Proceedings of Machine Learning Research vol 168:1–11, 2022 4th Annual Conference on Learning for Dynamics and Control

Optimal Pointing Sequences in Spacecraft Formation Flying using Online Planning with Resource Constraints

Samuel Y. W. Low SAMMMLOW@STANFORD.EDU

Mykel J. Kochenderfer MYKEL@STANFORD.EDU

Stanford University, Department of Aeronautics and Astronautics

Editors: R. Firoozi, N. Mehr, E. Yel, R. Antonova, J. Bohg, M. Schwager, M. Kochenderfer

Abstract

In spacecraft formation flying, establishing inter-satellite communication links is critical for data exchange and relative satellite navigation. In large formations, establishing links between the reference chief and all deputy satellites can weigh heavily on mission execution time and resources. This study strives to find the optimal sequence of pointing decisions for a single chief spacecraft to the entire formation, while respecting practical resource constraints such as power budgeting. The sequential decision making problem is formulated as a Markov decision process (MDP) and solved as a shortest path problem. Two-body astrodynamics and rigid body dynamics are assumed in the simulation. We compared several policies: a random policy, two types of greedy policies, one-step look-ahead, and forward tree search. Policies were tested on a single demonstration scenario, and then tested on 1,000 Monte Carlo trials using randomized formation geometries. The total pointing mission execution times and the relative runtimes were assessed across these policies. Results show effectiveness in finding the shortest sequential pointing sequence, demonstrating promise in autonomous decision making for spacecraft attitude control in future missions.

1. Introduction

Spacecraft formation flying is a nascent and promising topic in astrodynamics, with well-studied value propositions in literature. Formation flying enables novel mission concepts such as geolocation, gravimetry and interferometry, while also re-distributing mission risk across multiple agents (Brown and Eremenko, 2006). Maintaining inter-satellite communications (ISC) is critical for enabling large spacecraft formations (Bristow, Folta, and Hartman, 2000). Most ISC links are unidirectional, trading higher gain for lower size-weight-power (SWaP), with a modern interest leaning towards laser links for future missions, with Starlink being a prime example (Chaudhry and Yanikomeroglu, 2021). The directional nature of ISC necessitates finding the fastest sequence of targets to communicate with the full formation in order to minimize mission execution time. In fact, this scenario can be thought of as a moving-target Travelling Salesman Problem (TSP), where the decision-making agent is the formation chief. Formation architectures where multiple deputies take reference and instructions from a chief are commonly adopted (D'Amico, 2010; Goh, Low, and Poh, 2019). In time-critical operations, enabling an efficient targeting sequence between a chief and its deputies not only saves valuable mission execution time, but also enables greater operational autonomy on the formation in the absence of ground segment control and intervention.

Challenges: There are three challenges in implementing the shortest-path pointing problem in spacecraft formations in practice. First, the targets (deputies) in the TSP are not static, but constantly in orbital motion. This adds the complexity of time-varying dynamics in the sequential decision making process. Second, resource constraints such as power budgeting and data handling must be accounted for in the decision space. Third, the typically limited processing power of space-grade computers, compounded by the NP-hardness of TSPs, imply that the policy space is realistically confined to online planning techniques; offline techniques such as value iteration or a full search are computationally infeasible in-orbit. These challenges partially explain why the adoption of autonomy in spacecraft has been slow compared to other vehicles, as echoed by Frost (2011).

Related Work: As Frost (2011) highlights, the lack of flight heritage in autonomous features creates a causality dilemma where autonomy is often forsaken for simpler criteria-based (if-else) decisions using a threshold or reference, as proposed by Low and Chia (2018); Low et al. (2022). In formation flying, a novel reformulation of relative states (D'Amico, 2010) have resulted in innovative criteria-based control to ensure passive collision-free safety in large formations (Koenig and D'Amico, 2018). Beyond criteria-based tasking and from an algorithmic standpoint, the moving-target TSP for orbital motion has not been extensively studied in literature. While Helvig et al. (2003) did formulate an algorithm to solve the linear constant velocity moving-target TSP, the kinematics were over-simplified for an astrodynamics context. Rather than an exact optimal solution to the TSP, online planning methods (Kochenderfer, 2015) have garnered recent interest to pursue an approximation to the optimal solution instead. Harris et al. (2019) formulated both attitude and orbit control planning as a partially observable Markov decision process (POMDP), applying Deep Q-Learning (DQL) to plan Martian orbit insertion and pointing decision sequences. Harris and Schaub (2020) then included stochastic policies in prior work, bounding the agent behaviour with a 'shield' against unsafe actions. Herrmann and Schaub (2021) planned a more thorough sequence of station keeping tasks using Monte Carlo tree search (MCTS) for spacecraft charging, ground station downlink, target imaging, and reaction wheel de-saturation. A similar shortest path pointing problem was studied by Eddy and Kochenderfer (2020) for Earth-fixed targets, constrained also by on-board resources, and solved using forward search and Monte Carlo tree search. However, for in-orbit targets such as spacecraft in formations, autonomous and sequential pointing in-orbit remains an open problem in literature, to the best of our knowledge.

Key Contribution: This paper proposes and investigates the performance of online planning for sequential pointing in spacecraft formations. The higher research direction it builds towards is to study the enabling of autonomy in sequential decision making in distributed space systems.

Implementation: In section 2, the astrodynamics simulation environment is defined, using only two-body motion for orbits and rigid body dynamics for spacecraft bodies. In section 3, the problem is formulated as a Markov decision process (MDP). States, actions, transitions and rewards are defined. In section 4, the implementation details of the policies are outlined. In section 5, a single demonstration scenario comprising 1 chief and 8 deputies is illustrated for clarity, with results from subsequent Monte Carlo trials. The policies implemented are: a random policy, two greedy policy types, single-step value iteration or look-ahead, and forward tree search. Throughout this paper, the only resource constraint considered is the battery power; so as not to obfuscate the key objective of finding the shortest path with too many resource constraints. Finally, in section 6, this paper concludes with remarks on future research directions.

2. Astrodynamics Environment

The physics environment is based on the QUADRANT astrodynamics and attitude control library, while the formation geometries were designed using the QLUSTER formation flying design library. The 'true' inertial frame is defined such that $\hat{n}_2$ is sun-pointing, $\hat{n}_3$ points out of the ecliptic, and $\hat{n}_1$ completes the right handed triad as shown in fig. 1. Axes $\hat{n}_1$ and $\hat{n}_2$ lie on the ecliptic plane. The chief orbit has a mean geocentric altitude of 600 km, an inclination of $i = 60°$ about the Earth equatorial plane, argument of perigee $\omega = 90°$, and right ascension $\Omega = 90°$. The chief serves as a relative origin about which deputies are initialized.

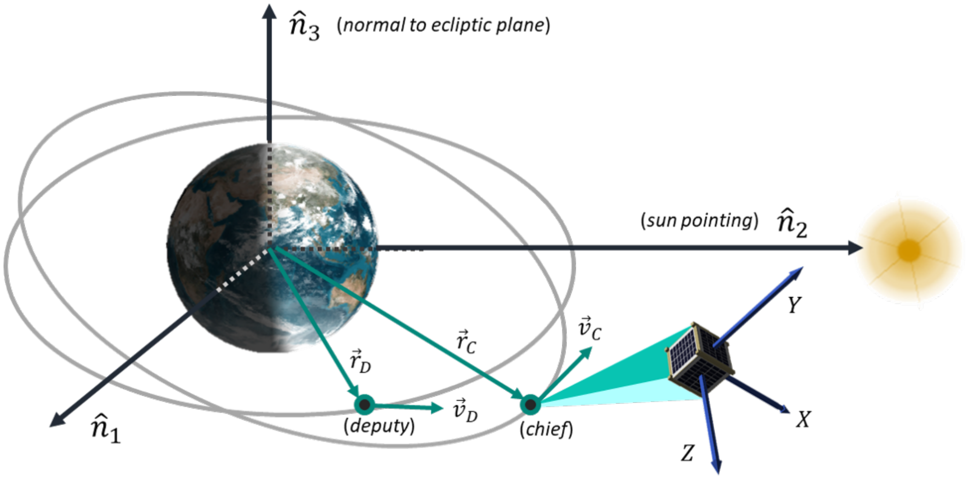


Figure 1: Inertial and body coordinate frames.

Each spacecraft is modelled as a rigid cube with a 3×3 principal inertia tensor $[I]$ having diagonal elements of 10 $\mathrm{kg\,m^2}$. Quaternions are the attitude coordinates used for target tracking. The angular acceleration $\dot{\boldsymbol{\omega}}_{B/N}$ and angular velocity $\boldsymbol{\omega}_{B/N}$ is between the body and inertial frame, expressed in body-fixed coordinates. The external torque $\boldsymbol{\tau}$ applied about the spacecraft center of mass is also expressed in body-fixed coordinates. The chief spacecraft is initialized with the body frame aligned with the inertial, without tumbling, $\boldsymbol{\omega}_{B/N} = \mathbf{0}$, and without external torques, $\boldsymbol{\tau} = \mathbf{0}$. Thus, the dynamics simply follow Euler's rigid body equations of motion:

$$[I]\dot{\boldsymbol{\omega}}_{B/N} = -\boldsymbol{\omega}_{B/N} \times [I]\boldsymbol{\omega}_{B/N} + \boldsymbol{\tau} \tag{1}$$

The chief's communications boresight and plane normal vector of the solar cells are assumed to point in the $+X$ direction as shown in fig. 1. For some deputy i, this information allows the construction of the direction cosine matrix $[RN]_i$ describing the rotation between the inertial frame and the chief-to-deputy communications pointing frame. For some deputy i chosen as the pointing target, the dynamics loop tracks how $[RN]_i$ evolves. Let $\boldsymbol{\rho}_i$ be the relative position of the chief to deputy i. This matrix $[RN]_i$ is then

$$\hat{c}_x = \frac{\boldsymbol{\rho}_i}{\|\boldsymbol{\rho}_i\|}, \qquad \hat{c}_y = \frac{\boldsymbol{\rho}_i \times \hat{n}_3}{\|\boldsymbol{\rho}_i \times \hat{n}_3\|}, \qquad \hat{c}_z = \hat{c}_x \times \hat{c}_y, \qquad [RN]_i = \begin{bmatrix} \hat{c}_{x1} & \hat{c}_{y1} & \hat{c}_{z1} \\ \hat{c}_{x2} & \hat{c}_{y2} & \hat{c}_{z2} \\ \hat{c}_{x3} & \hat{c}_{y3} & \hat{c}_{z3} \end{bmatrix} \tag{2}$$

If the chief decides to transit into sun-pointing, then the sun to inertial rotation matrix $[SN]$ will be used as the reference matrix:

$$[SN] = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \tag{3}$$

Unlike $[RN]$, $[SN]$ is not time varying because the sun is assumed inertially fixed and does not vary with the spacecraft motion.

With knowledge of the rotation between the inertial to communications frame $[RN]$, and body to inertial frame $[BN]$ (the latter is assumed to be measurable with attitude determination sensors in practice), the attitude error can be described using a composite rotation matrix

$$[BR] = [BN][RN]^T \tag{4}$$

This matrix translates directly into the quaternion $\boldsymbol{Q}_{B/R} = [q_0, q_1, q_2, q_3]^T$ describing the attitude error between the chief's current and target attitude. This quaternion can be fed into a commonly employed Lyapunov-based attitude control law that guarantees convergence (Schaub and Junkins, 1996; Tsiotras, 1996):

$$\boldsymbol{\tau} = -D_{gain}\, \boldsymbol{\omega}_{B/R} - P_{gain}\, [q_1, q_2, q_3]^T \tag{5}$$

where constants D_{gain} and P_{gain} are user-defined derivative and proportional control gains (see section 5), This control law is used for all pointing maneuvers. The information provided in this section is sufficient to help the reader replicate this environment.

3. Decision Making Process Formulation

We formulate the problem as a Markov decision process (MDP) because of the uncertainty inherent in this application. The chief must decide how to balance its choices between resource conservation (sun-pointing) and establishing links (deputy pointing). In addition, the chief needs to decide the optimal sequence of targets that establishes the shortest path, taking careful consideration of the dynamics. In an MDP, the state evolves probabilistically.

Actions are chosen at each epoch based on an observed state with a subsequent reward. The action space $\mathcal{A} = \{a_s, a_1, \ldots, a_N\}$. The state of the chief is a tuple (θ, S), where $\theta \in [0, 1]$ represents a power level of the chief, and $S \in \mathcal{S}$ is the discrete choice of target, where $\mathcal{S} = \{s_s, s_1, \ldots, s_N\}$. In $\mathcal{A}$ and $\mathcal{S}$, the subscript s refers to the sun-pointing mode, and numerical subscripts $i \in \{1, \ldots, N\}$ represent deputy-pointing mode.

The chief's state S will not re-visit a deputy s_i more than once. Transition between discrete states is dependent on the current power level θ and on the estimated control maneuver duration between current target i and next target j, denoted as Δ_{ij}. It also depends on a hyper-parameter λ, which balances power charging priority. Given an action a_j to point to deputy j, the transition probability from s_i into the sun-pointing mode s_s is

$$\alpha = (1 - \theta)^{\lambda}, \qquad \text{where} \qquad \theta \in [0, 1], \quad \lambda \in [0, \infty) \tag{6}$$

The intuition behind α is to inform the chief about the urgency of transiting into the charging state when power levels are low. Note that since $\theta \leq 1$, $\alpha \leq 1$. If λ decreases, then the chance of transiting into sun-pointing increases. The transition probabilities between states is given by

$$\begin{aligned} T(s' = s_s \mid s_i, a_s) &= 1 \\ T(s' = s_j \mid s_i, a_s) &= 0 \\ T(s' = s_s \mid s_i, a_j) &= \alpha \\ T(s' = s_j \mid s_i, a_j) &= 1 - \alpha \end{aligned} \tag{7}$$

A matrix representation of the transition used in Bellman updates is

$$T = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ \alpha & (1-\alpha) & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \alpha & 0 & 0 & \dots & (1-\alpha) \end{bmatrix} \tag{8}$$

For the pointing state S, the chief only makes a decision and hence a state transition when the body-to-reference attitude error $\boldsymbol{Q}_{B/R}$ converges to $\leq$ 0.1% error. This is assumed as the moment where pointing is successfully established. The power state θ can be generalized to other spacecraft resources, such as the amount of on-board data, the reaction wheel saturation levels, or as a timing parameter for periodic ground station contact, as was similarly formulated by Eddy and Kochenderfer (2020). Assuming a linear charging (sun-pointing) or discharging rate $\dot{\theta}$, the transition for power is a simple linear update

$$\theta \leftarrow \theta + \dot{\theta} \cdot \delta t \tag{9}$$

The distribution of immediate rewards is segmented into the sun-pointing state s_s and non-sun-pointing state s_i. In both cases, the rewards are a function of the power level θ prior to entering the state, and the expected maneuver duration needed to arrive at the state. The duration to transit the pointing from deputy i to j is Δ_{ij}, and from deputy i to the sun is Δ_{is}. Durations are estimated using a hallucinated propagation of the dynamics with larger time steps. These hallucinations only estimate Δ_{ij} and Δ_{is}. They are not actually executed in the dynamics. The reward functions are

$$R_s = \frac{\alpha}{\exp(\ \mu\ \Delta_{is}\ )} \qquad \text{and} \qquad R_j = \begin{cases} \frac{1-\alpha}{\exp(\ \mu\ \Delta_{ij}\ )}, & \text{if } j \notin V,\ j \neq i \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

where V is the set of already visited states.

The reward for sun-pointing is R_s, and the reward for pointing to an unvisited jth deputy is R_j for $j \notin V$ or $j \neq i$. The rewards include a soft-max hyper-parameter μ that controls the "hardness" of the rewards. These hyper-parameters λ and μ are important to tune to prevent reward hacking during the learning process. For example, if sun-pointing was rewarded more than proportionately over deputy-pointing, a possible reward hack could be to take the slowest battery-draining path and accumulate rewards from sun-pointing recharges, without finding the shortest path.

4. Online Planning Policies and Implementation

This section outlines the four policies we evaluated. A policy is executed only at a decision epoch. The decision epoch is triggered once the pointing error converges to $\leq 0.1\%$. In the time between decision epochs, both the attitude and orbits of all spacecraft are in constant propagation. The random and greedy policies were performed purely as performance benchmarks. One-step look-ahead and forward search are closed-loop planning techniques that account for information about future states, and hence can incorporate the astrodynamics in its planning.

4.1. Random Policy

The random policy instructs the chief at the decision epoch to select one out of all unvisited deputies with uniform probability and enters sun-pointing mode when $\theta \leq 0.2$.

4.2. Greedy Policy

The greedy policy instructs the chief at the decision epoch to select a deputy that is most "reachable" using a distance-like heuristic if $\theta > 0.2$; otherwise, the chief goes into sun-pointing mode. Two such heuristics are implemented: the principal rotation angle $\Phi_{B/R}$ and the angular velocity $\boldsymbol{\omega_{B/R}}$ between the chief's body frame B and the target frame R.

4.3. One-Step Look-Ahead

The one-step look-ahead policy initializes the utility $U(s) = 0$ for all states s. Then, at every kth decision epoch, the value function of the chief is updated using the Bellman equation once. The best pointing action maximizes the value at that epoch,

$$U_{k+1}(s) = \max_{a} \left(R(s,a) + \gamma \sum_{s'} T(s'|s,a)\, U_k(s') \right) \tag{11}$$

An important point for the look-ahead equation is that iterating the Bellman update more than once at the same kth decision epoch is not guaranteed to converge to the optimal value function due to the inherent time-varying dynamics (Lecarpentier and Rachelson, 2019). The number of ways in which the attitude dynamics can evolve for different permutations of paths taken are too large, making it computationally intractable to perform value iteration until convergence for all decision epochs. However, a single step look-ahead can still approximate the value function reasonably well, using a snapshot of the current orbital states and attitudes. Thus, eq. (11) is updated only once per decision epoch as the chief progresses through its pointing mission.

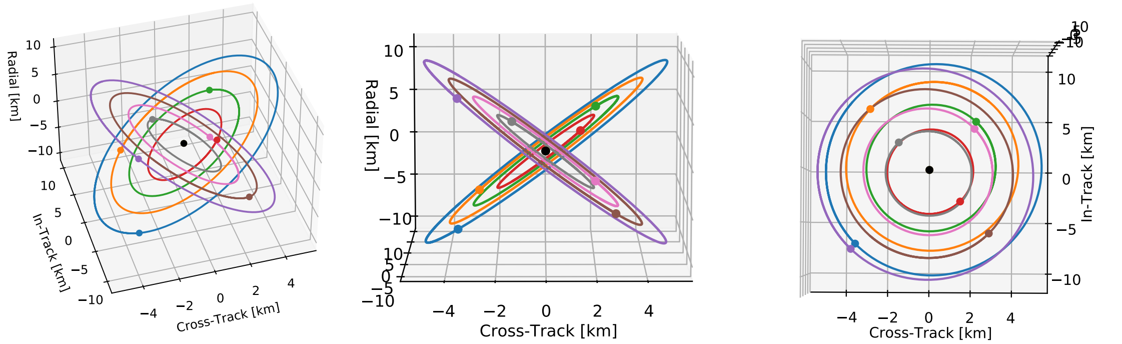
4.4. Forward Search

Forward search is a commonly used depth-first recursive search algorithm (Kochenderfer, 2015) that determines the best action from an initial state by expanding all possible transitions in the tree search to a specified depth, from a specific decision epoch. It returns the best action-value pair for each recursive call. Because of the dynamics, the rewards in eq. (10) and transition probabilities in eq. (7) are time-varying. Thus, every state transition moving forward in time (down the tree search in successive recursion) must also propagate both the orbit and attitude dynamics forward in time, so that the correct rewards and transitions can be computed. Dynamic programming can be used to store the chief orbit and attitude states before each recursive call, so that during the back-propagation of forward search (going back up the search tree), these states do not need to be re-computed. Also, to prevent revisiting deputies a second time, visited deputies are pruned out of the state space $\mathcal{S}$ when propagating down the tree, and re-populated back into $\mathcal{S}$ when returning up the search tree. Full implementation details are in the source code.

5. Simulation Results and Discussion

5.1. Simulation Results for a Single Scenario

This section illustrates the simulation results in two parts. First, an individual demonstration scenario was conducted using a formation geometry designed as two projected circular orbit (PCO) planes in fig. 2. The chief is at the origin of the Euler-Hill frame in fig. 2 flanked by eight deputies. The deputies are distributed evenly across the relative orbit plane, with relative phasing of $\pm 90^\circ$ between the argument of relative pericenter and the argument of latitude crossing.



(*a*) Isometric view (*b*) Frontal view (*c*) Nadir view

Figure 2: Initial formation geometry in the Euler-Hill frame, before geometric randomization.

In the demonstration scenario, the chief begins the mission with full power $\theta = 1$, and with its body frame aligned with the inertial frame. The time step of the dynamics is $0.1\,\mathrm{s}$, and the control feedback step is $1\,\mathrm{s}$ (or $1\,\mathrm{Hz}$). Table 1 summarizes the simulation parameters. Figure 3 shows the results of having the chief attempt to find the shortest pointing path in the demonstration scenario. The discrete changes in the color map correspond to state transitions, or equivalently a new pointing target. The annotations represent the states that are being transitioned into, with the red-marked S representing the state transition into sun-pointing mode.

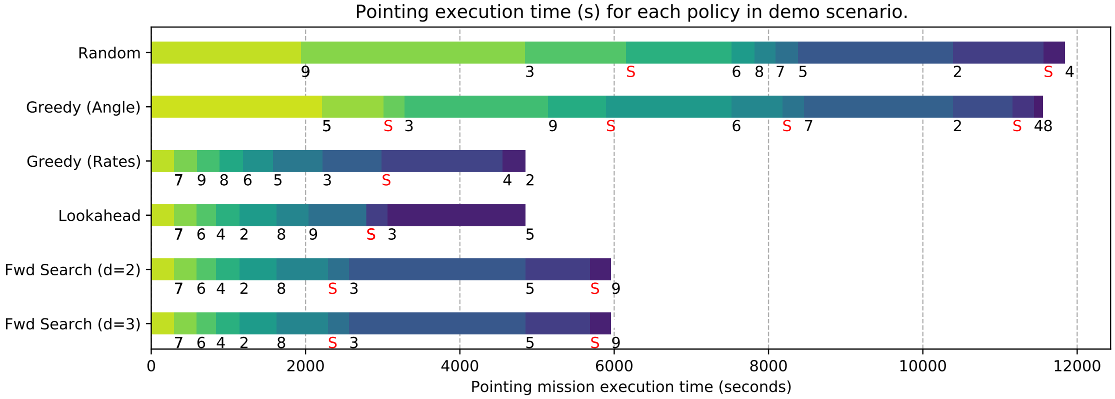


Figure 3: Results of pointing execution times for each policy with next-states annotated.

Table 1: Parameters for online planning and control.

γ	λ	μ	$\dot{\theta}_{drain}$	$\dot{\theta}_{charge}$	P_{gain}	I_{gain}	D_{gain}
0.75	2.0	1.7783×10^{-4}	-5.0×10^{-4}	2.5×10^{-3}	0.016	0.0	0.4

As expected, the random policy fared the worst. Surprisingly, a greedy policy with respect to the shortest angular distance fared almost as poorly as a random policy, but a greedy policy with respect to angular rates performed much better. A possible reason could be that the control convergence is more dependent on matching the relative orbit mean motion than simply the angular distance itself. Interestingly, forward search of depth 2 and 3 chose the same path. The one-step look-ahead performed best (for this scenario), completing its pointing mission to its 8 deputies within 4,848 seconds. While a single scenario provides clarity and validation on individual pointing sequences, we reran the same pointing mission through 1,000 Monte Carlo trials using randomized geometries. For each Monte Carlo trial, the geometry is initialized as pers fig. 2. Next, each deputy is given an additional and random radial and cross-track offset between 1 to 5 km. Finally, both the relative argument of latitude and pericenter, which describes the orientation of the relative orbit plane, will be randomly re-oriented with an angle $\in [-180°, +180°]$.

5.2. Simulation Results for Monte Carlo Trials

Figure 4 shows the histogram of mission execution times for all Monte Carlo trials. The statistics of the trials are shown in table 2 and runtimes relative to the random policy are shown in table 3.

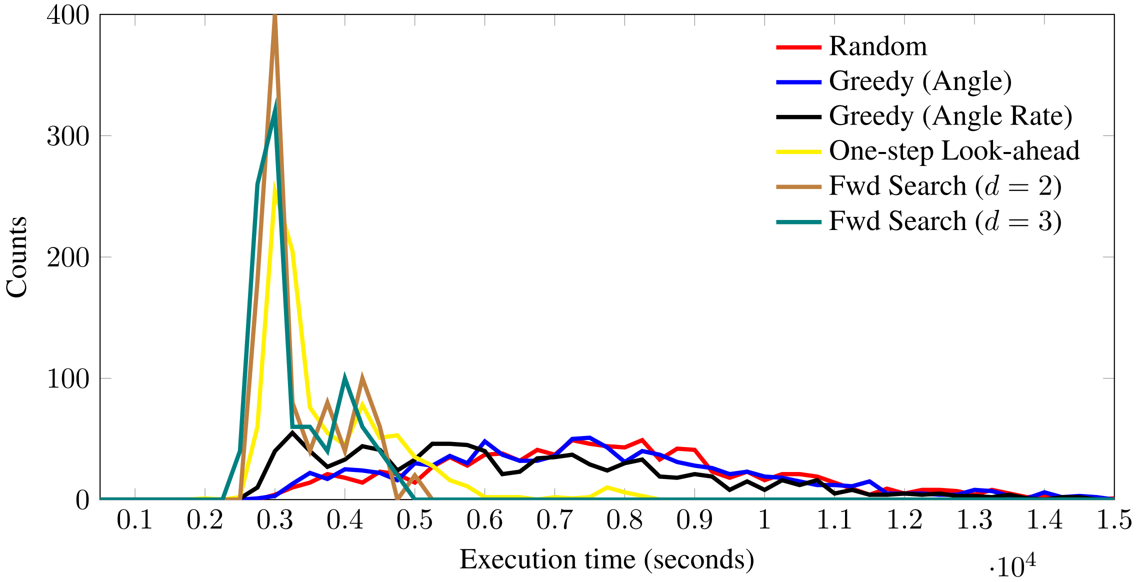


Figure 4: Histogram showing the distribution of total mission execution times for each policy, over 1000 randomized formation flying pointing missions, with bin resolution of 250 s.

Table 2: Mean and standard deviation of mission execution duration for each policy (in seconds).

Policy	Random	Greedy (Angle)	Greedy (Rate)	Look-Ahead	Fwd ($d = 2$)	Fwd ($d = 3$)
Mean	7613.683	7437.557	6327.291	3641.701	3207.360	3123.620
STD	2496.909	2592.223	2588.786	1006.586	608.178	572.167

Table 3: Normalized run-times of each policy relative to the random policy.

Random	Greedy (Angle)	Greedy (Rate)	Look-Ahead	Fwd ($d = 2$)	Fwd ($d = 3$)
1	∼1	∼1	2.857	11.548	103.571

5.3. Discussion of Results

The histogram in fig. 4 shows that the online planning policies appear to have a multi-modal distribution over mission execution duration. Each mode corresponds to the number of times sun-pointing had to be engaged. For the online planning policies, the dominant mode was the case where the chief could fully execute the pointing mission with only a single power charge. The secondary peaks corresponded to additional occurrences where the chief entered sun-pointing mode due to poor path planning resulting in the frequent need to recharge power. Interestingly, the greedy policy with respect the the angular distance fared just as unremarkably as a random policy. The greedy policy with respect to the angular rates fared slightly better, but averaged a mission execution time that was double that of the online planning policies. A likely reason is that the greedy policy prioritizes immediate rewards at the expense of leaving much longer paths in the future, which disregards future rewards succeeding the current horizon.

In terms of runtime, forward search with a depth of 3 provided marginal improvements in mission execution time, over forward search with a depth of 2, at the expense of increasing the relative runtime by an order of magnitude. The one-step look-ahead was computationally more efficient than forward search. This is because the utility does not have to be re-computed at the root node for every decision epoch in the same way forward search does. Only a single Bellman update is needed for a one-step look-ahead, and the utility from previous decision epochs is also carried over, discounted by γ. Although the motion of the dynamical system discounts the accuracy of the utility over time, it is still a useful approximation of the value function. Thus, the one-step look-ahead seems to be a promising compromise between mission execution duration and computational efficiency.

6. Conclusion

This study presented a framework for solving an astrodynamics pointing problem using online planning techniques. The proposed one-step look-ahead offers a computationally light-weight solution for planning the sequence of pointing to targets in-orbit. The proposed modified forward search algorithm offers slightly better performance at the expense of computational cost, largely due to the inclusion of the dynamics. The broader research goals of this study is driven to further the implementation of autonomous capabilities in decision making for spacecraft orbit and attitude control, which would make for an exciting future in unmanned space flight.

Acknowledgements: The authors of this paper would like to thank members of the Stanford Intelligent Systems Lab (SISL) and the Stanford Space Rendezvous Lab (SLAB) for their helpful insights and discussions on the topic of this paper.

Source Code: The code to this scenario is freely available on GitHub, under the QUADRANT spacecraft attitude tracking and control library. The formation flying orbit design and visualisation was performed using the QLUSTER formation flying library.

References

John Bristow, David Folta, and Kate Hartman. A formation flying technology vision. In *Space 2000 Conference and Exposition*, 2000.

Owen Brown and Paul Eremenko. The value proposition for fractionated space architectures. In *Space 2006*, 2006.

Aizaz U. Chaudhry and Halim Yanikomeroglu. Laser intersatellite links in a starlink constellation: A classification and analysis. *IEEE Vehicular Technology Magazine*, 16(2):48–56, 2021.

Simone D'Amico. *Autonomous Formation Flying in Low Earth Orbit*. PhD thesis, TU Delft, 2010.

Duncan Eddy and Mykel Kochenderfer. Markov decision processes for multi-objective satellite task planning. In *IEEE Aerospace Conference*, 2020.

Chad R. Frost. Challenges and opportunities for autonomous systems in space. In *National Academy of Engineering's U.S. Frontiers of Engineering Symposium*, 2011.

S. T. Goh, K. S. Low, and E. K. Poh. Leader-followers satellite formation control for low-thrust small satellite application. In *International Symposium on Space Technology and Science (ISTS)*, 2019.

Andrew T. Harris and Hanspeter Schaub. Spacecraft command and control with safety guarantees using shielded deep reinforcement learning. In *AIAA Scitech Forum*, 2020.

Andrew T. Harris, Thibaud Teil, and Hanspeter Schaub. Spacecraft decision-making autonomy using deep reinforcement learning. In *AAS/AIAA Space Flight Mechanics Meeting*, 2019.

C. S. Helvig, Gabriel Robins, and Alex Zelikovsky. The moving-target traveling salesman problem. *Journal of Algorithms*, 49(1):153–174, 2003.

Adam Herrmann and Hanspeter Schaub. Autonomous spacecraft tasking using monte carlo tree search methods. In *AAS/AIAA Space Flight Mechanics Meeting*, 2021.

Mykel J. Kochenderfer. *Decision Making Under Uncertainty: Theory and Application*. MIT Press, 2015.

Adam W. Koenig and Simone D'Amico. Robust and safe n-spacecraft swarming in perturbed near-circular orbits. *Journal of Guidance, Control, and Dynamics*, 41(8):1643–1662, 2018.

Erwan Lecarpentier and Emmanuel Rachelson. Non-stationary markov decision processes, a worst-case approach using model-based reinforcement learning. In *Advances in Neural Information Processing Systems*, 2019.

Samuel Y. W. Low and Yong Xian Chia. Assessment of orbit maintenance strategies for small satellites. In *AIAA / USU Small Satellite Conference*, 2018.

Samuel Y. W. Low, Yongjun Moon, Wen Tao Liu, and Chek-Wu Tan. Designing a reference trajectory for frozen repeat near-equatorial low earth orbits. *Journal of Spacecraft and Rockets*, 59(1): 84–93, 2022.

Hanspeter Schaub and John L. Junkins. Stereographic orientation parameters for attitude dynamics: A generalization of the rodrigues parameters. *Journal of the Astronautical Sciences*, 44(1):1–19, 1996.

Panagiotis Tsiotras. Stabilization and optimality results for the attitude control problem. *Journal of Guidance, Control, and Dynamics*, 19(4):772–779, 1996.