Optimality and Application of Tree Search Methods for POMDP-based Sensor Tasking

Samuel J. Fedeler[*], **Marcus J. Holzinger**[†]
University of Colorado at Boulder
William Whitacre
Draper Laboratory

September 16, 2020

ABSTRACT

A critical problem in modern Space Situational Awareness is that of tasking a portfolio of sensors to track an increasingly large catalog of satellites. As this set of space objects grows over time, telescopes and RF receivers must be directed to maximize information gain. This research focuses on the problem of catalog maintenance, and Monte Carlo Tree Search methods are leveraged to solve this problem over a receding, finite time horizon. A proof by induction is demonstrated to bound error in the estimation of the value of tasking decisions. The tree search implementation is then presented for a large scale simulation, in which a set of ground and space-based observers is tasked in real time for observation of a set of 1000 objects over the course of a day. A measurement innovation-based greedy tasking algorithm is applied as a point of comparison. The Monte Carlo Tree Search methods are shown to outperform the greedy tasker. In addition, analysis is performed considering the extensibility of Monte Carlo Tree Search for embarrassingly parallel implementations, as well as a quantitative study of convergence toward an optimal solution.

1. INTRODUCTION

Determination of policies for a set of sensors tasked to maintain custody of space objects in various orbit regimes is increasingly relevant to Space Situational Awareness (SSA). As a result of accelerating growth in satellite populations, it is imperative that limited observational assets are utilized efficiently. The problem at hand quickly becomes combinatoric as the object catalog considered expands, and multiple competing objectives are often desired to leverage uncued detection of objects in addition to catalog maintenance. As such, the sensor tasking problem is largely broken into tractable subproblems, in which the objective is to capture a single aspect of the overarching goal. A variety of methodologies have previously been proposed for object discovery, typically with special focus on the geostationary (GEO) regime.

Often, when pure search for new space objects (SOs) is desired, methodologies such as striping are applied. This strategy is shown to be effective for GEO maintenance and search [1, 2] using either a targeted field through which SOs are allowed to drift or declination striping and multi-stripe raster scanning. While these strategies are useful for large populations, this problem can also be considered from the perspective of maintaining or further resolving known information on an existing catalog of SOs. Erwin et al. develop metrics for sensor tasking based on the Fisher information; Williams et al. extend this work, considering metrics of instability such as Lyapunov exponents [3, 4]. More recently, Frueh et al. [5] demonstrate the effectiveness of treating this problem as a local optimization rather than using a heuristic approach, adding an urgency function to ensure no SO remains unobserved. A variety of optimization techniques with bases from reinforcement learning [6, 7] to Dempster-Shafer theory [8] have also been utilized. Dynamic programming methods have previously been successful [9] utilizing a receding horizon approach.

This paper considers the subproblem of catalog maintenance, posing the multi-sensor tasking problem as a Partially Observable Markov Decision Process (POMDP) with the goal of determining time histories of optimal pointing for a given set of sensor locations and specifications. Monte Carlo Tree Search (MCTS) [10–15] approaches are considered given the broad action space and continuous observation and state spaces inherent to the sensor tasking problem. In previous work [16], a tree search methodology was developed treating belief in states as a set of Gaussian random variables. While many MCTS methods represent belief as a set of particles, this methodology allows for high dimensional states to be represented in the tree search format while avoiding the curse of dimensionality inherent to particle-based techniques.

[*]PhD Precandidate, Draper and NSF Graduate Research Fellow, Smead Department of Aerospace Engineering Sciences
[†]Associate Professor, H.J. Smead Faculty Fellow, Smead Department of Aerospace Engineering Sciences

Several contributions are submitted in this proposed work. Primarily, an induction-based proof is applied to demonstrate convergence and guarantees of the tree search methodology as a function of tree search iterations; this proof extends that of Auger et al. [14] in its application to the POMDP framework. Instead of a randomized, generative state transition model, the POMDP framework must use measurements and likelihood sampling in the traversal from sampled actions to associated observations. To accompany this derivation, large scale simulations are performed to numerically evaluate algorithmic behavior over a variety of use cases. The tree search algorithm is applied to a large-scale, multi-observer use case studying a population of 1000 objects ia full day observation campaign.

The methodologies discussed are most impactful in that tree search techniques can be applied in an embarrassingly parallel manner, allowing for broad exploration of potential tasking decisions. There is also high potential for the use of tree search methods in situations where dynamic environments and measurement models must be considered. Given appropriate modeling, the MCTS methods discussed are easily augmented to incorporate false alarm and detection probabilities, weather and lighting conditions, and sensor uncertainties. As such, MCTS techniques offer addition and extension to information theoretic methodologies such as [3] and [4], allowing for broader understanding of the tasking problem.

A brief overview of MCTS will be given in Section II, outlining critical concepts to the methodology. Section III will proceed to describe the proof by induction demonstrating bounds in estimation error for the value of tasking actions. Finally, Section IV will discuss the application of MCTS to large scale simulation, demonstrating results as compared to greedy, information-theoretic tasking methods.

2. DECISION PROCESSES AND MONTE CARLO TREE SEARCH

We first present a brief review of critical concepts for Monte Carlo Tree Search. In addition to this review, further background can be gained from [10–12, 15, 17]. Generally, MCTS can be applied to any sequential decision process; there has been broad focus on application of MCTS game-theoretic problems such as Go, a strategic game with a massive set of potential board positions [17, 18]. The focus of this work narrows the problem context to partially-observable Markov decision processes (POMDPs), in which the underlying states in the system studied are not directly observable.

2.1 Partially-Observable Markov Decision Processes

A POMDP can be formally represented by the 7-tuple $(S, A, T, R, O, H, \Gamma)$ with the following definitions. The problem is defined over a state space S; this is a representation of a discrete or continuous space in which the studied system may evolve over $\Re^n$. Decisions are made over an action space A; again, this space may be either discrete or continuous, with dimension $\Re^p$. States evolve with transition probabilities $T : \Re^n \times \Re^p \to \Re^n$. Generally, T represents the propagation of both states and uncertainties over time, but actions taken may also impact the evolution of states. A reward $R : \Re^n \times \Re^p \to \Re^1$ applies an arbitrary objective function to determine the value of an associated change in state and action. The system is observed over the observation space O, with probabilities defined by H acting on the current state. As with the state and action spaces, O may be discrete or continuous over the domain $\Re^m$. $H : \Re^n \to \Re^m$ is simply a measurement function incorporating uncertainty in some manner. Immediate rewards are favored over distant rewards by the discount factor $\Gamma \in [0, 1]$.

The goal of a POMDP is the determination of an optimal policy π (sequence of actions $a_{1:N} \in A$) such that the discrete time Bellman equation V [19] is maximized given an initial belief in states b_0, where

$$V^{\pi}(b_0) = \sum_{t=0}^{\infty} \Gamma^t E[R(\mu_t, a_t)|b_0, \pi] \tag{1}$$

and μ_t is the state at time t and a_t is the associated action. The set of actions may terminate or occur over an infinite horizon. Note the differentiation between immediate reward R and the value function V that describes reward over a potentially infinite horizon. Traditional methods of solving MDPs such as value iteration [20] or grid-based algorithms become challenging in practice as observation and belief spaces become large. This led to the development of MCTS and other sample-based planning based methodologies.

2.2 Monte Carlo Tree Search

Monte Carlo evaluation allows random actions to be simulated until a valuable result is reached. The following concepts are critical to the understanding of the MCTS methodology. Algorithm 1 can be used as a reference as

concepts are discussed.

1. **Nodes** - Following general data structures terminology we refer to an arbitrary index in the search tree as a node. The search tree is initialized by a root node. Any node may have zero to many child nodes, and a node with no children is referred to as a leaf node. Other than the root node, any node must have a parent node. We also differentiate between action nodes, where a new action is sampled, and observation nodes, where an observation is generated and associated with an action.

2. **Rollout-based planning** - Generally, MCTS is applied over a set depth or until the problem at hand is resolved to a terminal state. To explore a large decision space, a methodology to select new actions must be determined. We define a rollout heuristic as the means to generate a new set of actions from a leaf node in a search tree. The rollout heuristic can take a variety of forms; fully random sequences could be chosen, or system knowledge can be applied to inform the relative value of actions. In any case, actions must be generated until the terminal state or maximal search depth is reached. If a new action is not needed, a child node is selected and tree search is recursively simulated from that node.

3. **Backpropagation** - Backpropagation is defined as the means by which immediate rewards simulated by leaf nodes in the search tree impact the estimated value at parent nodes. Given a rollout or simulation routine that returns the discounted cumulative reward R_i sampled for a sequence of actions, one must determine how to revalue the immediate action taken. Generally, the average reward returned for an immediately sampled action $a \in A$

$$V(a) = V(a) + \frac{R_i - V(a)}{N(a)}$$

   is utilized, but other statistical measures may also be incorporated, especially if there are concerns about the variance of rewards.

4. **Selection** - If a new action is not generated, one must determine what previously sampled action to take. Generally, the selection method must balance more detailed exploration of simulated actions with high expected value with further exploration of undersampled actions. As such, a deterministic score function is applied for selection such that the child node maximizing

$$sc_n(i) = V(i) + \sqrt{\frac{f(N)}{N(i)}}$$

   is selected. $N(i)$ represents the number of times child node i was previously selected, and $f(N)$ is an arbitrary non-decreasing map from $\Re^1$ to $\Re^1$. This second term is derived from multi-armed bandit literature, and can be related to a confidence interval for the true value of an action [21]. Generally, the natural logarithm is utilized, but other methods such as polynomial exploration have been applied [14].

5. **Progressive widening** - Generally, large state and action spaces can lead to curses of dimensionality in decision processes. When state and observation spaces are large or continuous, curses of history can also occur. As actions lead to transitions described by generative models, search trees can become infinitely wide after a single transition; that is, an arbitrary action will lead to a different representation of belief for each associated observation that is sampled. As such, in order to limit the breadth of the search tree, one must artificially limit the number of actions explored, as well as the number of observations associated with each sampled action. This so-called arm-increasing rule or progressive widening is analyzed in [22]. Widening is applied for MCTS by [13] with success; this is the first example of double progressive widening, in which the search tree breadth is slowly widened for both generation of new action sequences and state transition or observation generation. Generally, whether progressive widening is allowed is determined by a rule as a function of visits to the parent node i

$$|i| \leq N(i)^{\alpha_d}$$

   such that the number of child nodes are upper bounded by a power law $\alpha_d \in (0,1)$. Note that $||$ is utilized to describe the number of children at an arbitrary node.

With these concepts in mind, the tree search routine from a root node is as follows. First, an action is determined using progressive widening. If the tree is allowed to widen, a new action is sampled; otherwise, a previous action is chosen that maximizes the score function. Next, it is determined whether new transitions or observations should be generated (the specificities depending on whether the problem is fully observable). If a new observation node is generated, the rollout model is applied; otherwise, a previous transition is chosen. If the problem is fully observable, each transition is given equal selection probability; otherwise, previous observation nodes are selected according to measurement likelihoods. The simulation process is then recursively completed from the selected child node. Finally, cumulative rewards from the rollout or recursive simulation are utilized to update expected reward, and total cumulative reward for the search iteration is returned.

Algorithm 1 The recursive simulation routine for the MCTS algorithm, returning an updated history and reward.

procedure SIMULATE(b, h, d)
  if $d = 0$ **then**
    return $\{h,\ 0\}$
  $a \leftarrow$ PROGRESSIVEWIDEN(h)
  $b' \leftarrow$ T(b, a)
  $y \leftarrow$ H(b',a)
  $r \leftarrow$ R(b', b)

  if $|C(ha)| \leq k_y N(ha)^{\alpha_y}$ **then**
    $W(hay) \leftarrow Z(y \,|\, b,\ a)$
  else
    $\{b',\ y,\ r\} \leftarrow$ select $C(ha)$ w.p. $\frac{W(hay)}{\sum_y W(hay)}$

  if $y \notin C(ha)$ **then**
    $C(ha) = C(ha) \bigcup \{b',\ y,\ r,\ W(hay)\}$
    $\{hay, r_{t+}\} \leftarrow$ ROLLOUT(b', hay, d-1)
    $R_i \leftarrow r + \gamma r_{t+}$
  else
    $\{hay, r_{t+}\} \leftarrow$ SIMULATE(b', hay, d-1)
    $R_i \leftarrow r + \gamma r_{t+}$

  $N(h) \leftarrow N(h) + 1$
  $N(ha) \leftarrow N(ha) + 1$
  $V(ha) \leftarrow V(ha) + \frac{R_i - V(ha)}{N(ha)}$

  return $\{h,\ total\}$

Variable	Definition
b	belief in states
h	history (initial search node)
d	depth
a	action
T	state transition
H	measurement function
y	simulated observation
R	reward function
r	sampled reward
$\|C(ha)\|$	number children associated with a given action
W	node likelihood weight

3. MODEL AND METHODOLOGY

The next section outlines a proof by induction determining an upper bound for the error in the expected optimal value as a function of node visits and tree depth. Generally, the methodology utilized by [14] is followed. We extend this proof, developing bounds for POMDPs and outlining convergence in further detail. We begin with several definitions from [14].

Definition: Exponentially Sure in n. Some property P depending on integer N is exponentially sure in N (e.s) if there exist positive constants C, h, η such that the probability P holds is at least

$$p(P) \geq 1 - Cexp(-hN^{\eta}) \tag{2}$$

Definition: Consistency. There exists a coefficient $C_d > 0$ such that for all nodes at integer depth d,

$$|V(z) - V^*(z)| \leq C_d N(z)^{-\gamma_d} \tag{3}$$

exponentially surely in $N(z)$. That is, the difference between the estimated value function and true value exponentially decreases to zero as nodes are visited.

Definitiion: Regularity Hypothesis. For any $\Delta > 0$, we assume there exist $\theta > 0$ and $p > 1$ throughout the simulation such that the probability the value function of a sampled action i differs from the Bellman optimal value is bounded by Δ is

$$p(V(i) \geq V^*(z) - \Delta) \geq \min(1, \theta\Delta^p) \tag{4}$$

Note that to simplify the discussion of the proof, nodes are split into decision nodes, where an action is selected, and observation nodes, where a measurement is applied and a belief update is performed. First, we consider bounds on the error of the estimate of the value function for the transition from a decision node to an observation node.

3.1 Observation nodes are selected according to observation likelihood

Consider a general observation likelihood ω_i, the joint probability of a set of measurements Y_i given a prior state estimate X such that

$$\omega_i = p(Y_i, X) = \prod_{j=1}^{|Y_i|} p(\mathbf{y_j}, X)$$

When widening does not occur, when an action is selected, one must traverse to a previously generated observation node associated with that action. We first apply a general sampling scheme assuming that whenever a traversal is taken from a decision node w to an observation node i, random sampling occurs in proportion to observation likelihood

$$p(i)_0 = \frac{\omega_i}{\sum_{j=1}^{|w|} \omega_j}$$

Because of widening methods, this sampling methodology leads to a bias towards measurements generated early in the simulation process. As such, we introduce a modified form:

$$p(i)_{corr} = p(i)_0 \frac{p(i)_0 N}{N(i)}$$

where N describes the number of visits to the parent decision node and observation node i, respectively. Normalizing this result, we find

$$p(i) = \frac{p(i)_{corr}}{\sum_{j=1}^{|w|} p(j)_{corr}} = \frac{\frac{\omega_i^2 N}{(\sum_{j=1}^{|w|} \omega_j)^2 N(i)}}{\sum_{j=1}^{|w|} \frac{\omega_j^2 N}{(\sum_{k=1}^{|w|} \omega_k)^2 N(j)}} = \frac{\frac{\omega_i^2}{N(i)}}{\sum_{j=1}^{|} w| \frac{\omega_j^2}{N(j)}}$$

Applying this weighting scheme, nodes will converge to their likelihood given infinite simulation, but further refining to the sampling methodology is possible. As observational nodes are sampled according to likelihood, the ratio between the weight and the number of visits to a child observation node converges as

$$\frac{\omega_i}{N(i)} \approx \frac{1}{N}$$

If an observation node is undersampled, this ratio becomes larger than $\frac{1}{N}$. As such, observation node selection can be made purely deterministic, such that the next sampled observation is selected by the criterion

$$o = \underset{i}{\operatorname{argmax}} \frac{\omega_i}{N(i)} \tag{5}$$

The converging behavior of this result is seen in Figure 1, in which the cardinality of observations is slowly widened, and observations are scaled to approximately reflect observation likelihood.

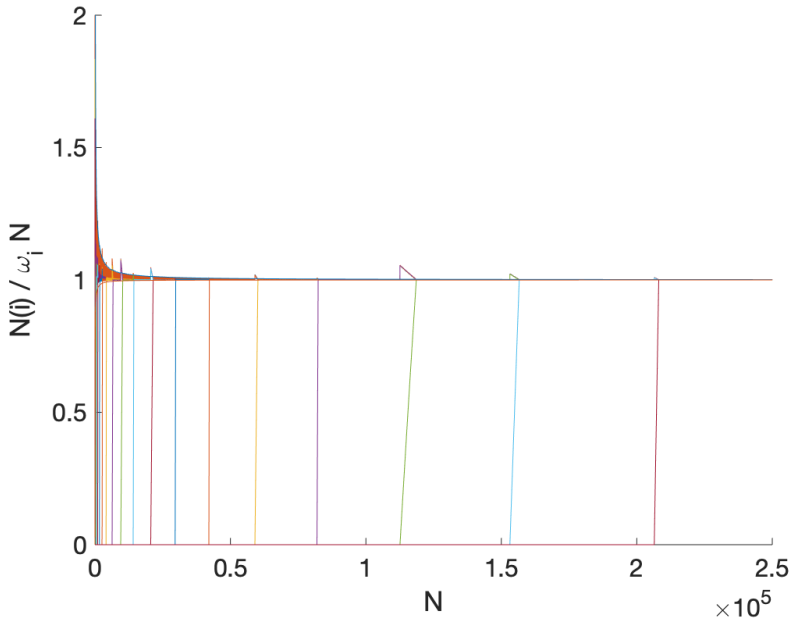


Fig. 1: Observations sampled deterministically according to likelihood.

Using deterministic sampling, the number of visits to each child node can then be upper and lower bounded by

$$N(i) \geq \frac{\omega_i N^2}{N + |w| - 1} \tag{6}$$

$$N(i) \leq \omega_i N + 1 \tag{7}$$

where $|w|$ is the current number of observation nodes. Assume that double progressive widening is applied such that for observation nodes, there exists some widening coefficient α_o where

$$|w| = \lfloor N^{\alpha_o} \rfloor \tag{8}$$

If a new child node has recently been generated, the number of visits to that node are then explicitly

$$N(i) = N - \left\lceil \lfloor N^{\alpha_o} \rfloor^{\frac{1}{\alpha_o}} \right\rceil$$

We assume that $N(i)$ is large relative to the expected number of visits $\omega_i N$, such that the computed bounds hold. These bounds are then utilized to evaluate the consistency of the observation node transition.

3.2 Consistency of Observation Nodes

This section applies Hoeffding's inequality to upper bound the probability the sum of bounded random realizations diverges from the expected sum. For a set of bounded random variables X, Hoeffding's inequality is generally defined as

$$p(|S_n - E[S_n]| \leq t) \geq 1 - 2\exp\left(-\frac{2t^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right) \tag{9}$$

$$S_n = \sum_{i=1}^{n} X_i, \;\; X_i \in [a_i, b_i]$$

We first hope to evaluate the consistency of the estimate of the value function for observation nodes, and in doing so wish to recursively determine the desired widening parameter α_o; studying the error in the estimation at decision node w, we apply the estimate as a function of the associated observation nodes such that

$$|V(w) - V^*(w)| = \left| \sum_{i=1}^{|w|} \frac{N(i)}{N} V(i) - V^*(w) \right| \tag{10}$$

Applying the triangle inequality, Equation 10 can be related to estimate error at each observation node

$$|V(w) - V^*(w)| \leq \left| \sum_{i=1}^{|w|} \frac{N(i)}{N} (V(i) - V^*(i)) \right| + \left| \sum_{i=1}^{|w|} \frac{N(i)}{N} (V^*(i) - V^*(w)) \right| \tag{11}$$

Considering the first term of this result, note that the number of visits $N(i)$ to node i is lower bounded by Equation 6, and upper bounded by Equation 7. The consistency definition of Equation 3 also applies. As such an upper bound to this term can be expressed as

$$\left| \sum_{i=1}^{|w|} \frac{N(i)}{N} (V(i) - V^*(i)) \right| \leq \left| (\sum_{i=1}^{|w|} \omega_i + \frac{1}{N})(C_d N(i)^{-\gamma_d}) \right| \leq \left| \sum_{i=1}^{|w|} (\omega_i + \frac{1}{N}) C_d (\frac{\omega_i N^2}{N + |w| - 1})^{-\gamma_d}) \right|$$

When the number of child nodes at w is small relative to N, this bound is approximately

$$\left| \sum_{i=1}^{|w|} \frac{N(i)}{N} (V(i) - V^*(i)) \right| \leq \left| \sum_{i=1}^{|w|} (\omega_i + \frac{1}{N}) C_d (\omega_i N)^{-\gamma_d}) \right| \tag{12}$$

To express this result as a function of the desired widening parameter α_o, we consider the expected error is solely a function of the expected weight

$$E[\omega_i] = \frac{E[p(Y_i, X)]}{\sum_{j=1}^{|w|} E[p(Y_j, X)]} = \frac{1}{|w|} = \frac{1}{\lfloor N^{\alpha} \rfloor}$$

To further bound the error expressions, it is also critical to consider the variance of expected errror. We demonstrate this evaluation assuming that the joint probabilities for state estimates and measurements are Gaussian random variables, but the structure of these arguments holds for an arbitrary distribution. With these assumptions, the second moment of observation node likelihoods is expressed as

$$E[\omega_i^2] = \frac{E[p(Y_i,X)^2]}{|w|E[p(Y_j,X)^2] + (|w|^2 - |w|)E[p(Y_j,X)]^2}$$
$$= \frac{1}{|w|} \frac{(2\pi)^{-k}|P|^{-1}}{(2\pi)^{-k}|P|^{-1} + 3^{\frac{k}{2}}(|w|-1)(2^{-2k}\pi^{-k}|P|^{-1})}$$

$$E[\omega_i^2] = \frac{1}{|w|} \frac{1}{1 + 3^{\frac{k}{2}}(|w|-1)2^{-k}} \tag{13}$$

To reach this result, we must apply the expectation operator to

$$E[p(Y_i,X)] = \int_{-\infty}^{\infty} \left(\frac{1}{(2\pi)^{\frac{k}{2}}|\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(Y_i - H_iX)^T \Sigma_j^{-1}(Y_i - H_iX) \right) \right)^2 dX$$
$$E[p(Y_i,X)^2] = \int_{-\infty}^{\infty} \left(\frac{1}{(2\pi)^{\frac{k}{2}}|\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(Y_i - H_iX)^T \Sigma_j^{-1}(Y_i - H_iX) \right) \right)^3 dX$$

These expressions are unormalized Gaussians with an adjusted variance such that

$$E[p(Y_i,X)] = \frac{1}{(2\pi)^k|\Sigma|} \int_{-\infty}^{\infty} \exp\left(-(Y_i - H_iX)^T \Sigma_j^{-1}(Y_i - H_iX) \right) dX = \frac{1}{(2\pi)^k|\Sigma|} \pi^{\frac{k}{2}}|\Sigma|^{\frac{1}{2}}$$
$$= 2^{-k}\pi^{-\frac{k}{2}}|\Sigma|^{-\frac{1}{2}}$$

and similarly

$$E[p(Y_i,X)^2] = \frac{1}{(2\pi)^{\frac{3k}{2}}|\Sigma|^{\frac{3}{2}}} \int_{-\infty}^{\infty} \exp\left(-\frac{3}{2}(Y_i - H_iX)^T \Sigma_j^{-1}(Y_i - H_iX) \right) dX$$
$$= (2\pi)^{-k}3^{-\frac{k}{2}}|\Sigma|^{-1}$$

Equation 13 is expressed from these results, and likelihood sample variance can then be computed as

$$var(\omega_i) = E[\omega_i^2] - E[\omega_i]^2 = \frac{1}{|w|} \frac{1}{1 + 3^{\frac{k}{2}}(|w|-1)2^{-k}} - \frac{1}{|w|^2} \tag{14}$$

Note that variance follows as $O(|w|^{-2})$ as the number of child nodes grows large. Variance in observation weights is not a function of properties of the decision made, but simply the number of samples generated. Asthe sample standard deviation decreases with $\frac{1}{|w|}$ in the same order as the expectation, the following arguments on boundedness hold at the same rates to arbitrary variance.

Returning to Equation 12, we can express the result as

$$E\left[\left| \sum_{i=1}^{|w|} (\omega_i + \frac{1}{N}) C_d (\omega_i N)^{-\gamma_d} \right| \right] = N^{\alpha_o} \left| (E[\omega_i] + \frac{1}{N}) C_d (E[\omega_i]N)^{-\gamma_d} \right|$$

$$\approx C_d(1 + N^{\alpha_o - 1})N^{-(1-\alpha_o)\gamma_d} = O(N^{-(1-\alpha_o)\gamma_d} + N^{-(1-\alpha_o)(1+\gamma_d)}) \tag{15}$$

Since $\alpha_o \in (0,1),\ \gamma_d \in (0,1)$, error in the first term then decreases with the dominating term $O(N^{-(1-\alpha_o)\gamma_d})$.

Considering the second term in Equation 11, using Equation 7 note the result is upper bounded by

$$\left| \sum_{i=1}^{|w|} \frac{N(i)}{N} \left(V^*(i) - V^*(w) \right) \right| \leq \left| \sum_{i=1}^{w} |(\omega_i + \frac{1}{N}) \left(V^*(i) - V^*(w) \right) \right|$$

Again taking the expectation, we can apply Hoeffding's inequality, finding

$$E\left[\left|\sum_{i=1}^{|w|}(\omega_i+\frac{1}{N})(V^*(i)-V^*(w))\right|\right]=\left|\sum_{i=1}^{|w|}(N^{-\alpha_o}+N^{-1})\left(V^*(i)-V^*(w)\right)\right|\leq t \tag{16}$$

with probability at least

$$1-2\exp\left(-\frac{2t^2}{N^{\alpha_o}(N^{-\alpha_o})^2}\right)$$

This result must also be constrained as strictly less than or equal to the dominating term such that $t\leq N^{-(1-\alpha)\gamma_d}$. Choosing to make these terms decrease at an equivalent rate, we also would like this probability to grows as a function in t $1-2\exp(-Ct^{-1})$, giving the result

$$t^{-3}=N^{\alpha_o}$$
$$N^{3(1-\alpha_o)\gamma_d}=N^{\alpha_o}$$

$$\alpha_o=\frac{3\gamma_d}{1+3\gamma_d} \tag{17}$$

This result is then substituted back into the dominating term $O(N^{-(1-\alpha_o)\gamma_d})$. With $\alpha_o=\frac{3\gamma_d}{1+3\gamma_d}$ we find a minimal exponent

$$\gamma_{d-\frac{1}{2}}=\frac{\gamma_d}{1+3\gamma_d} \tag{18}$$

and the nodes are recursively consistent. Note that this result for convergence is equivalent to that for random MDP transitions in [14].

3.3 Alternate proof for Observation Nodes with Guarantees

In many cases, this result may be sufficient, but it does rely on assumptions on the variance of the random variables considered. As the likelihood weights that challenge this derivation can be considered as a set of realizations, the application of Samuelson's inequality can also be considered, in which upper and lower bounds for a set of samples can be expressed as

$$\bar{X}-\sigma\sqrt{N-1}\leq X_j\leq\bar{X}+\sigma\sqrt{N-1}\;\forall X_j\in X \tag{19}$$

where $\bar{X}$ is the sample mean, σ is the sample variance, and N samples are taken.

Applied to the likelihood weights, we find

$$\omega_i\leq\frac{1}{|w|}+\sqrt{|w|-1}var(\omega_i)\leq\frac{1}{|w|}+\frac{|w|-1}{|w|}(\frac{1-a}{1-a+a|w|})^{\frac{1}{2}}\approx\frac{1}{|w|}+(\frac{1-a}{a|w|})^{\frac{1}{2}}$$

as $a|w|>>1-a$, where $a=3^{\frac{k}{2}}2^{-k}$ and k is the state space dimension. Similarly applying the lower bound

$$\frac{1}{|w|}-(\frac{1-a}{a|w|})^{\frac{1}{2}}\leq\omega_i\leq\frac{1}{|w|}+(\frac{1-a}{a|w|})^{\frac{1}{2}}$$

Applying these new guaranteed bounds, we return to Equation 12 with

$$\left|\sum_{i=1}^{|w|}\frac{N(i)}{N}(V(i)-V^*(i))\right|\leq\left|\sum_{i=1}^{|w|}\left(\frac{1}{|w|}+(\frac{1-a}{a|w|})^{\frac{1}{2}}+\frac{1}{N}\right)C_d\left((\frac{1}{|w|}-(\frac{1-a}{a|w|})^{\frac{1}{2}})N\right)^{-\gamma_d}\right|$$

Noting $|w|=\lfloor N^{\alpha_o}\rfloor$, the result then has runtimes on the order

$$O(N^{-\gamma_d(1-\alpha_o)}+N^{-\gamma_d(1-\frac{\alpha_o}{2})}+N^{\frac{\alpha_o}{2}-\gamma_d(1-\alpha_o)}+N^{\frac{\alpha_o}{2}-\gamma_d(1-\frac{\alpha_o}{2})}+N^{-(1+\gamma_d)(1-\alpha_o)}+N^{-1-\alpha_o-\gamma_d(1-\frac{\alpha_o}{2})}) \tag{20}$$

With $\alpha_o, \gamma_d \in [0,1)$, this term is then dominated with runtime $O(N^{\frac{\alpha_o}{2}-\gamma_d(1-\alpha_o)})$. Restricting this exponent to be negative, we find

$$\alpha_o < \frac{\gamma_d}{\frac{1}{2}+\gamma_d} \tag{21}$$

Now taking the second term of Equation 11, we then reapply Hoeffding's inequality.

$$\begin{aligned}\left|\sum_{i=1}^{|w|}\frac{N(i)}{N}(V^*(i)-V^*(w))\right| &\leq \left|\sum_{i=1}^{|w|}(\omega_i+\frac{1}{N})(V^*(i)-V^*(w))\right| \\ &\leq N^{\alpha_o}\left|(\frac{1}{|w|}+(\frac{1-a}{a|w|})^{\frac{1}{2}}+\frac{1}{N})(V^*(i)-V^*(w))\right| \leq t\end{aligned}$$

with probability at least (dropping the small N^{-1} term)

$$1-2\exp\left(-2t^2N^{-\alpha_o}(N^{-\alpha_o}+bN^{-\frac{\alpha_o}{2}})^{-2}\right) \approx 1-2\exp\left(-2\frac{t^2}{b^2+2bN^{-\frac{\alpha_o}{2}}}\right)$$

for $b=(\frac{1-a}{a})^{\frac{1}{2}}$. Forcing this probability to converge as $O(Ct^{-1})$ with $t^{-1} \leq N^{\frac{\alpha_o}{2}-\gamma_d(1-\alpha_o)}$ as the denominator of exponent is dominated by the term in N, we find the relation

$$\begin{aligned}t^3 &= N^{\frac{\alpha_o}{2}} \\ N^{-\frac{3\alpha_o}{2}+3\gamma_d(1-\alpha_o)} &= N^{\frac{\alpha_o}{2}} \\ \alpha_o(2+3\gamma_d) &= 3\gamma_d\end{aligned}$$

$$\alpha_o = \frac{3\gamma_d}{2+3\gamma_d} \tag{22}$$

Reapplying this to the maximum exponent, we find convergence rates of

$$\gamma_{d-\frac{1}{2}} = -\left(\frac{\alpha}{2}-\gamma_d(1-\alpha)\right)$$

$$\gamma_{d-\frac{1}{2}} = \frac{\gamma_d}{4+6\gamma_d} \tag{23}$$

on guaranteed bounds for error in estimating the value function across observational nodes.

3.4 Consistency of Decision Nodes

Supposing there exists some constant for consistency across observation nodes $\gamma_{d-\frac{1}{2}}$, we now wish to determine progressive widening coefficients for decision nodes α_d and a recursion for convergence factor γ_{d-1}. The regularity hypothesis Equation 4 must be applied as an assumption.

We start by establishing an exploration function f that ensures decision nodes are selected infinitely often given infinite simulation. Lemma 3 of [14] holds. For an arbitrary non-decreasing map f from $\Re^1$ to $\Re^1$, a score can be computed at observation node z for child decision node i as

$$sc_n(i) = V_n(i) + \sqrt{\frac{f(N)}{N(i)}} \tag{24}$$

All children must be selected infinitely often provided that $\lim_{+\infty} f = +\infty$. In particular, bounding behavior on visits can be defined

$$N(i) \geq \frac{1}{4}\min\left(f(N^{1-\alpha_d}), N^{1-\alpha_d}\right) \tag{25}$$

First, the use of polynomial exploration as in [14] is justified, as compared to methodologies used in [11,13]. Consider a polynomial exploration function

$$f(N) = N^e \tag{26}$$

where $e \in (0,1)$.

An upper bound on estimation error is determined as [14]

$$V(z) - V(z) \leq (1 + C_{d-\frac{1}{2}}) N^{-\gamma_{d-\frac{1}{2}} \frac{1-\alpha_d}{1+\gamma_{d-\frac{1}{2}}}} \tag{27}$$

for observation node z. One must then determine a fixed coefficient γ_{d-1} such that all child decision nodes w of z verify exponentially surely

$$|V(z) - V^*(z)| \leq C_{d-1} N^{-\gamma_{d-1}}$$

In order to find a lower bound, the assumptions of Equation 4 are followed. An expression for the lower bound is desired as a function of the minimal number of visits described in Equation 25. As such, we choose the bound Δ to scale in proportion to the minimal number of visits

$$\Delta = \left(\frac{1}{4} min(f(N), N) \right)^{-\gamma_{d-\frac{1}{2}}} \tag{28}$$

Now consider a time $N^{\xi(1-\alpha_d)}$, with some positive coefficient ξ. At this step, knowing the widening coefficient, the number of children of observation node z is at least $\left\lfloor N^{\xi(1-\alpha_d)\alpha_d} \right\rfloor$. Assuming the exploration function is strictly less than N, the probability not a single child node lies within the bound Δ is

$$\begin{aligned} p_n &= (1 - \theta \Delta^p)^{\left\lfloor N^{\xi(1-\alpha_d)\alpha_d} \right\rfloor} \\ \log p_n &\approx N^{\xi(1-\alpha_d)\alpha_d} \log(1 - \theta \Delta^p) \end{aligned}$$

When the exponentially sure component is small ($\theta \Delta^p << 1$) a Taylor series expansion can be applied and

$$\log p_n \approx -4^{\gamma_{d-\frac{1}{2}} p} N^{\xi(1-\alpha_d)\alpha_d} \theta f(N^{\xi(1-\alpha_d)})^{-\gamma_{d-\frac{1}{2}} p} \tag{29}$$

For the proof to proceed, this result must monotonically decrease in N such that

$$p_n(N \to \infty) \to 0$$

It also is not desired for this result to be a function of the undetermined regularity constant p. For this to be the case, f must be a polynomial function; applying Equation 26,

$$\log p_n \approx -4^{\gamma_{d-\frac{1}{2}} p} N^{\xi(1-\alpha_d)(\alpha_d - e\gamma_{d-\frac{1}{2}} p)}$$

The quantity $\alpha_d - e\gamma_{d-\frac{1}{2}} p$ is then restricted such that it is only a function of α_d

$$\begin{aligned} \alpha_d - e\gamma_{d-\frac{1}{2}} p &> 0 \\ e &= \frac{c\alpha_d}{\gamma_{d-\frac{1}{2}} p} \\ \alpha_d - e\gamma_{d-\frac{1}{2}} p &= (1-c)\alpha_d \end{aligned}$$

To determine the arbitrary constant c, the log probability not a single node's value function is estimated to be within Δ of the Bellman optimal value is constrained to $O(-C\Delta^{-1})$. Returning to Equation 29,

$$\begin{aligned} \Delta^{-1} &= N^{(1-c)\xi\alpha_d(1-\alpha_d)} \\ N^{e\gamma_{d-\frac{1}{2}}p\xi(1-\alpha_d)} &= N^{c\xi\alpha_d(1-\alpha_d)} = N^{(1-c)\xi\alpha_d(1-\alpha_d)} \\ c &= 0.5 \end{aligned}$$

Then, $e = \frac{\alpha_d}{2\gamma_{d-\frac{1}{2}}p}$.

So long as ξ is lower bounded by a constant in the domain $(0,1)$, the estimation error is then lower-bounded by the quantities Δ, $N^{\xi-1}$, $\frac{N^{\alpha_d+e-1}}{\Delta^2}$ [14].

ξ is a chosen quantity; therefore, the second quantity can be bounded as $O(\Delta)$ as follows

$$\begin{aligned} N^{\xi-1} = C\Delta = C4^{\gamma_{d-\frac{1}{2}}}N^{-\gamma_{d-\frac{1}{2}}\xi e(1-\alpha)} \\ \xi - 1 = -\gamma_{d-\frac{1}{2}}\xi e(1-\alpha_d) \\ \xi = \frac{1}{1+\gamma_{d-\frac{1}{2}}e(1-\alpha_d)} = \frac{p}{p+0.5\alpha_d(1-\alpha_d)} < 1 \\ N^{\xi-1} = 4^{-\gamma_{d-\frac{1}{2}}}\Delta < \Delta = O(\Delta) \end{aligned}$$

We now wish to evaluate the third term $\frac{N^{\alpha_d+e-1}}{\Delta^2}$, desiring $N^{\alpha_d+e-1} = O(\Delta^3)$. Considering this, rather arbitrarily add the constraint

$$\begin{aligned} \alpha_d + e - 1 = \left(1 + \frac{1}{2p\gamma_{d-1}}\right)\alpha_d - 1 \leq -\frac{1}{2} \\ \alpha_d \leq \frac{2p\gamma_{d-1}}{1+2p\gamma_{d-1}} \end{aligned}$$

To reduce constants let $p = 2$ and

$$\alpha_d = \frac{\gamma_{d-1}}{1+4\gamma_{d-1}} \tag{30}$$

is found to satisfy the constraint. Then,

$$\log(\Delta^3) = 6\xi e(1-\alpha_d)\gamma_{d-1} = \frac{6\alpha_d(1-\alpha_d)}{4+\alpha_d(1-\alpha_d)} \leq \frac{3}{13}$$

and

$$O(\Delta^3) \geq O(N^{-\frac{3}{13}}) > O(N^{-\frac{1}{2}}) \geq O(N^{\alpha_d+e-1}) \tag{31}$$

and the third term must be upper bounded by $O(\Delta)$. Therefore, the term $O(\Delta)$ lower bounds the estimation error. Substituting into a recursive form, we find

$$V(z) - V(z) \geq CN^{-\frac{\gamma_{d-\frac{1}{2}}(1+3\gamma_{d-\frac{1}{2}})}{1+4(1+4\gamma_{d-\frac{1}{2}})^2}}$$

and

$$\gamma_{d-1} = \frac{\gamma_{d-\frac{1}{2}}(1+3\gamma_{d-\frac{1}{2}})}{1+4(1+4\gamma_{d-\frac{1}{2}})^2} \tag{32}$$

3.5 Proof Conclusion

To analyze convergence as a function of a maximal depth d_{max}, we initialize the widening coefficient at an observation node as $\alpha_o(d_{max}) = 1$. Then, using Equation 18 $\gamma_{d_{max}} = \frac{1}{3}$. It follows using Equation 32 that $\alpha_d(d_{max-\frac{1}{2}}) = \frac{1}{7}$ and $\gamma_{d_{max-\frac{1}{2}}} = \frac{6}{205}$. Recursively, parent nodes may be analyzed until the root node is reached, and polynomial bounds on value function estimation error convergence are obtained.

4. SIMULATION RESULTS

In order to demonstrate the validity of application of Monte Carlo Tree Search for the sensor tasking problem, we now consider a representative example scenario. In this case, tracks are assumed to exist for a large set of 1000 objects and maintenance of these tracks is desired. These objects are generated from real two-line element data, and restricted to a subset of the full NORAD catalog with a semi-major axis greater than 8300 km. A visualization of the resultant population can be seen in Figure 2. Initial states are taken from the epoch 3/11/2020, UTC 00:00:00. Note that this subset of objects contains a set of near-geostationary satellites that are occluded during a portion of this simulation, adding to the complexity of the tasking scenario. In addition, the chronologically first 1000 objects admitted by the desired constraints are chosen; as such, many of the objects considered follow slightly inclined graveyard orbits about the geostationary belt. SOs utilized within the simulation are required to have a semi-major axis greater than 8378.14 kilometers. Uncertainties are randomly initialized as diagonal covariance matrices, with positional uncertainties $P_r \in (0.1, 10.0)km^2$ and velocity uncertainties $P_v \in (1e-9, 5e-8)(km/s)^2$. The same initial uncertainties are applied in each simulation. Tasking decisions are made over a 24 hour period using a set of three observers.

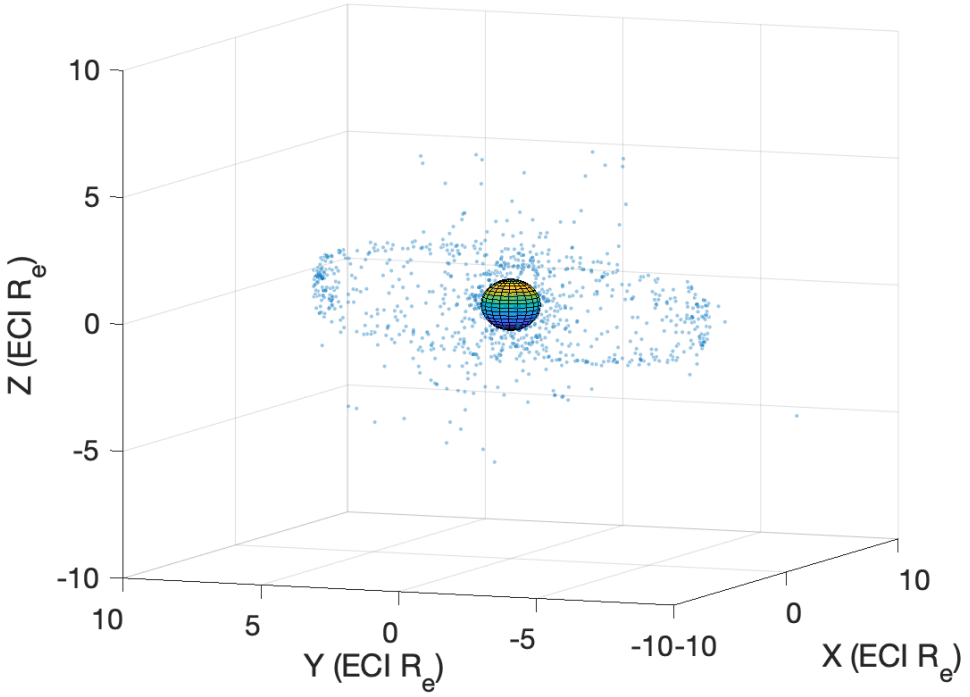


Fig. 2: Initial object position estimates utilized

4.1 Observers

In this problem, we consider the use of two ground-based sensors and a single space-based observer placed on a sun-synchronous orbit at an altitude of 600 km. The two ground-based sensors are placed in Boulder, CO and on the island Diego Garcia. Further details on sensor placement are outlined in Tables 1 and 2.

	Boulder	**Diego Garcia**
Latitude (deg)	40.0150 N	7.3195 S
Longitude (deg)	105.2705 W	72.4229 E
Altitude (km)	0	0

Table 1: Ground-based sensor locations

	Sun-synchronous
a (km)	6978.14
e	0
i (rad)	1.70674
Ω (rad)	$\arctan \frac{y_\odot}{x_\odot}$
ω (rad)	0
M (rad)	0

Table 2: Space-based sensor elements

Note that $\odot$ refers to the Earth-Sun vector, and that the Sun-synchronous observer right ascension of the ascending node is defined relative to that vector at the epoch time. It is assumed that each sensor is perfectly agile; that is, each pointing decision is independent of any previous decision, and slew constraints are disregarded. Unobscured measurements are assumed to have uncertainty 16 asec^2 in both right ascension and declination. In addition, a variety of viewing constraints are placed on each observer, largely as a function of solar and lunar viewing directions.

4.1.1 Earth-Fixed Observation Constraints

Four major constraints are placed on the viability of a ground-based observation. First, observations are only allowed at night, defined by the angle between the Earth-Sun vector and the vector to the observer in the ECI frame exceeding 90 degrees,

$$\vec{r}_o \cdot \vec{r}_\odot < 0$$

Next, the expected azimuth θ and elevation ψ directions from the observer to a potential object are computed. The object is only considered visible if it is not below a critical elevation of 20 degrees, near the moon line of sight, or occluded by the Earth's shadow. A critical angular displacement of 2 degrees is required to avoid the lunar obstruction. Occulsion is modelled as a right circular cone with apex angle

$$\Theta = 2\Phi = 2\arctan((R_\odot - R_\oplus)/|\vec{r}_\odot|)$$

The vector to the object $\vec{r}_s$ is then split into a component parallel to $\vec{r}_\odot$, $r_{s\parallel}$, and a perpendicular component $r_{s\perp}$. The object is considered occluded if it lies within the projected cone such that

$$r_{s\perp} < R_\oplus - r_{s\parallel}\tan\Phi$$

and

$$\vec{r}_o \cdot \vec{r}_\odot < 0$$

4.1.2 Space-Based Observation Constraints

Similar constraints are applied for space-based observers within the simulation paradigm. Rather than a daytime constraint, a space-based observer is simply constrained from pointing within a critical angle toward the sun of 2 degrees. Similar constraints are applied for lunar exclusion. The previously described occlusion model also is applied in this case. In addition, one also must consider whether the line of sight from a space-based observer to an object is obstructed by the Earth in a similar manner to the elevation constraint applied for a ground-based observer. To determine this constraint, using the observer vector $\vec{r}_o$ and observer to object vector $\vec{r}_{os} = \vec{r}_s - \vec{r}_o$, we require

$$\theta = \arccos\frac{\vec{r}_o \cdot \vec{r}_{os}}{r_o r_{os}}$$
$$r^* = r_o \sin\theta > R_\oplus + a^*$$

where a^* is a critical altitude for visibility, chosen as 100 km.

Observations for each sensor are taken at a 15 second cadence. It is assumed that measurements are instantaneously shared with a central simulating node for decision making. Right ascension α and declination δ measurements are utilized for each sensor. Expected measurements are computed as

$$\alpha = \arctan \frac{y_{os}}{x_{os}}$$
$$\delta = \arcsin \frac{z_{os}}{r_{os}}$$

Note that $(x,y,z)_{os}$ are the positional components of the vector from the observer to the object state estimate. The measurement Jacobian is computed as

$$H = \begin{bmatrix} \frac{\partial \alpha}{\partial \vec{r}_s} \\ \frac{\partial \delta}{\partial \vec{r}_s} \end{bmatrix} = \begin{bmatrix} -\frac{y_{os}}{r_{xy}^2} & \frac{x_{os}}{r_{xy}^2} & 0 & 0 & 0 & 0 \\ -\frac{x_{os} z_{os}}{r_{os}^2 r_{xy}} & -\frac{y_{os} z_{os}}{r_{os}^2 r_{xy}} & \frac{r_{xy}}{r_{os}^2} & 0 & 0 & 0 \end{bmatrix}$$

where

$$r_{xy} = \sqrt{r_{os}^2 - z_{os}^2} = \sqrt{x_{os}^2 + y_{os}^2}$$

4.2 Dynamics

A simplified dynamics model is applied in this scenario, solely utilizing two body orbital motion. This model can easily be extended to incorporate usual perturbing forces, but the applied model is sufficient to elucidate the general structure of growth of uncertainties over time.

4.3 Tree Search

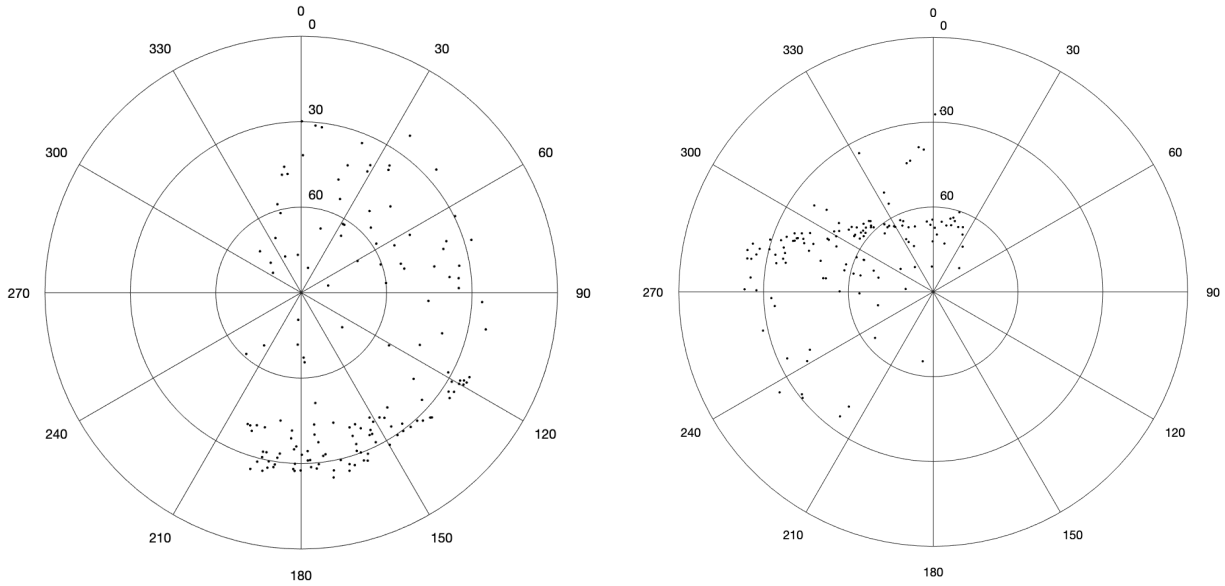
The tree search methodology uses double progressive widening with actions and observations, with a discount factor $\gamma = 0.9$. The simulation routine is applied over a maximum depth $d = 15$, and the routine is parallelized over a set of threads. Critical to the decision exploration process are the rollout heuristic used to generate new actions to take and the reward function utilized to evaluate the value of an action taken.

4.3.1 Rollout Policies

In this simulation, a softmax rollout policy is utilized to determine actions taken. We assume that a single action maps directly to observation of a single object. Initial values of actions i, z_i are taken by weighting and summing the positional and velocity covariance traces or setting the value to 0 if the object is not expected to be visible. These values are then normalized and transformed with the softmax function

$$\sigma_i = \frac{e^{z_i}}{\sum_{i=0}^{|a|} e^{z_i}}$$

where $|a|$ is the branching factor or set of possible decisions for an observer. Note that objects with a value of 0 are also given a transformed value of 0, rather than $e^0 = 1$. Actions are then sampled over the weighted sum of softmax values. Observers are considered independent for the purposes of action generation; that is, a softmax-sampled action for one observer will not impact the sampled action for another observer. Generally, for this scenario, we note a branching factor of approximately 150 for each observer, resulting an a set of around 3 million viable actions. Figures 3a and 3b demonstrate two potential decision scenarios for the Boulder and Diego Garcia observing sites, respectively.



(a) Visible objects from Boulder, 3/11/2020, 2:0:0 UTC

(b) Visible objects from Diego Garcia, 3/11/2020, 0:0:0 UTC

Fig. 3: Sky plots of potential observations at two ground-based sites.

An information-based sampling policy is also considered, following the ad-hoc strategy of [3]. Weighting scores are computed by comparing the trace of uncertainty projected into measurement space and measurement uncertainty such that

$$sc_{ijk} = \alpha\, tr(H_{ijk} P_i H_{ijk}^T) + (1-\alpha) tr(R_{ijk})$$

for object i being tracked by observer j at timestep k. H represents the measurement jacobian, and R represents the expected measurement noise associated with the potential observation. α is a hyperparameter such that $\alpha \in (0,1)$.

Generated decisions are randomly sampled over the sum of scores, and as in the case of the softmax policy, actions are generated for each observer independently.

4.3.2 Reward Functions

The weighted change in covariance trace is utilized as a means to evaluate the value of actions taken. This weighted change incorporates the loss of information in the propagation step to the next observation time. Because of this, this reward can be negative, and should not be thought of as a metric. It is, however, useful in the context of this work, especially as it is easily extended to apply to multiple objects and multiple observers. The trace can be related to information-theoretic quantities such as the change in differential entropy; for a multivariate normal distribution, differential entropy can be computed as

$$h(P) = \frac{1}{2} \ln((2\pi e)^N |P|)$$

and

$$tr(P) \geq N|P|^{\frac{1}{N}}$$

As such, the trace grows in much the same manner as differential entropy.

4.4 Results

In the tree search methodology, at timestep $k+1$, the simulation routine is allowed to proceed for the duration of exposure time for the previous measurement step (15 seconds). The search tree is initialized with all previously

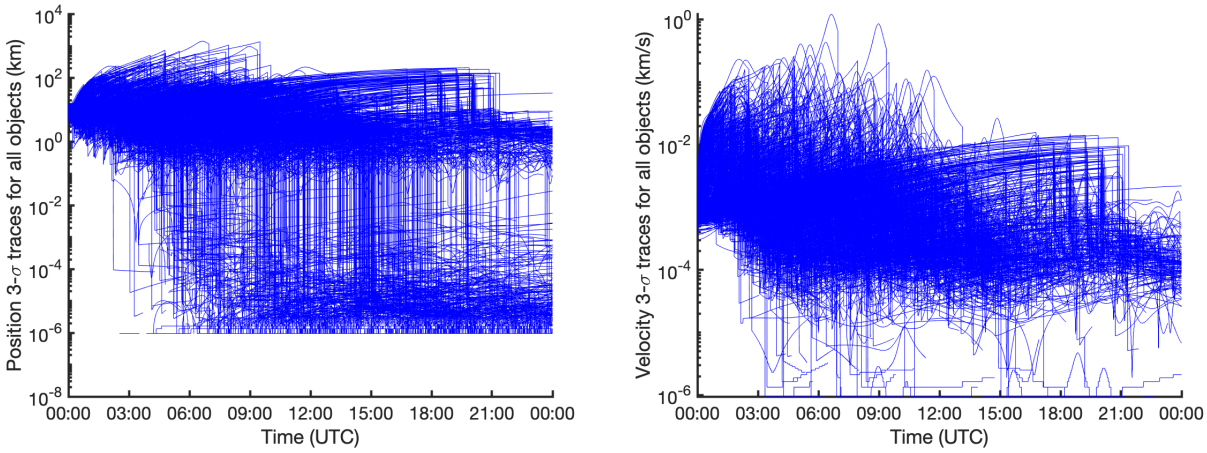
received measurements $y_{1:k-1}$ and a generated measurement y_k^* for the current exposure. When the current exposure finishes, the simulation process terminates, and the next action is chosen as the best child node generated during the tree search. A new root node is generated using the newly received measurement y_k and a generated measurement for the next exposure y_{k+1}^*. The process then repeats ad infinitum.

Results for a single simulation are presented using the ad-hoc MCTS rollout policy. As a means for comparison, we adopt the greedy version of this methodology, a slight modification from the ad-hoc method of [3]. Scores for tasking actions are computed in measurement space and at each timestep, the action that maximizes the score function is selected.

Applying the ad-hoc MCTS policy, Figure 4 demonstrates an overlay of position and velocity traces for all satellites, computed as

$$tr_r(P) = 3\sqrt{P_{xx} + P_{yy} + P_{zz}}$$
$$tr_v(P) = 3\sqrt{P_{\dot{x}\dot{x}} + P_{\dot{y}\dot{y}} + P_{\dot{z}\dot{z}}}$$

The ad-hoc methodology is able to maintain uncertainties for all tracks by the end of simulation utilizing the set of observers given. Uncertainties for several objects initially grow relatively large, on the order of 1000 km 3-σ in positional uncertainty and 1 km/s 3-σ in velocity uncertainty. It is likely that in reality these uncertainties would lead to a loss of track for a ground-based sensor. If a sensor has a 1 degree diameter diagonal field of view, the spherically projected field of regard has a diameter of approximately 650 km. Assuming observations are taken pointing at the maximum likelihood location of the tasked SO, for some of these objects there is a significant probablity the true state may lie outside of the sensor field of view. However, we assume for the purposes of this simulation that a measurement may be resolved. In the future, probability of detection can be incorporated, and prioritizing maintenance for high-uncertainty SOs may be beneficial. In any case, by the end of simulation, all uncertainties are reduced below 100 km 3-σ in position and 10 m/s 3-σ in velocity.



(a) Evolution of positional uncertainties over time

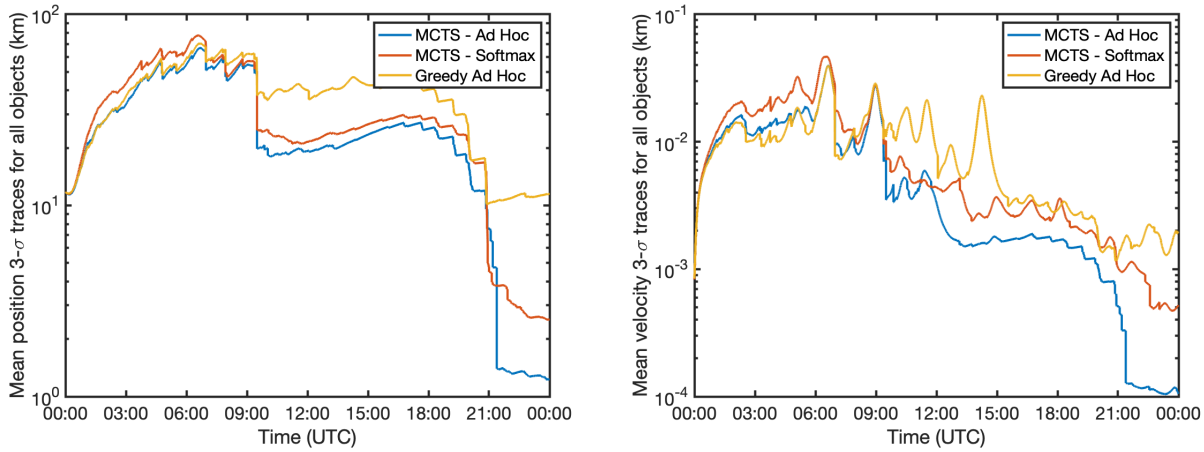
(b) Evolution of velocity uncertainties over time

Fig. 4: Object positional and velocity uncertainties.

In Figure 5, average uncertainties are displayed for each methodology, and in Figure 6, the standard deviation in positional and velocity uncertainty traces is displayed. This provides further insight into the diversity of tasking decisions made over time, and one can note very consistent behavior relative to the mean uncertainties. Standard deviations in velocity uncertainties are quite small, while positional uncertainty standard deviations are observed to be relatively large. This is a function of the observational methods used in the study, in which the sensors utilized have very small angular uncertainties. Consider an example geostationary object at an altitude of 35786 km. When directly overhead, an unobstructed measurement with variance 16 asec^2 has an equivalent spread in position space of

$$\sigma_r^2 = \sigma_\theta^2 (\frac{\pi}{3600(180)})^2 (35786)^2 = 0.48km$$

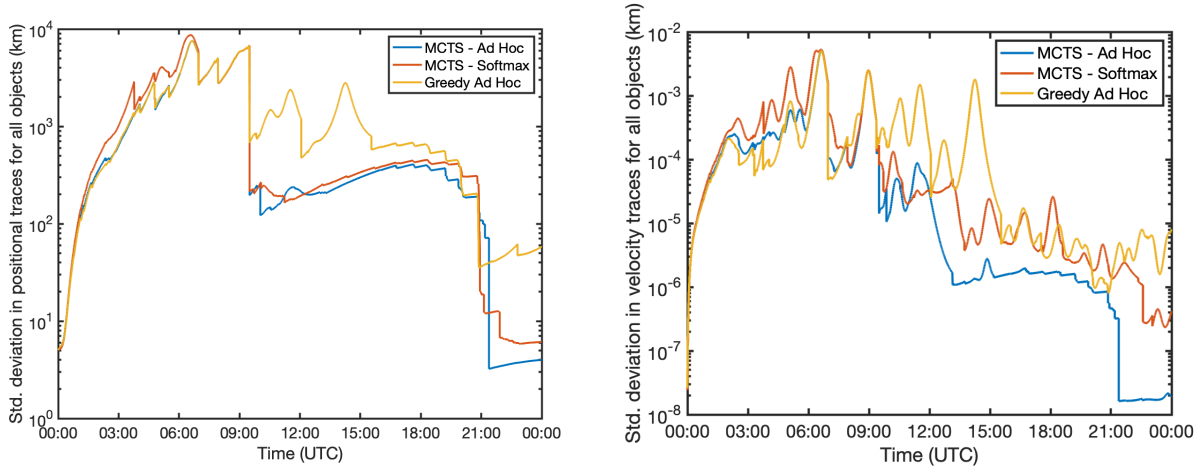
This is quite small relative to the potential growth of the positional covariance ellipsoids. As such, one can expect uncertainties to quickly decrease when objects are measured, and it can be noted that the positional standard deviations are dominated by unmeasured objects, and that the standard deviation values can be approximately related to the mean, scaled by the square root of the number of objects tracked.



(a) Evolution of mean positional uncertainty over time

(b) Evolution of mean velocity uncertainty over time

Fig. 5: Average positional and velocity uncertainties for each tasking methodology.



(a) Positional uncertainty standard deviation

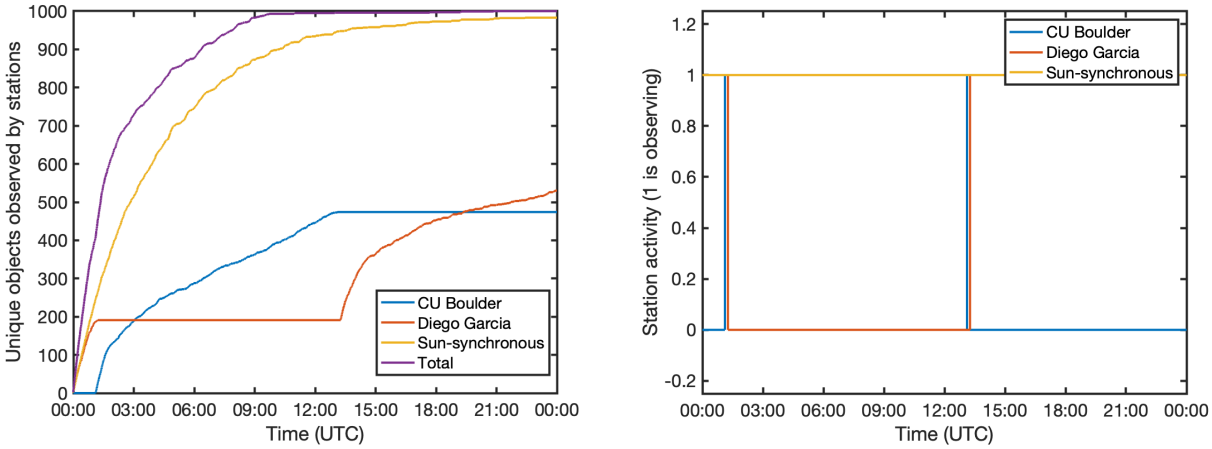
(b) Velocity uncertainty standard deviation

Fig. 6: Spread of object state uncertainties.

In the figures, an initial growth in both uncertainty ellipsoids on average and variances of uncertainty is observed over approximately 6 hours of simulation; at that point, uncertainties seem to reach a peak, after which both positional and velocity uncertainties begin to grow smaller on average over time. From this interval, each tasking methodology is able to maintain the object catalog. One can also note a large decrease in positional uncertainty near 21:00 UTC. At this point, the final object in the study is observed for the first time. Also of interest is the fact that steady-state average traces never seem to be reached. In the future, longer simulations could aid in ascertaining the lower-bound capabilities of the set of observers used for catalog maintenance, but in any case, it appears that further knowledge of the object set can be gained.

It is noted that both MCTS methodologies achieve a performance improvement over the greedy approach. While the ad-hoc MCTS methodology demonstrates an improvement over the softmax policy, that policy is still interesting in

that is completely measurement-free, and does not require computation of measurement jacobians. One can infer that even in a simulation case consisting largely of high-altitude SOs that don't move rapidly in measurement space, there is much value in incorporating lookahead over a horizon.



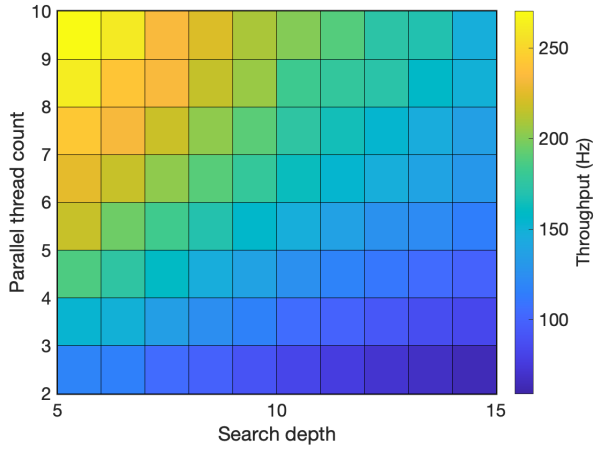
(a) Total unique objects tracked from each station over time

(b) Observing activity for each station

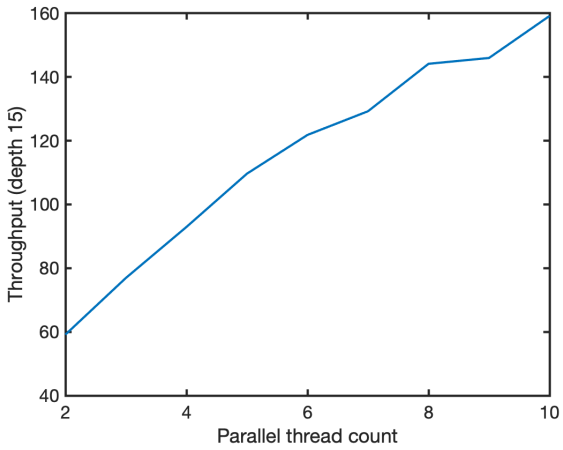
Fig. 7: Activity information for ground and space-based observers.

These insights are supported by Figure 7 in which it can be noted that the majority of the object population is very quickly observed. The sun-synchronous observer studies a very diverse set of objects, accumulating approximately 970 unique detections over the course of the simulation. All objects are observed by 21:00 UTC. Ground-based activity can be seen with switching at sunset and sunrise for both the Boulder, CO and Diego Garcia-based observers. Note that there is an overlap during which both ground-based sites are active. On the date of the simulation, sunrise at Diego Garcia occurred at approximately 1:15 UTC, while sunset in Boulder occurred at approximately 1:05 UTC. The sun-synchronous observer is always active.

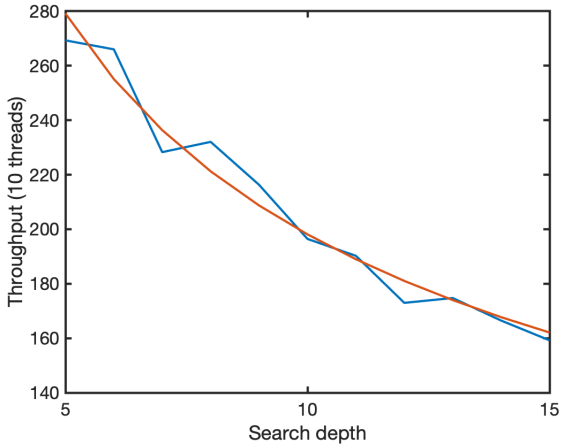
In addition to study of the evolution of the object catalog, the computational efficiency of the search methods are also analyzed. In Figure 8, search throughput in Hertz is analyzed as a function of search tree depth and breadth of parallelization. We define throughput as the number of times the search tree is recursively iterated through in entireity; that is, a single iteration is considered as a full traversal from the root node to a leaf node at the maximal search depth. In each subplot, analysis is performed for the simulation case, with the same set of 1000 objects and 3 observers for which decisions are made. Analysis is performed over 30 seconds of real time for each data point.



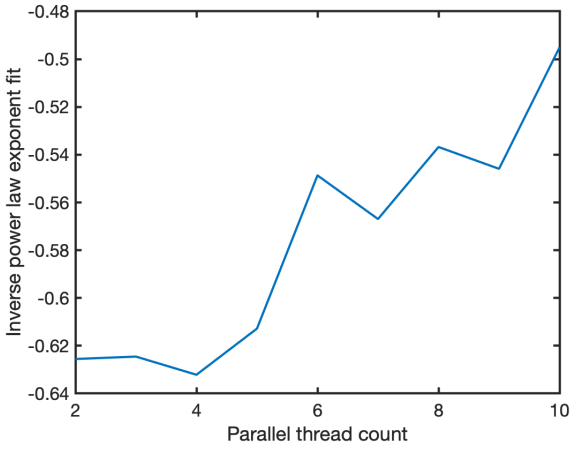
(a) Surface of search tree throughput as a function of depth and level of parallelization.



(b) Linear growth of throughput as thread count increases.



(c) Inverse decay in throughput as search depth increases.



(d) Power law behavior of throughput at a given depth as thread count increases.

Fig. 8: Studies of Monte Carlo tree search throughput in the parallelized implementation.

Theoretically, without competition, throughput can be expected to scale linearly as more threads are added. Approximately linear scaling can be noted in Figure 8b with some overhead as a result of tree node resource competition between threads. Because of synchronization challenges, threads inevitably experience idle time; this resource competition is concentrated at the root node, but as depth is increased, one can expect less overhead, since a thread will only spend $\frac{1}{d}$ percent of non-idle time on the root node.

Figure 8c demonstrates the trade between search depth and throughput at a fixed level of parallelization. This trade can be expected follow an inverse curve for a sequential implementation, but interestingly, we observe that throughput decreases at a slower rate when parallelization is applied. When throughput as a function is fit to a power law, the resulting best-fit exponent is found to increase as a function of parallel thread count. This is demonstrated in Figure 8d, and the power law increases to approximately an inverse square root fit with 10 threads utilized.

These results are quite useful for considering the search methodology in a massively parallel, distributed setting. Assuming these results apply and linear gains in throughput from further parallelization are obtained with k parallel threads and losses in throughput as a function of search depth d^a, $a > -1$, consider throughput $\tau(k, \gamma)$. If another

thread is added, and tree search is now performed to depth $d+1$, one can expect

$$\tau(k+1,\gamma+1) = \frac{k+1}{k}(\frac{d}{d+1})^a \tau(k,\gamma)$$

With this knowledge, one can then consider a trade space of search depth and computational resources for a desired search throughput.

Finally, we provide a quantification of simulation convergence toward an ideal solution as a function of search tree simulations. This can be challenging to visualize for a large scale problem such as a full day simulation, so the problem scope is reduced to a single ground-based observer tasked for 20 observations over a 5 minute period. To challenge the observer, 15 geostationary objects are placed within the field of view of the sensor; additionally, 15 low-Earth orbit objects are placed such that they may move out of view during the observation period. In Figure 9, the cumulative reward over the course of the tasking period is plotted against the number of times the search tree is traversed before each tasking decision is made. In each case, a search depth of 10 is utilized. The reward is computed as

$$R = \sum_{i=1}^{30} \Delta(\omega_r tr(P_r) + \omega_v tr(P_v)) \tag{33}$$

$$\omega_r = \frac{\mu}{R_{\oplus}^3} = 1.536e-6$$

$$\omega_v = 1$$

the weighted sum of the change in covariance traces, with values chosen to normalize the positional and velocity covariances. As is seen, the result converges toward a near-optimal solution relatively quickly in this scenario. Note that the initial data point with a single search tree traversal can be considered a greedy algorithm, in that actions are essentially sampled from the softmax policy without lookahead. As such, this Figure also presents further comparison to a greedy policy over the lookahead horizon. While this result suggests convergence toward the optimal policy at around 1000 search tree traversals, this behavior is not necessarily extensible to any decision problem. Generally, the convergence rate will be a function of the branching factor of the problem, and as more objects are studied and more observers are introduced, the search tree implementation should become more distributed to accomodate additional computational burden.

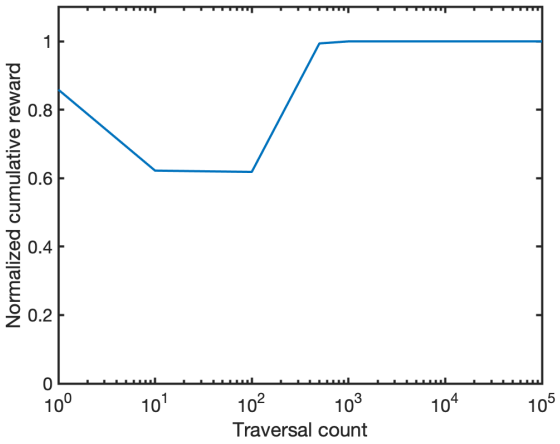


Fig. 9: Convergence of MCTS to increasing rewards.

5. DISCUSSION

In the future, a direction of interest is the addition of posing the tasking problem as a multi-objective optimization in the context of tree search. Leveraging multiple competing objectives leads to further combinatoric increases in

complexity, and MCTS has potential in terms of quickly finding near-optimal solutions. Recent literature [23–25] has applied a variety of strategies to the larger question of multi-objective optimization, from posing the problem as a POMDP and using greedy algorithms with developed value functions to the application of evolutionary algorithms.

Of particular interest is the means of quantifying the relative value of a search action as a competing objective. While [23] develops objective functions for undiscovered objects, an occupancy grid implementation is not sufficient for the more complex dynamics inherent to the space object tracking problem. Potential avenues for consideration could apply knowledge of the public catalog or treat the cardinality of objects tracked as an objective.

The geometry of angles-only detection of new objects leads to additional challenges and avenues of research. Detailed literature outlines the "too-short arc" problem for initial orbit determination, and generally, admissible regions [26] offer strategies for constraining sets of potential orbits for which the SO state is only partially observable. Worthy and Holzinger utilize admissible regions techinques to develop time-optimal follow-up campaigns for constraining an admissible region [27]. This research can be extended to incorporate information-theoretic weighting, and value of a follow-up observation could be extracted as a combination of the reduction of area of the projected admissible region, the mutual information between a potential measurement and previous knowledge of the vacuous prior, and the expected probablity a true follow-up measurement is actually achieved.

We also hope to integrate multi-target tracking methodologies in future research. In particular, Finite Set Statistics-based estimation approaches such as the PHD or multi-Bernoulli filter offer computationally efficient, robust approximations to the general multi-target Bayes filter. While this paper considers very simplified dynamics and assumes the challenge of measurement association may be resolved, MCTS can be extended to incorporate these methodologies.

Finally, we hope to explore a variety of rollout policy implementations. Insofar, these policies have been information-theoretic, but uninformed by any prior or online outcomes. This application is ripe for the introduction of further developments from reinforcement learning literature. Offline learning can be performed with large sets of training data. Public tasking histories can be sourced from sensor porfolios such as the Deep Space Network. In addition, online learning methodologies could be considered to tune the rollout policy over time. AlphaZero [28] applies a Deep Neural Network with parameters tuned over time through MCTS evaluation. Similar methodologies may improve the rollout heuristic applied and reduce the computational complexity of action generation.

6. CONCLUSION

Monte Carlo Tree Search was applied to solve a multi-sensor tasking problem, with theoretic bounds developed to support MCTS application. An analytic proof by induction is developed to support these bounds. The resultant algorithm is applied at a large scale consistant with modern catalogue maintenance needs. Results are consistent with and exceed information-theoretic methodologies, demonstrating the value of MCTS sample-based planning utilizing lookahead over a receding horizon. The potential for implementation of MCTS in an embarassingly parallel manner is also illustrated, and distributed versions of the methodology are of future interest. In addition, broad future applications for MCTS are outlined, especially the application of MCTS for multi-objective optimization.

This work is supported by a National Science Foundation Graduate Research Fellowship, as well as by the Draper Fellows program.

REFERENCES

[1] John Africano, Thomas Schildknecht, Mark Matney, Paul Kervin, Eugene Stansbery, and Walter Flury. A Geosynchronous Orbit Search Strategy. *Journal of Allergy and Clinical Immunology*, 1(2):357–369, 2004.

[2] Thomas Schildknecht. Optical surveys for space debris. *Astronomy and Astrophysics Review*, 14(1):41–111, 2007.

[3] R. Scott Erwin, Paul Albuquerque, Sudharman K. Jayaweera, and Islam Hussein. Dynamic sensor tasking for space situational awareness. *Proceedings of the 2010 American Control Conference, ACC 2010*, pages 1153–1158, 2010.

[4] Patrick S. Williams, David B. Spencer, and Richard S. Erwin. Coupling of estimation and sensor tasking applied to satellite tracking. *Journal of Guidance, Control, and Dynamics*, 36(4):993–1007, 2013.

[5] Carolin Frueh, Hauke Fielder, and Johannes Herzog. Heuristic and optimized sensor tasking observation strategies with exemplification for geosynchronous objects. *Journal of Guidance, Control, and Dynamics*, 41(5):1036–1048, 2018.

[6] Richard Linares and Roberto Furfaro. Dynamic Sensor Tasking for Space Situational Awareness via Reinforcement Learning. *AMOS*, pages 1–10, 2016.

[7] Richard Linares and Roberto Furfaro. An Autonomous Sensor Tasking Approach for Large Scale Space Object Cataloging. *Advanced Maui Optical and Space Surveillance Technologies Conference (AMOS)*, pages 1–17, 2017.

[8] Andris D. Jaunzemis, Marcus J. Holzinger, and K. Kim Luu. Sensor tasking for spacecraft custody maintenance and anomaly detection using evidential reasoning. *Journal of Aerospace Information Systems*, 15(3):131–156, 2018.

[9] Zachary Sunberg, Suman Chakravorty, Richard Scott Erwin, and Senior Member. Information Space Receding Horizon Control for Multisensor Tasking Problems. *IEEE Transactions on Cybernetics*, 46(6):1325–1336, 2016.

[10] Hyeong Soo Chang, Michael C. Fu, Jiaqiao Hu, and Steven I. Marcus. An adaptive sampling algorithm for solving Markov decision processes. *Operations Research*, 53(1):126–139, 2005.

[11] Levente Kocsis and Csaba Szepesvari. Bandit based Monte-Carlo Planning. *Lecture Notes in Computer Scienc*, 4212:282–293, 2006.

[12] David Silver and Joel Veness. Monte-Carlo Planning in Large POMDPs. *Advances in neural information processing systems (NIPS).*, pages 1–9, 2010.

[13] Adrien Couetoux, Jean-baptiste Hoock, Nataliya Sokolovska, and Olivier Teytaud. Continuous Upper Confidence Trees. *Learning and Intelligent Optimization*, (section 3):433–445, 2011.

[14] David Auger and Adrien Cou. Continuous Upper Confidence Trees with Polynomial Exploration – Consistency. pages 194–209, 2013.

[15] Zachary Sunberg and Mykel J. Kochenderfer. Online algorithms for POMDPs with continuous state, action, and observation spaces. 2018.

[16] Samuel Fedeler and Marcus Holzinger. Monte Carlo Tree Search Methods for Telescope Tasking. *AIAA SciTech Forum*, pages 1–19, 2020.

[17] Rémi Coulom. Efficient Selectivity and Backup Operators in Monte-Carlo Tree Search emi Coulom To cite this version : Efficient Selectivity and Backup Operators in Monte-Carlo Tree Search. *5th International Conference on Computer and Games*, 2006.

[18] S Gelly, Y Wang, R Munos, and O Teytaud. Modification of UCT with Patterns in Monte-Carlo Go. *INRIA Technical Report*, 6062(November):24, 2006.

[19] Sheldon Ross. Introduction to Stochastic Dynamic Programming. 1983.

[20] Joelle Pineau, Geoff Gordon, and Sebastian Thrun. Point-based value iteration: An anytime algorithm for POMDPs. *IJCAI International Joint Conference on Artificial Intelligence*, pages 1025–1030, 2003.

[21] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time Analysis of the Multiarmed Bandit Problem. *Machine Learning*, 47:235–256, 2002.

[22] Yizao Wang, Jean Yves Audibert, and Rémi Munos. Algorithms for infinitely many-armed bandits. *Advances in Neural Information Processing Systems 21 - Proceedings of the 2008 Conference*, pages 1729–1736, 2009.

[23] Hoa Van Nguyen, Hamid Rezatofighi, Ba-Ngu Vo, and Damith C. Ranasinghe. Multi-Objective Multi-Agent Planning for Jointly Discovering and Tracking Mobile Object. 2019.

[24] Han Cai, Yang Yang, Steve Gehly, Changyong He, and Moriba Jah. Sensor tasking for search and catalog

maintenance of geosynchronous space objects. *Acta Astronautica*, 175(May):234–248, 2020.

[25] Yun Zhu, Jun Wang, and Shuang Liang. Multi-objective optimization based multi-bernoulli sensor selection for multi-target tracking. *Sensors (Switzerland)*, 19(4), 2019.

[26] A. Milani, M. E. Sansaturio, G. Tommei, O. Arratia, and S. R. Chesley. Multiple solutions for asteroid orbits: Computational procedure and applications. *Astronomy & Astrophysics*, 431(2):729–746, 2005.

[27] Timothy S. Murphy and Marcus J. Holzinger. Generalized Minimum-Time Follow-up Approaches Applied to Tasking Electro-Optical Sensor Tasking. *Advanced Maui Optical and Space Surveillance (AMOS) Technologies Conference*, XX(X):1–33, 2017.

[28] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362(6419):1140–1144, 2018.