Final Project Proposal: Active Information Gathering for Non-Cooperative Resident Space Objects Using Monte Carlo Tree Search

Rahul Ayanampudi rayanam@stanford.edu
Sebastian Martinez sebasmp@stanford.edu
AA228/CS238, Stanford University

1. Goal

Future missions for in-orbit servicing and active debris removal will require a servicer spacecraft to autonomously approach and characterize a non-cooperative Resident Space Object (RSO). Before proximity operations can be attempted safely, the servicer must fully characterize the target. This process involves determining the relative orbit and 3D shape of the target Kruger et al. (2024).

Current observer systems are passive, leading to issues with range ambiguity and slow, uncertain convergence. This project will enable the agent to actively decide where to maneuver to acquire the most informative observations Kruger and D'Amico (2025). However, the spacecraft active sensing problem can be very complex as it involves image processing, navigation, guidance, and control subproblems coupled with the fact that maneuvers that are optimal for orbit determination (e.g., those that induce parallax) are not necessarily optimal for 3D shape reconstruction, which requires a diverse set of viewing angles.

In order to bound the problem, the team will assume the agent has perfect knowledge of the state, dynamics, and control for itself and the target. Hence, the team will focus on determining the optimal sequence of maneuvers to most efficiently reconstruct a target's 3D shape.

2. Decision Making

The simplified active sensing problem can be formulated as a Partially Observable Markov Decision Process (POMDP). The environment is a representation of the Low Earth Orbit (LEO) orbital space containing both the servicer (the agent) and the target RSO. The environment is partially observable.

The physical state (s_{phys}) of the servicer relative to the target (position, velocity, attitude) is assumed to be perfectly known to the agent. However, the true 3D shape of the RSO ($\mathcal{S}_{RSO_{shape}}$) is the environment's hidden state. The agent never knows $\mathcal{S}_{RSO_{shape}}$ directly. It can only infer it by taking observations.The agent's internal belief state ($b_{RSO_{shape}}$) is its own model of this hidden state.

The environment's state changes in response to the agent's actions. The physical part of the environment is deterministic. When the agent takes an action a (a Δv maneuver), the resulting transition to physical state s'_{phys} is exactly predictable using the relative orbital elements propagator (no randomness in the physics). The RSO's true shape $\mathcal{S}_{RSO_{shape}}$ is unknown to the agent and does not change over time.

The interaction loop between the agent and the environment is simple. The agent (MCTS planner) selects and executes an action a (a Δv maneuver) and the environment responds by transitioning the agent to a new, deterministic physical state s'_{phys} (via orbital mechanics). It provides the agent's camera with a new observation o (an image/scan) of the hidden $\mathcal{S}_{RSO_{shape}}$ from the new vantage point in the trajectory. The agent calculates its own reward $R = \text{InfoGain} - \text{Cost}$. The $Cost$ comes from the action a, and the InfoGain is calculated by the agent itself based on how the observation o changed its internal belief.

3. Sources of Uncertainty

The primary source of uncertainty is the agent's lack of knowledge about the RSO's 3D shape, which is modeled by the agent's internal belief state, b_{shape}. This could be represented as a probabilistic voxel grid, where each 3D cell i has an associated probability $P_i(\text{occupied})$ of containing mass. The mission goal is to drive these probabilities from their initial 0.5 (maximum uncertainty as all possible shapes are equally likely) to 0 or 1.

The second source is the inherent sensor noise in the observation process. Even with a perfect maneuver, the camera used to observe the target is not perfect. An observation o (e.g., a point cloud) is only a noisy, probabilistic (sampled from distribution) reflection of the target's true geometry, not a perfect ground-truth snapshot. The sensor noise is modeled within the observation likelihood function, $P(o|\mathcal{S}, s_{phys})$. This function defines the probability of getting observation o given a true underlying shape $\mathcal{S}$ and a viewing state s_{phys}.

References

Justin Kruger and Simone D'Amico. Autonomous navigation of a satellite swarm using inter-satellite bearing angles. *IEEE Transactions on Aerospace and Electronic Systems*, 2025.

Justin Kruger, Tommaso Guffanti, Tae Ha Park, Mason Murray-Cooper, Samuel Low, Toby Bell, Simone D'Amico, Christopher Roscoe, and Jason Westphal. Adaptive end-to-end architecture for autonomous spacecraft navigation and control during rendezvous and proximity operations. In *AIAA SCITECH 2024 Forum*, 2024. doi: 10.2514/6.2024-0001.

Appendix

The following solution framework is proposed. Note that this optional section is for additional feedback.

- **States** (S)
  - s_{phys} (Physical State): The fully observable physical configuration, assumed known perfectly.
    - Relative Position & Velocity: $\mathbf{r}_{rel}, \mathbf{v}_{rel}$
    - Relative Attitude: $\mathbf{q}_{rel}$
    - Time: t
  - b_{shape} (Belief State): The agent's probabilistic representation of the RSO's 3D shape (uncertainty).
    - Representation: Probabilistic voxel grid, where each cell i has probability $P_i(\text{occupied}) \in [0, 1]$.
    - Initial State: At $t = 0$, $b_{shape,0}$ has $P_i(\text{occupied}) = 0.5$ for all i (maximum entropy).
- **Actions** (A)
  - $\mathbf{a}_0$: Do Nothing (coast for time step Δt).
  - $\mathbf{a}_{1...N}$: Impulsive Δv Maneuvers (e.g., small/large thrusts in $\pm\mathbf{R}, \pm\mathbf{T}, \pm\mathbf{N}$ directions).
- **Transitions** (T)
  - MCTS uses a generative model $G(s, a)$ to sample a next state $s' = (s'_{phys}, b'_{shape})$.
  - **1. Propagate Physical State (Deterministic):**
    - The orbital propagator calculates the exact next physical state s'_{phys} after action a and time Δt.
    - $s'_{phys} = \text{Propagate}(s_{phys}, a, \Delta t)$
  - **2. Sample an Observation (Stochastic):**
    - Sample a hypothetical shape $\hat{\mathcal{S}}$ from the current belief b_{shape}.
    - Simulate a noisy observation o from the new vantage point s'_{phys} viewing $\hat{\mathcal{S}}$.
    - $o = \text{SimulateCamera}(s'_{phys}, \hat{\mathcal{S}})$
  - **3. Update Belief State (Deterministic):**
    - Compute the new belief b'_{shape} via a Bayesian update on b_{shape} using observation o.
    - $b'_{shape} = \text{Update}(b_{shape}, o, s'_{phys})$
  - **4. Return:**
    - The simulator G returns $s' = (s'_{phys}, b'_{shape})$.
- **Rewards** (R)

  - $R(s, a, s') = \text{InfoGain}(b_{shape}, b'_{shape}) - \text{Cost}(a)$
  - $\text{Cost}(a)$: Fuel penalty, $c \cdot ||\Delta v_a||$.
  - $\text{InfoGain}(b_{shape}, b'_{shape})$: Reduction in uncertainty (entropy).
    * Shannon Entropy: $H(b) = -\sum_{i \in \text{voxels}} [P_i \log_2(P_i) + (1 - P_i) \log_2(1 - P_i)]$
    * Information Gain: $\text{InfoGain} = H(b_{shape}) - H(b'_{shape})$

- **Observations (O)**
  - Simulated, noisy sensor reading (e.g., 2D image, 3D point cloud) from the camera model.
  - The evidence o is used to perform the Bayesian belief update $b'_{shape} = \text{Update}(b_{shape}, o, s'_{phys})$.