(s_i) and a cost $c_{FLY} \times n_i$,

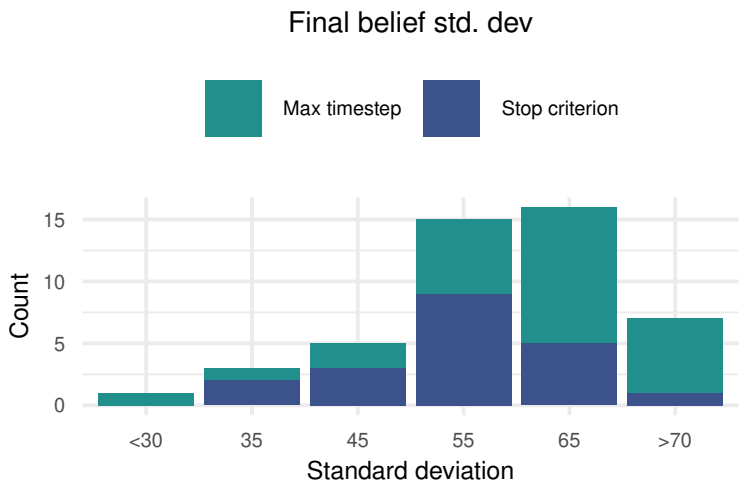


Figure 4.5: Count of incorrect decisions made by the intelligent agent split by termination type and standard deviation at time of termination. "Max timestep" corresponds to $T = 250$ being reached and "Stop criterion" corresponds to Eq. 3.7 being satisfied while $t < 250$.

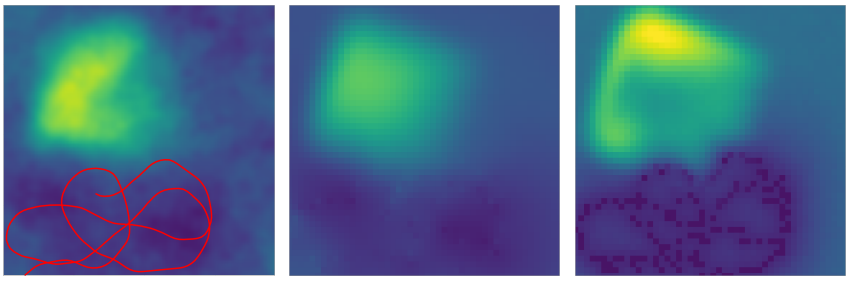
where n_i is the number of flying actions the agent takes before termination. By sampling without replacement from our population of 150 cases until the budget is fully expended, deducting $c_{\text{FLY}} \times n_i$ from the remaining budget with each sample i, we find that the intelligent agent would produce almost double the profit than the traditional agent in the real world (see Tab. 4.4).

	Traditional agent	Intelligent agent
Profit (GO)	5002	2757
Cost (Survey)	-900	-278.7
Total profit	4102	2478.3

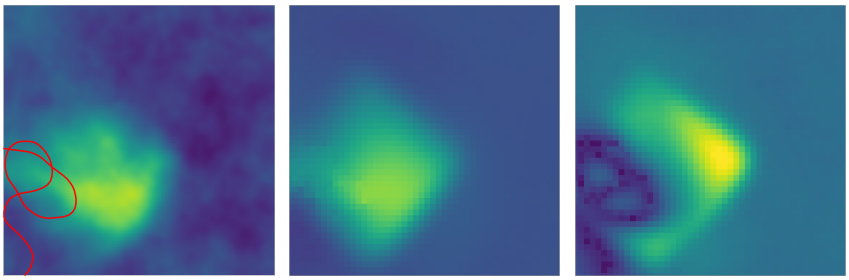
Table 4.3: A profit analysis of the 150 cases that we performed experiments on.

	Traditional agent	Intelligent agent
Profit (GO)	$1491.8_{\pm 7.7}$	$2661.1_{\pm 3.4}$
Cost (Survey)	-270	-270
Total profit	$1221.8_{\pm 7.7}$	$2391.1_{\pm 3.4}$

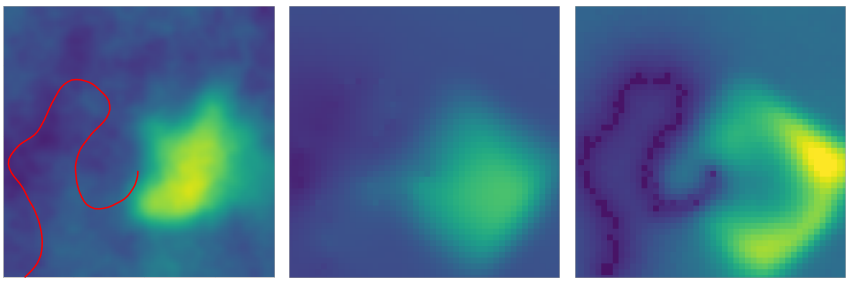
Table 4.4: An analysis of the agent's expected profit when restricted to a survey budget of $c_{\text{SURVEY}} = 270$ units. Samples are drawn without replacement from the 150 cases and the flying cost $c_{\text{FLY}} \times n$, where n is the number of flying actions before termination, is deducted from the remaining budget. Once the budget reaches 0, the sum of the profit in the sample produces Profit (GO). We used 1000 iterations of this procedure which enabled standard errors shown in the table.



(a) The intelligent agent flies in circles around a region that does not correspond to the geophysical anomaly, eventually being terminated by the time cutoff. This motivates improvements to the agent's learning capabilities to encourage a more direct route to the anomaly.



(b) The intelligent agent suffers from its inability to plan beyond 5 timesteps. Our heuristic $H(s)$ in Equation 3.3 uses an indicator function to encourage the agent to remain in the mapped area. However, when the agent approaches a map boundary with a heading perpendicular to the boundary, all possible sequences $a_t, \ldots, a_{t+4}$ result in the agent leaving the mapped area. Hence, $H(s_{t+4}) = 0$ for all states s_{t+4} and the agent cannot learn to remain on the map. Whereas, earlier on in the simulation when the agent approaches the boundary at a reduced angle, there exists $a_t, \ldots, a_{t+4}$ such that $H(s_{t+4}) > 0$ and the agent avoids crossing the boundary through planning.



(c) The intelligent agent is on the border of the region containing geophysical anomalies, however, the stop criterion is not restrictive enough and the agent terminates interaction with the environment despite a large standard deviation $\sigma_T = 62.6$. The agent makes the incorrect decision with belief mean estimate 98 and true value 189 units. This case could be solved by forcing a threshold maximum value on the standard deviation before termination.

Figure 4.6: A selection of examples where undesired behaviour from the intelligent agent is observed. Geophysical map with agent trajectory (left), belief mean (centre), and belief standard deviation (right).

Chapter 5

Conclusions & Discussion

5.1 Conclusion

Given millions of kilometres of airborne geophysical surveys are flown for mineral exploration (see Section 2.1), even a 1% reduction in flight distances could yield substantial financial savings. This dissertation tests this hypothesis by comparing a traditional agent using a grid-based flight path with an adaptive agent that adjusts its path based on observations in real time. The traditional agent flies 24 km to survey a 2.4 km^2 area, while the adaptive agent is limited to 10 km[1]—just 41.7% of the traditional distance—thereby pushing our hypothesis to the extreme. Our results suggest that intelligently chosen flight paths can produce accurate and confident estimates of geophysical anomaly magnitude, though further refinement (see Section 5.2) is necessary to facilitate consistent performance in terms of GO/NO-GO decisions.

Approaches to information gathering problems typically focus on minimizing spatial uncertainty and rely on preset time limits or uncertainty thresholds for termination. We instead propose a geophysical data acquisition POMDP that explicitly accounts for the final drill-or-abandon decision when planning paths for data acquisition; a design choice that ensures our approach aligns with the real-world objectives of geophysical surveys. Our path planning solution leverages the benefits of nonmyopic planning, and due to advances in POMDP-related algorithms, can do so with reasonable computational effort.

Despite the agent demonstrating it can perform desirably in some cases, the *POMCPOW action selection* component requires attention. Adaptions to the reward function and search tree construction (discussed in Section 5.2) would enable the agent to per-

[1]The intelligent agent is limited to a maximum of 10 km per survey but uses 7.432 km on average.

form consistently over variations in anomaly shape, orientation, and magnitude. The *geophysical environment* system component is reflective of real-world airborne geophysical surveys by implementing continuous flying paths based on an aircraft's bank angle and velocity. Observations comprise a derivative signal and embody geophysical noise stemming from other minerals in the subsurface as well as sensor noise. Finally, the *belief update* uses particle filter approximation to seamlessly integrate all previous actions and observations into the current belief of the world. We successfully balance particle filter convergence by perturbing anomalies at each timestep with perturbation magnitude governed by parameter ω.

5.2 Discussion and future work

The agent's performance in terms of GO/NO-GO decision correctness is dependent on accurately and confidently quantifying the state anomaly value $v(s)$. Fig. 4.3 shows the system is capable of obtaining such estimates when a highly informative path is followed. But, changes are necessary to improve the agent's consistency in selecting such a path, as evidenced by Figs. 4.6 *(a-c)*. We visualize the contrast between desirable and observed behaviour in Figs. 5.1 *(b)* and *(c)*.

Addressing this issue requires modifying the agent's learning mechanism. Currently, the agent learns through profit-based rewards and heuristics. A belief MDP (Kaelbling et al., 1998) reformulates POMDPs by making the reward $R(b,a)$ a function of belief b rather than rewards $R(s,a)$ or $R(s,a,s')$ based on states. This change is powerful since it permits access to state uncertainty in the reward function. Inspired by the approach in Cao et al. (2023), who balance exploration of "high-interest areas" (in our case, geophysical anomalies) with uncertainty reduction for adaptive informative path planning, we could use an upper confidence bound reward

$$R(b,a) = \mu(x_{ij}) + c\sigma(x_{ij}), \tag{5.1}$$

where $\mu(x_{ij})$ and $\sigma(x_{ij})$ respectively represent the current mean estimate $v(s)$ and uncertainty in that estimate at grid cell x_{ij}; and c is a parameter to balance exploration and exploitation. This approach aligns the agent's path selection with the goal of identifying areas with high mineral prospectivity, rather than merely pursuing uncertainty reduction, as typical of many information gathering agents. We hypothesize this reward would incentivize the agent to follow the path shown in Fig. 5.1 *(b)*. Furthermore,

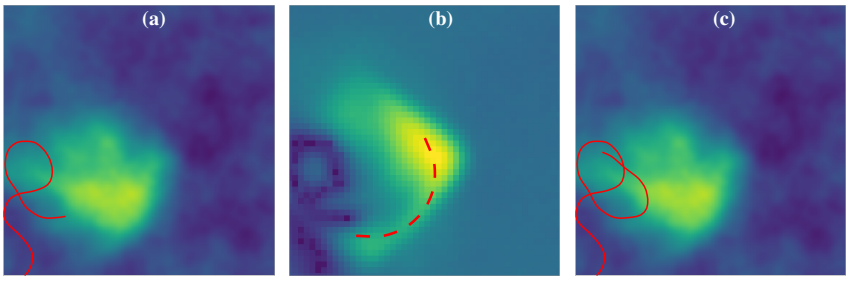


Figure 5.1: The agent's trajectory after 70 timesteps plotted over the geophysical map *(a)* shows it is approaching the anomaly. The desired behaviour is for the agent to circle the geophysical anomaly by taking the path over the area of high uncertainty shown by the dotted red line plotted over the belief standard deviation *(b)*. Instead, the agent's path after 90 timesteps *(c)* shows it chooses to behave undesirably.

this formulation would permit relaxing our assumption of exactly one significant geophysical anomaly per 2400 m^2 region; which would further align our approach with the real world in which significant geophysical anomalies can occur at various spatial frequencies.

A change in model formulation to a belief MDP necessitates a switch to belief-space planning (instead of planning over state trajectories). The computationally challenging nature of belief-space planning, which performs a belief update between each layer of the tree, limits tractable planning horizons considerably. This sparked the pivot of popular tree-based algorithms DESPOT (Ye et al., 2016) and POMCPOW (Sunberg and Kochenderfer, 2018) towards state trajectory based trees, inspired by POMCP (Silver and Veness, 2010). We suggest employing the recent advancement *BetaZero* (Moss et al., 2024); an algorithm that enables online belief-space planning in long-horizon problems by learning offline neural network approximations of the optimal policy.

Rather than using state uncertainty at a grid cell level as in Equation 5.1, we could reward for uncertainty reduction in the estimate for $v(s)$. Another reformulation of POMDPs, namely $\rho-$POMDPs (Mauricio Araya et al., 2010), are defined such that the reward ρ is associated with transitioning from belief b to belief b' after taking action a. We could implement a reward such that

$$\rho(b,a,b') = \frac{\sigma(b) - \sigma(b')}{\sigma(b)} \tag{5.2}$$

where $\sigma(b)$ is the uncertainty in belief b's current estimate for $v(s)$. A candidate algorithm for solving such a formulation is $\rho-$POMCP (Thomas et al., 2020); an extension of POMCP to ρ-POMDPs.

Bibliography

Åström, K. (1965, 2). Optimal control of Markov processes with incomplete state information. *Journal of Mathematical Analysis and Applications 10*(1), 174–205.

Astuti, G., G. Giudice, D. Longo, C. D. Melita, A. Orlando, and G. Muscato (2009). An overview of the "volcan project": An UAS for exploration of volcanic environments. In K. P. Valavanis, P. Oh, and L. A. Piegl (Eds.), *Unmanned Aircraft Systems: International Symposium On Unmanned Aerial Vehicles, UAV'08*, Dordrecht, pp. 471–494. Springer Netherlands.

Atallah, L., B. Lo, R. King, and G.-Z. Yang (2010, 6). Sensor Placement for Activity Detection Using Wearable Accelerometers. In *2010 International Conference on Body Sensor Networks*, pp. 24–29. IEEE.

Bai, S., T. Shan, F. Chen, L. Liu, and B. Englot (2021, 4). Information-Driven Path Planning. *Current Robotics Reports 2*(2), 177–188.

Barfoot, T. D. (2017, 7). *State Estimation for Robotics*. Cambridge University Press.

Bar-Shalom, Y., X. Li, and T. Kirubarajan (2002, 1). *Estimation with Applications to Tracking and Navigation*. Wiley.

Bergholm, F. (1987, 11). Edge Focusing. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-9*(6), 726–741.

Browne, C. B., E. Powley, D. Whitehouse, S. M. Lucas, P. I. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakis, and S. Colton (2012, 3). A Survey of Monte Carlo Tree Search Methods. *IEEE Transactions on Computational Intelligence and AI in Games 4*(1), 1–43.

Campbell, G. A. (2014, 6). Rare earth metals: a strategic concern. *Mineral Economics 27*(1), 21–31.

Cao, Y., Y. Wang, A. Vashisth, H. Fan, and G. A. Sartoretti (2023). CAtNIPP: Context-Aware Attention-based Network for Informative Path Planning. In *Proceedings of The 6th Conference on Robot Learning*, pp. 1928–1937.

Caselton, W. F. and J. V. Zidek (1984, 8). Optimal monitoring network designs. *Statistics & Probability Letters 2*(4), 223–227.

Chen, J. and J. C. Macnae (1997). Terrain corrections are critical for airborne gravity gradiometer data. *Exploration Geophysics 28*(2), 21–25.

Corso, A., Y. Wang, M. Zechner, J. Caers, and M. J. Kochenderfer (2022, 10). A POMDP Model for Safe Geological Carbon Sequestration.

Couëtoux, A., J.-B. Hoock, N. Sokolovska, O. Teytaud, and N. Bonnard (2011). Continuous Upper Confidence Trees. In C. Coello (Ed.), *Learning and Intelligent Optimization*, pp. 433–445. Berlin, Heidelberg: Springer.

Coulom, R. (2007). Efficient Selectivity and Backup Operators in Monte-Carlo Tree Search. In *International conference on computers and games*, pp. 72–83. Springer.

Cox, S. F. (2005, 1). Coupling between Deformation, Fluid Pressures, and Fluid Flow in Ore-Producing Hydrothermal Systems at Depth in the Crust. *Geoscience World*.

Davies, R. S., M. J. Davies, D. Groves, K. Davids, E. Brymer, A. Trench, J. P. Sykes, and M. Dentith (2021, 9). Learning and expertise in mineral exploration decision-making: An ecological dynamics perspective.

Dransfield, M. (2007). Airborne gravity gradiometry in the search for mineral deposits. In *Proceedings of exploration*, pp. 341–354.

Dransfield, M., A. Christensen, M. Rose, P. Stone, and P. Diorio (2001, 9). Falcon Test Results from the Bathurst Mining Camp. *Exploration Geophysics 32*(3-4), 243–246.

Dunbabin, M. and L. Marques (2012, 3). Robots for environmental monitoring: Significant advancements and applications. *IEEE Robotics and Automation Magazine 19*(1), 24–39.

Hauskrecht, M. (2000, 8). Value-Function Approximations for Partially Observable Markov Decision Processes. *Journal of Artificial Intelligence Research 13*, 33–94.

Hinze, W. J., R. R. Von Frese, and A. H. Saad (2013). *Gravity and magnetic exploration: Principles, practices, and applications*. Cambridge University Press.

Hoffimann, J. (2018, 4). GeoStats.jl – High-performance geostatistics in Julia. *Journal of Open Source Software 3*(24), 692.

Hoffmann, G. M. and C. J. Tomlin (2010, 1). Mobile Sensor Network Control Using Mutual Information Methods and Particle Filters. *IEEE Transactions on Automatic Control 55*(1), 32–47.

Joshi, S. and S. Boyd (2009, 2). Sensor Selection via Convex Optimization. *IEEE Transactions on Signal Processing 57*(2), 451–462.

Journel, A. and T. Zhang (2006, 7). The Necessity of a Multiple-Point Prior Model. *Mathematical Geology 38*(5), 591–610.

Julian, K. D. and M. J. Kochenderfer (2020, 10). Image-based Guidance of Autonomous Aircraft for Wildfire Surveillance and Prediction. In *2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC)*, pp. 1–8. IEEE.

Jung, B. K., J. R. Cho, and W. B. Jeong (2015, 7). Sensor placement optimization for structural modal identification of flexible structures using genetic algorithm. *Journal of Mechanical Science and Technology 29*(7), 2775–2783.

Kaelbling, L. P., M. L. Littman, and A. R. Cassandra (1998, 5). Planning and acting in partially observable stochastic domains. *Artificial Intelligence 101*(1-2), 99–134.

Kass, M. A. and Y. Li (2008, 12). Practical aspects of terrain correction in airborne gravity gradiometry surveys. *Exploration Geophysics 39*(4), 198–203.

Kebede, B. and T. Mammo (2021, 4). Processing and interpretation of full tensor gravity anomalies of Southern Main Ethiopian Rift. *Heliyon 7*(4), e06872.

Kochenderfer, M. J., T. A. Wheeler, and K. H. Wray (2022). *Algorithms for decision making*. MIT press.

Krause, A., A. Singh, and C. Guestrin (2008). Near-Optimal Sensor Placements in Gaussian Processes: Theory, Efficient Algorithms and Empirical Studies. *Journal of Machine Learning Research 9*, 235–284.

Lauri, M., D. Hsu, and J. Pajarinen (2022, 9). Partially Observable Markov Decision Processes in Robotics: A Survey. *IEEE Transactions on Robotics 39*(1), 21–40.

Li, Y. and D. W. Oldenburg (2000, 1). Incorporating geological dip information into geophysical inversions. *GEOPHYSICS 65*(1), 148–157.

Lim, Z. W., D. Hsu, and W. S. Lee (2016, 4). Adaptive informative path planning in metric spaces. *The International Journal of Robotics Research 35*(5), 585–598.

Littman, M. L., A. R. Cassandra, and L. P. Kaelbling (1995). Learning policies for partially observable environments: Scaling up. In *Machine Learning Proceedings 1995*, pp. 362–370. Elsevier.

Lovejoy, W. S. (1991, 12). A survey of algorithmic methods for partially observed Markov decision processes. *Annals of Operations Research 28*(1), 47–65.

Lyatsky, H. (2010). Magnetic and gravity methods in mineral exploration: The value of well-rounded geophysical skills. *Recorder (Canadian Society of Exploration Geophysics)*, 30–35.

Mariethoz, G. and J. Caers (2014). *Multiple-point geostatistics: stochastic modeling with training images*. John Wiley & Sons.

Mauricio Araya, O. Buffet, V. Thomas, and F. Charpillet (2010). A POMDP extension with belief-dependent rewards. *Advances in neural information processing systems 23*.

Maybeck, P. S. (1982). *Stochastic models, estimation, and control*. Academic press.

Meliou, A., A. Krause, C. Guestrin, and J. M. Hellerstein (2007). Nonmyopic informative path planning in spatio-temporal models. In *Association for Advancement of Artificial Intelligence (AAAI)*, pp. 602–607.

Mern, J. and J. Caers (2023, 1). The Intelligent Prospector v1.0: geoscientific model development and prediction by sequential data acquisition planning with application to mineral exploration. *Geoscientific Model Development 16*(1), 289–313.

Mern, J., A. Yildiz, Z. Sunberg, T. Mukerji, and M. J. Kochenderfer (2021, 5). Bayesian Optimized Monte Carlo Planning. *Proceedings of the AAAI Conference on Artificial Intelligence 35*(13), 11880–11887.

Monahan, G. E. (1982, 1). State of the Art—A Survey of Partially Observable Markov Decision Processes: Theory, Models, and Algorithms. *Management Science 28*(1), 1–16.

Moss, R. J., A. Corso, J. Caers, and M. J. Kochenderfer (2024). BetaZero: Belief-State Planning for Long-Horizon POMDPs using Learned Approximations. *RLJ 1*(1).

Mudd, G. M. and S. M. Jowitt (2014, 11). A Detailed Assessment of Global Nickel Resource Trends and Endowments. *Economic Geology 109*(7), 1813–1841.

Oldenburg, D. and D. Pratt (2007). Geophysical inversion for mineral exploration: A decade of progress in theory and practice. In *Proceedings of exploration*, pp. 61–95.

Olierook, H. K. H., R. Scalzo, D. Kohn, R. Chandra, E. Farahbakhsh, C. Clark, S. M. Reddy, and R. D. Müller (2020). Bayesian geological and geophysical data fusion for the construction and uncertainty quantification of 3D geological models.

Ott, J., E. Balaban, and M. J. Kochenderfer (2022, 9). Sequential Bayesian Optimization for Adaptive Informative Path Planning with Multimodal Sensing. *Proceedings - IEEE International Conference on Robotics and Automation 2023-May*, 7894–7901.

Pei, X.-Y., T.-H. Yi, C.-X. Qu, and H.-N. Li (2019, 6). Conditional information entropy based sensor placement method considering separated model error and measurement noise. *Journal of Sound and Vibration 449*, 389–404.

Ross, S., J. Pineau, S. Paquet, and B. Chaib-draa (2008, 7). Online Planning Algorithms for POMDPs. *Journal of Artificial Intelligence Research 32*, 663–704.

Shahriari, B., K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas (2016, 1). Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proceedings of the IEEE 104*(1), 148–175.

Shani, G., J. Pineau, and R. Kaplow (2013, 7). A survey of point-based POMDP solvers. *Autonomous Agents and Multi-Agent Systems 27*(1), 1–51.

Silver, D. and J. Veness (2010). Monte-Carlo planning in large POMDPs. *Advances in neural information processing systems 23*.

Singh, A., A. Krause, and W. J. Kaiser (2009). Nonmyopic Adaptive Informative Path Planning for Multiple Robots. Technical report, UCLA: Center for Embedded Network Sensing.

Sunberg, Z. N. and M. J. Kochenderfer (2018, 6). Online Algorithms for POMDPs with Continuous State, Action, and Observation Spaces. *Proceedings of the International Conference on Automated Planning and Scheduling 28*, 259–263.

Sutton, R. S. and A. G. Barto (2018). *Reinforcement learning: An introduction*. MIT press.

Techy, L., D. A. Paley, and C. A. Woolsey (2009, 8). UAV coordination on convex curves in wind: An environmental sampling application. In *2009 European Control Conference (ECC)*, pp. 4967–4972. IEEE.

Thomas, V., G. Hutin, and O. Buffet (2020). Monte Carlo Information-Oriented Planning. In *Frontiers in Artificial Intelligence and Applications*, pp. 2378–2385. IOS Press.

Thrun, S. (2002). Probabilistic robotics. *Communications of the ACM 45*(3), 52–57.

Trendafilova, I., W. Heylen, and H. V. Brussel (2001, 6). Measurement point selection in damage detection using the mutual information concept. *Smart Materials and Structures 10*(3), 528–533.

Tryggvason, B., B. Main, and B. French (2004). A high resolution airborne gravimeter and airborne gravity gradiometer. *Airborne Gravity*, 41–48.

Turner, J. M. (2022, 6). The matter of a clean energy future. *Science 376*(6600), 1361.

Wang, Y., M. Zechner, J. M. Mern, M. J. Kochenderfer, and J. K. Caers (2022, 8). A sequential decision-making framework with uncertainty quantification for groundwater management. *Advances in Water Resources 166*, 104266.

White, C. C. (1991, 12). A survey of solution techniques for the partially observed Markov decision process. *Annals of Operations Research 32*(1), 215–230.

Williams, C. and C. Rasmussen (1995). Gaussian processes for regression. *Advances in neural information processing systems 8*.

Ye, N., A. Somani, D. Hsu, and W. S. Lee (2016, 9). DESPOT: Online POMDP Planning with Regularization. *Journal of Artificial Intelligence Research 58*, 231–266.

Yildiz, A., J. Mern, M. J. Kochenderfer, and M. F. Howland (2023, 11). Towards sequential sensor placements on a wind farm to maximize lifetime energy and profit. *Renewable Energy 216*, 119040.

Yin, Z., C. Zuo, E. J. MacKie, and J. Caers (2022, 2). Mapping high-resolution basal topography of West Antarctica from radar data using non-stationary multiple-point geostatistics (MPS-BedMappingV1). *Geoscientific Model Development 15*(4), 1477–1497.

Zhang, B. and G. S. Sukhatme (2007, 4). Adaptive Sampling for Estimating a Scalar Field using a Robotic Boat and a Sensor Network. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 3673–3680. IEEE.

Zhang, Q., H. Wu, X. Mei, D. Han, M. D. Marino, K. C. Li, and S. Guo (2023, 11). A Sparse Sensor Placement Strategy Based on Information Entropy and Data Reconstruction for Ocean Monitoring. *IEEE Internet of Things Journal 10*(22), 19681–19694.

Appendix A

Gaussian processes

Gaussian processes[1] (GPs; Williams and Rasmussen 1995) are probabilistic surrogate models that represent distributions over functions. Suppose we have a set of input points $\mathbf{X} = [x^1, \ldots, x^n]$ and their corresponding outputs $\mathbf{y} = [y^1, \ldots, y^n]^\top$. A Gaussian process can predict the values $\hat{\mathbf{y}}$ at a new set of input points $\mathbf{X}^*$ as follows:

$$\begin{bmatrix} \hat{\mathbf{y}} \\ \mathbf{y} \end{bmatrix} \sim N\left(\begin{bmatrix} \mathbf{m}(\mathbf{X}^*) \\ \mathbf{m}(\mathbf{X}) \end{bmatrix}, \begin{bmatrix} \mathbf{K}(\mathbf{X}^*, \mathbf{X}^*) & \mathbf{K}(\mathbf{X}^*, \mathbf{X}) \\ \mathbf{K}(\mathbf{X}, \mathbf{X}^*) & \mathbf{K}(\mathbf{X}, \mathbf{X}) \end{bmatrix} \right),$$

where

$$\mathbf{m}(\mathbf{X}) = \begin{bmatrix} m(x^1) \\ \vdots \\ m(x^n) \end{bmatrix},$$

and the covariance matrix $\mathbf{K}$ is given by:

$$\mathbf{K}(\mathbf{X}, \mathbf{X}') = \begin{bmatrix} k(x^1, x'^1) & \cdots & k(x^1, x'^j) \\ \vdots & \ddots & \vdots \\ k(x^n, x'^1) & \cdots & k(x^n, x'^j) \end{bmatrix}.$$

To sample from the posterior distribution of a GP, we use the following conditional distribution:

$$\hat{y} \mid y \sim N(\mu^*, \sigma^*),$$

where the mean μ^* and variance σ^* are calculated as:

[1] We credit Yildiz et al. (2023) for the discussion here.

$$\mu^* = \mathbf{m}(\mathbf{X}^*) + \mathbf{K}(\mathbf{X}^*, \mathbf{X})\mathbf{K}(\mathbf{X}, \mathbf{X} + \Sigma)^{-1}(\mathbf{y} - \mathbf{m}(\mathbf{X})),$$

$$\sigma^* = \mathbf{K}(\mathbf{X}^*, \mathbf{X}^*) - \mathbf{K}(\mathbf{X}^*, \mathbf{X})\mathbf{K}(\mathbf{X}, \mathbf{X} + \Sigma)^{-1}\mathbf{K}(\mathbf{X}, \mathbf{X}^*).$$

Appendix B

Spherical variogram

A spherical variogram is a spatial model used to describe the correlation between data points as a function of their separation distance. The distance h between a pair of points gives rise to the semivariance $\gamma(h)$ as follows

$$\gamma(h) = (s-n)\left[\left(\frac{3}{2}\left(\frac{h}{r}\right)+\frac{1}{2}\left(\frac{h}{r}\right)^3\right)\cdot \mathbf{1}\{h \geq r\} + \mathbf{1}\{h > 0\}\right] + n \cdot \mathbf{1}\{h > 0\},$$

where $\mathbf{1}$ is the indicator function. The sill s represents the maximum value that the variogram reaches as the distance between two points increases. The nugget corresponds to the semivariance when $h = 0$, often attributed to measurement errors. Finally, the range r is the distance at which the spatial correlation becomes negligible, meaning the semivariance $\gamma(h)$ reaches the sill.

The behaviour of the function is such that for h values less than or equal to the range r, the semivariance $\gamma(h)$ increases with distance according to the spherical model, reflecting the gradual decrease in spatial correlation. As h approaches r, the semivariance nears the sill s, indicating that the spatial correlation becomes weaker. For h values greater than the range r, the semivariance $\gamma(h)$ remains constant at the sill s, showing that the spatial correlation between points beyond this distance is negligible, and the points are effectively uncorrelated.

Appendix C

Map colour scale

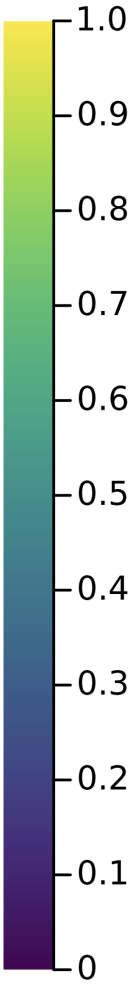


Figure C.1: The colour bar for all map plots in this report.

Appendix D

POMCPOW Example Tree

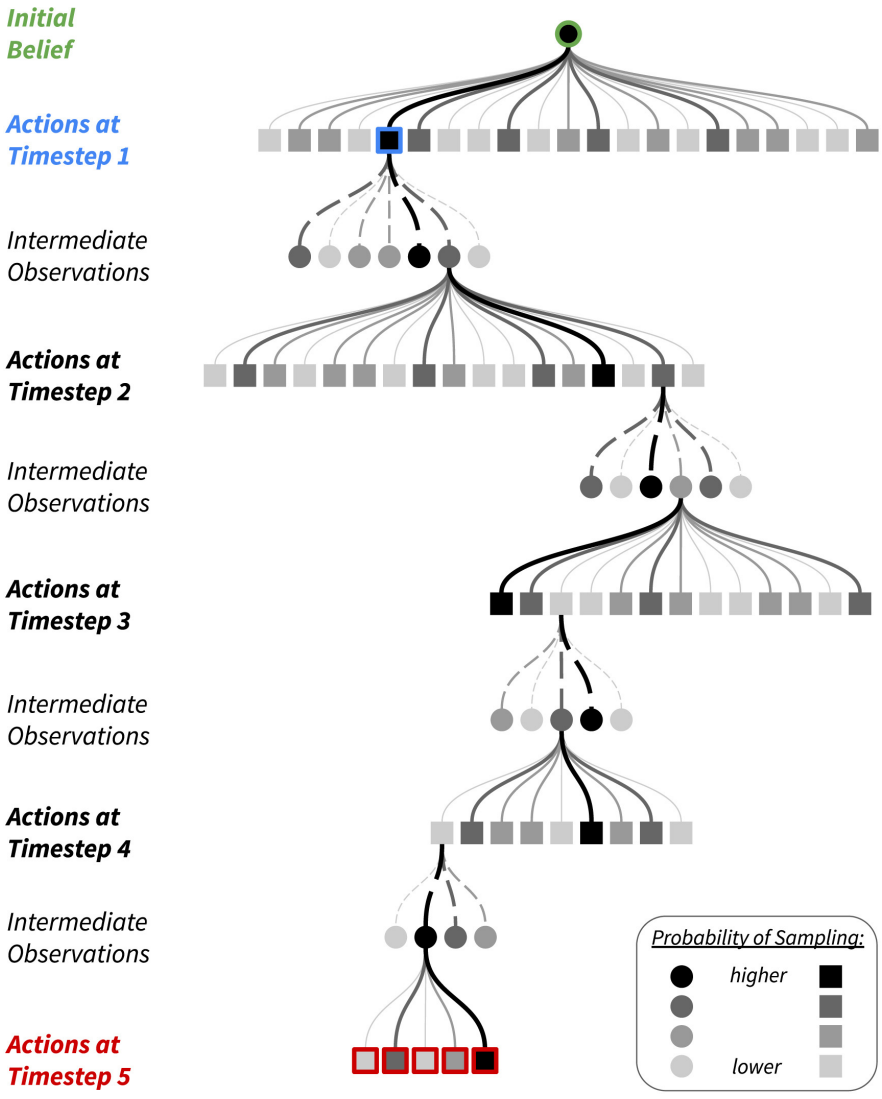


Figure D.1: A POMCPOW tree. Image credit: Fig. 5, Yildiz et al. (2023).