Project Title: Active Information Gathering for Non-Cooperative Resident Space Objects Using Reinforcement Learning

Team Members: Rahul Ayanampudi, Sebastian Martinez

Emails: rayanam@stanford.edu, sebasmp@stanford.edu

1 Motivation

Future autonomous servicing and debris-removal missions require a servicer spacecraft to approach and characterize a non-cooperative Resident Space Object (RSO). Before safe proximity operations can begin, the servicer must estimate the target's relative orbit and 3D shape Kruger et al. (2024). Passive sensing approaches suffer from range ambiguity and slow convergence, so the project objective is to enable an active information-gathering strategy in which the spacecraft selects maneuvers to acquire the most informative observations Kruger and D'Amico (2025).

The simplified active sensing problem can be formulated as a POMDP where the physical state is assumed to be perfectly known to the agent but the target shape is hidden. The agent updates its probabilistic voxel-grid belief of the shape with simulated camera observations. The goal is to implement and evaluate Monte Carlo Tree Search (MCTS) as the planning algorithm for selecting the most informative maneuvers that efficiently reduce uncertainty under orbital dynamics.

2 Current Progress

Orbital Elements Propagator and Observation Model.
The team has developed a custom simulator that defines the servicer and target spacecraft configurations, establishes their initial states, and propagates their relative orbital elements. The servicer can maneuver (impulsive Δv) and make observations of the target through its camera (modeled with specific resolution, field of view, and noise) at each time step. A ray casting algorithm (3D DDA) determines which cells of the target's probabilistic voxel grid the servicer can observe Amanatides and Woo (1987). The extent of characterization of the target is determined by computing the Shannon entropy of the agent's voxel grid belief.

MCTS Planner.
The framework for the MCTS planner has been implemented to select the most informative maneuvers. At each time step in the simulation, the planner builds a tree with finite depth. Each node in the tree represents a state and it has a branch for each of the 13 possible discrete actions. The agent can continue along its current path by making no maneuver or a small/large Δv in the radial, tangential, and normal (RTN) directions. The value, or reward, of each node is the difference between the information gained, which is quantified by the entropy reduction, and the cost of the action. Based on the action value function, $Q(s, a)$ computed with MCTS, of the root node (current state), the optimal action of the agent is determined. The agent's current simulation state, optimal action, associated reward, and corresponding next state are saved to a replay buffer. Then, the optimal action for the current state of the agent is taken in the simulation and the agent's state is propagated to the next time step. This process is repeated at the new state of the agent.

3 Revised Timeline

Week 9: Refine the reward structure, if necessary, and verify the current MCTS implementation. Analyze the MCTS pipeline: selection (UCB), expansion logic, rollout procedure, and backpropagation of rewards.

Thanksgiving: Perform hyperparameter tuning of the MCTS planner (tree depth, maximum iterations, and exploration constant) and the action discretization and simulation time steps. Evaluate MCTS planner performance by comparing against baseline comparisons of passive agents and agents that take random actions. Use this period to explore structural improvements to baseline MCTS framework.

Week 10 (Final Report): Conduct full experimental runs. Generate final plots, trajectory visualizations, and uncertainty profiles. Prepare final report and presentation.

4 Team Contributions

- **Rahul Ayanampudi**: Developed relative orbital dynamics simulator, probabilistic voxel grid, and observation scheme.
- **Sebastian Martinez**: Configured MCTS approach.

References

John Amanatides and Andrew Woo. 1987. A Fast Voxel Traversal Algorithm for Ray Tracing. *EuroGraphics* (1987).

Justin Kruger and Simone D'Amico. 2025. Autonomous Navigation of a Satellite Swarm using Inter-Satellite Bearing Angles. *IEEE Trans. Aerospace Electron. Systems* (2025).

Justin Kruger, Tommaso Guffanti, Tae Ha Park, Mason Murray-Cooper, Samuel Low, Toby Bell, Simone D'Amico, Christopher Roscoe, and Jason Westphal. 2024. Adaptive End-to-End Architecture for Autonomous Spacecraft Navigation and Control During Rendezvous and Proximity Operations. In *AIAA SCITECH 2024 Forum*. doi:10.2514/6.2024-0001